



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

DZIEDZINA: NAUKI INŻYNIERYJNO-TECHNICZNE

DYSCYPLINA: INFORMATYKA TECHNICZNA I TELEKOMUNIKACJA

ROZPRAWA DOKTORSKA

*Metodyka i narzędzie informatyczne dla realizacji
projektów wdrożeniowych systemu ERP
typu roll-out oraz upgrade systemu*

Autor: mgr inż. Adam Domagała

Promotorzy rozprawy: prof. dr. hab. inż. Jarosław Wąs
dr hab. inż. Edyta Kucharska, prof. AGH

Praca wykonana: Akademia Górniczo - Hutnicza im. Stanisława Staszica
w Krakowie
Wydział Elektrotechniki, Automatyki, Informatyki
i Inżynierii Biomedycznej

Kraków 2024



AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY

FIELD OF SCIENCE: ENGINEERING AND TECHNOLOGY

SCIENTIFIC DISCIPLINE: INFORMATION AND COMMUNICATION TECHNOLOGY

DOCTORAL THESIS

*Methodology and IT tools for implementing ERP system
rollout and upgrade projects*

Author: mgr inż. Adam Domagała

Supervisors: prof. dr. hab. inż. Jarosław Wąs
dr hab. inż. Edyta Kucharska, prof. AGH

Completed in: AGH University of Krakow
Faculty of Electrical Engineering, Automatics, Computer
Science and Biomedical Engineering

Krakow, 2024

PREAMBUŁA

Badania te są wynikiem przemysłowego programu doktoranckiego, finansowanego przez Ministerstwo Nauki i Szkolnictwa Wyższego (MNiSW) i realizowanego we współpracy z InfoConsulting Poland Sp. z o.o. i Akademią Górniczo-Hutniczą, Wydziałem Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej.

Streszczenie

Celem niniejszej pracy doktorskiej było zmodernizowanie procesu wdrażania systemów ERP w firmie. W ramach przygotowanego rozwiązania autor rozwinął obecną metodykę wdrożeniową systemów ERP oraz opracował i usprawnił narzędzia informatyczne wspomagające prowadzenie i realizację projektów wdrożeniowych. Przeprowadzona została analiza literatury w zakresie podejmowanej w rozprawie tematyki ze szczególnym uwzględnieniem metodyk wdrożeniowych systemów informatycznych oraz architektury współczesnych systemów ERP. Wskazano podział projektów wdrożeniowych na typy, w szczególności z uwzględnieniem takich typów projektów jak Upgrade i Rollout. Zaproponowana została propozycja zmian w zakresie działań realizowanych w ramach poszczególnych faz i ich wykorzystania w zależności od typu projektu. Stworzone zostało narzędzie „InfoSearch” wspierające proces wytwarzania modyfikacji w oprogramowaniu, które wykorzystuje algorytmy klasteryzacji i wyznaczania podobieństwa tekstów.

Praca koncentruje się na dwóch kluczowych zagadnieniach: ulepszeniu metodyki wdrożeniowej oraz opracowaniu i optymalizacji narzędzi informatycznych wspierających realizację projektów. W ramach pierwszego obszaru, zaktualizowano metodykę wdrożeniową, szczególnie dla projektów typu **upgrade** i **rollout**, co pozwoliło na znaczne skrócenie czasu ich realizacji dzięki eliminacji dwóch faz projektowych. Drugi obszar badań obejmował zaprojektowanie nowych narzędzi, takich jak wyszukiwarka „InfoSearch” oraz system wspomagania decyzji, co w efekcie zwiększyło efektywność realizacji projektów.

Opracowane narzędzia zostały wprowadzone do użytku w firmie, a ich skuteczność została potwierdzona w praktyce. Wyniki przeprowadzonych badań stanowią istotny wkład w rozwój wiedzy na temat optymalizacji wdrożeń systemów ERP oraz zarządzania projektami IT.

Abstract

The objective of this doctoral dissertation was to modernize the ERP system implementation process within a company. In the proposed solution, the author enhanced the existing ERP implementation methodology and developed improved IT tools to support the management and execution of implementation projects. A literature review was conducted, focusing on methodologies for implementing information systems and the architecture of contemporary ERP systems. The classification of implementation projects into types was outlined, with special emphasis on categories such as Upgrade and Rollout. Modifications were suggested regarding the activities carried out in various phases of implementation, adapting these phases based on the project type. Furthermore, a tool named "InfoSearch" was developed to support the process of generating software modifications, utilizing clustering algorithms and text similarity analysis.

The dissertation focuses on two key aspects: improving the implementation methodology and designing and optimizing IT tools to support project execution. Regarding the first area, the implementation methodology was updated, particularly for projects such as upgrades and rollouts, which resulted in a significant reduction in project duration by eliminating two project phases. The second area of research involved the creation of new tools, such as the "InfoSearch" search engine and a decision support system, which led to increased project efficiency.

The developed tools have been implemented in the company, and their effectiveness has been confirmed in practice. The results of the conducted research provide a significant contribution to the body of knowledge concerning the optimization of ERP system implementations and IT project management.

Spis treści

1. Wstęp	11
1.1. Geneza pracy	11
1.2. Cel pracy	12
1.3. Zawartość.....	14
2. Charakterystyka projektów wdrożeniowych systemów ERP	15
2.1. Charakterystyka systemów ERP.....	15
2.1.1. Definicja projektu wdrożeniowego	18
2.1.2. Znaczenie projektu wdrożeniowego dla przedsiębiorstw	18
2.2. Charakterystyka typów projektów wdrożeniowych	20
2.2.1. Implementacja.....	22
2.2.2. Upgrade.....	23
2.2.3. Rollout	25
2.2.4. Reimplementacja.....	26
3. Przegląd metodyk projektów informatycznych	29
3.1. Model kaskadowy	29
3.2. Agile	31
3.3. Scrum.....	34
3.4. Kanban.....	36
3.5. PRINCE2	38
3.6. PMBOK	41
3.7. Podsumowanie	44
4. Architektura systemu i metodyka projektowa kluczowych dostawców ERP	47
4.1. SAP.....	48
4.1.1. Architektura	48
4.1.2. Metodyka wdrożeniowa.....	51
4.2. Microsoft Dynamics 365	54

4.2.1. Architektura	54
4.2.2. Metodyka wdrożeniowa.....	58
4.3. IFS.....	63
4.3.1. Architektura	63
4.3.2. Metodyka wdrożeniowa.....	67
4.4. Podsumowanie przeglądu systemów	67
5. Zaproponowana metodyka i fazy projektu wdrożeniowego systemu ERP	71
5.1. Fazy projektu wdrożeniowego.....	72
5.2. Faza I - Przygotowanie projektu.....	75
5.3. Faza II - Projektowanie i prototypownie rozwiązania/ Przygotowanie Koncepcji biznesowej	77
5.3.1. Kluczowe różnice według rodzaju projektu.....	77
5.3.2. Działania projektowe Fazy II.....	79
5.4. Faza III - Przygotowanie Rozwiązania / Potwierdzenie Prototypu	82
5.5. Faza IV - Przygotowanie rozwiązania	90
5.6. Faza V - Uruchomienie produkcyjne.....	95
6. Zaproponowane i wdrożone autorskie rozwiązania i narzędzia wspierające projektowanie i wytwarzanie oprogramowania	99
6.1. Narzędzia nadrzędne względem faz projektowych	100
6.1.1. VAP - Value Added Product - optymalizacja kosztowa.....	101
6.1.2. Uniwersalna struktura projektu i działań projektowych	105
6.2. Faza I Przygotowanie projektu	109
6.3. Faza II Projektowanie i prototypowanie rozwiązania/ Przygotowanie Koncepcji biznesowej	112
6.3.1. Aplikacja InfoSearch	113
6.4. Faza III.....	121
6.4.1. Reorganizacja procesu wytwarzania oprogramowania i narzędzia wspierające.....	121
6.4.2. Cykl Deminga - programowanie CRIM w ramach portfela projektowego.	130
6.5. Faza IV.....	142
6.5.1. Szablon scenariusza rozruchu produkcyjnego jako narzędzie wspierające zespół wdrożeniowy.....	142
6.6. Faza V	151
6.6.1. Przekazanie projektu wdrożeniowego do serwisu	151

6.6.2. Narzędzia wspierające zarządzanie projektem - nau czki projektowe (z eng. lessons learnt)	154
7. Podsumowanie	161

1. Wstęp

1.1. Geneza pracy

Optymalizacja pracy, zmniejszenie kosztów wytwarzania, zwiększenie zysków i zwiększenie kontroli działania to jedne z wielu dylematów, z którymi na co dzień boryka się każde z przedsiębiorstw i organizacji. W przypadku dużych i średnich przedsiębiorstw, podstawą dla efektywnego działania jest zastosowanie w pracy narzędzi informatycznych wspomagających proces zarządzania. Obecnie w większości dużych firm taką rolę spełnia rozwiązanie w postaci systemów ERP. Systemy te są również adaptowane do wspomaganie pracy w różnym zakresie przez średnie i małe przedsiębiorstwa.

Zastosowanie systemu ERP w przedsiębiorstwie to długotrwały, kosztowny i skomplikowany proces, który wymaga kompletnego odwzorowania systemów biznesowych organizacji w aplikacji. Dlatego też konieczne jest przeprowadzenie, w ramach projektu wdrożeniowego, co, bynajmniej, nie jest tożsame z prostą instalacją aplikacji/oprogramowania.

Projekty wdrożeniowe są realizowane najczęściej przy użyciu metodyk standardowych, takich jak PRINCE2, PMBOK lub własnych metodyk bazujących na standardowych rozwiązaniach np. IFS Implementation, SAP Activate. Metodyki te dobrze sprawdzają się podczas realizacji standardowych projektów wdrożeniowych systemu ERP, jednak klasyczne wdrożenia występują coraz rzadziej. Jest to spowodowane wieloletnią informatyzacją większości dużych i rozwiniętych przedsiębiorstw, a rynek systemów ERP jest w znacznym stopniu nasycony. Obecnie istnieje przede wszystkim potrzeba aktualizacji istniejącego oprogramowania w firmie, z powodu dostosowania do najnowszych standardów technologicznych lub rozwoju przedsiębiorstwa. W takich przypadkach realizowane są projekty wdrożeniowe typu upgrade systemu do wyższej wersji, rollout na kolejną spółkę lub rozbudowa ERP o dodatkowe moduły. Znane metodyki, nie są w pełni dostosowane do realizacji tego typu projektów, dlatego potrzebna jest optymalizacja narzędzi i metodyki. Stosowane obecnie metodyki i dostępne rozwiązania posiadają wiele wad. Do najważniejszych należą:

- ściśle określony model dla procesu wdrożeniowego niezależnie od przyjętej w umowie/planie metody projektowania i finansowania rozwiązania,
- nieefektywne wykorzystanie budżetu projektowego ze względu na następstwo etapów (model kaskadowy),
- brak mechanizmów wsparcia w zakresie gotowych narzędzi dla zespołu wdrożeniowego w przypadku realizacji projektów dotyczących upgrade'u systemu, rolloutu dla nowej firmy/oddziału jednej grupy kapitałowej czy projektu polegającego na przeniesieniu rozwiązania do firmy o analogicznych procesach produkcyjnych lub asortymencie.

Niniejsza praca stanowi uzupełnienie brakujących rozwiązań w zakresie metodyk prowadzenia *wdrożeniowych* i *rewdrożeniowych* systemów ERP.

1.2. Cel pracy

Celem realizowanej pracy jest wytworzenie nowatorskiego rozwiązania do wspomagania procesu wdrażania dużych systemów informatycznych klasy ERP ze szczególnym uwzględnieniem projektów typu Rollout i Upgrade systemu.

Zaproponowane rozwiązanie ma za zadanie usprawnienie procesu wdrożeniowego systemu ERP, co w konsekwencji spowoduje ograniczenie kosztów administracyjnych w procesie implementacji systemów ERP oraz przyspieszenie realizacji zadań w łańcuchu dostawy oprogramowania. Optymalizacja procesu zostanie osiągnięta poprzez opracowanie dodatkowych narzędzi informatycznych, procedur stanowiskowych i procesowych oraz instrukcji z zakresu *najlepszych praktyk* dostosowanych do rodzaju wdrożenia oraz typu organizacji, której wdrożenie dotyczy. Podstawowym zadaniem wytworzonego narzędzia back office będzie wypracowanie unikalnych rozwiązań wspierających kierownika projektu w zarządzaniu projektem poprzez częściowy *Reengineering* procesu wdrożeniowego. Istotnym zadaniem jest również efektywniejsze wykorzystanie zasobów ludzkich, finansowych, technologicznych, oraz informacyjnych w ramach wytwarzanego oprogramowania.

Cel główny projektu zostanie osiągnięty poprzez realizację następujących celów szczegółowych:

1. Analiza stosowanych metodyk projektów informatycznych pod kątem przydatności w procesie wdrożenia systemu ERP
2. Analiza i porównanie dostępnych metodyk projektowych kluczowych dostawców ERP pod kątem wspierania procesów *rewdrożenia* systemu

3. Opracowanie propozycji zmian w zakresie działań realizowanych w ramach poszczególnych faz i ich wykorzystania w zależności od typu projektu.
4. Usprawnienie procesu wytwarzania nowych technologii – modyfikacji systemu ERP na potrzeby biznesu klienta.
5. Usprawnienie procesu akceptacji i rozliczania prac w projektach realizowanych wg metodyki „Time & Materials”, czyli wdrożenie systemu realizowanego w oparciu o określony czas realizacji i koszt jednostkowy osobo-dnia.
6. Opracowanie mobilnego dostępu „on-line” do informacji, co usprawni komunikację i dostęp do informacji projektowej dla całego zespołu wdrożeniowego: dla klienta do aktualnie realizowanych zadań w ramach etapu, oraz dla konsultanta, programisty i kierownika projektu do realizowanych zadań oraz ewentualnych opóźnień w ich realizacji.
7. Wytworzenie modyfikacji do istniejących rozwiązań informatycznych mających na celu usprawnienie procesu zarządzania projektem wdrożenia systemu ERP w zakresie:
 - (a) realizacji projektu re-ERP (podniesienie wersji (upgrade) systemu),
 - (b) realizacji projektu rollout (rozszerzenie systemu dla kolejnej firmy w ramach grupy kapitałowej).
8. Opracowanie nowych procedur dla realizacji projektów wdrożeniowych związanych z obecnie wykorzystywaną metodyką kaskadową. Zakres usprawnień dotyczy wszystkich faz projektowych, ale przede wszystkim fazy projektowania i przygotowania rozwiązania. Potencjalnie zmiany mogą spowodować przyspieszenie realizacji projektu i obniżenie kosztów wytwórczych. Podejście to pozwala na wykorzystanie metodyki zwinnej do realizacji projektu i właściwe wykorzystanie budżetu w przypadku rozliczeń „Time and Materials”.
9. Opracowanie zestawów dokumentacji projektowej metodyki wdrożeniowej systemu IFS oferujących rozwiązania dostosowane do potrzeb każdego typu projektu, co usprawni kolejne etapy realizacji projektu, ograniczy ryzyko związane z ewentualnymi zmianami wynikającymi z niedostosowania metodyki do realizowanego typu projektu.

Tezę rozprawy doktorskiej można sformułować następująco: Modyfikacja metodyki wdrożeniowej i narzędzi informatycznych wspierających prowadzenie projektów umożliwi zmniejszenie pracochłonności zespołu wdrożeniowego w projektach wdrożeniowych typu Rollout i Upgrade.

1.3. Zawartość

Rozprawa - doktorat wdrożeniowy jest odpowiedzią na zidentyfikowane potrzeby informatyczno-biznesowe organizacji w zakresie optymalizacji głównego produktu spółki - wdrożeń systemów ERP.

W rozdziale pierwszym przedstawiono genezę i cele rozprawy, które określają zakres prac badawczych i wdrożeniowych.

W rozdziale drugim opisana została charakterystyka systemów ERP, a także przedstawiona została definicja projektu wdrożeniowego i jego znaczenie dla przedsiębiorstw. Dodatkowo przedstawiony został podział projektów wdrożeniowych ze względu na różne rodzaje i specyfikę wdrożenia: implementacja, rollout, upgrade, reimplementacja.

W rozdziale trzecim przedstawiono przegląd i charakterystykę wybranych metodyk projektowych systemów informatycznych: model kaskadowy, Agile, Scrum, Kanban, PRINCE2, PMBOK. Zaprezentowane zostało również uzasadnienie, dlaczego w procesie wdrożenia systemu ERP nie jest wskazane stosowanie ogólnych metodyk, lecz wytworzenie dedykowanych metodyk wdrożeniowych, które są przystosowane zarówno do typu wdrożenia, jak i charakterystyki organizacji, w której wdrażany jest system ERP.

W rozdziale czwartym została dokonana analiza architektur, metodyk i możliwości rozbudowy oprogramowania w zakresie największych systemów klasy ERP - 3 główne systemy według rankingu Gartnera. Zidentyfikowane zostały największe wyzwania w zakresie możliwości wsparcia procesu wdrożeniowego - praktyczna adaptacja metodyki wdrożeniowej w projekcie.

W rozdziale piątym zawarta została propozycja zmian do obecnie obowiązującego modelu kaskadowego w zakresie inżynierii oprogramowania systemów spadkowych. Przedstawione zostały praktyczne zmiany zaadaptowane w organizacji projektów wdrożeniowych oraz propozycje dalszego rozwoju narzędzi usprawniających proces wytwórczy oprogramowania. W znacznym stopniu propozycje zmian, wsparcie i optymalizacja procesu dotyczą etapów projektowania i przygotowania rozwiązania.

W rozdziale szóstym zaproponowane zostały narzędzia i rozwiązania, które zostały utworzone w celu optymalizacji procesu wdrożeniowego. Rozdział opisuje również zmiany rozwojowe we wcześniej używanych rozwiązaniach integracyjnych i wdrożenie nowych narzędzi dla poszczególnych faz projektu. Przedstawiono również praktyczną adaptację nowych rozwiązań, które wspierając projekt wdrożenia niezależnie od fazy projektu i jego poziomu zaawansowania.

Rozdział siódmy stanowi podsumowanie rozprawy zawierające omówienie efektów pracy, wnioski końcowe oraz możliwe kierunki rozwoju zarówno z perspektywy naukowej i biznesowej podejmowanej tematyki systemów ERP.

2. Charakterystyka projektów wdrożeniowych systemów ERP

W rozdziale przedstawiona została definicja i charakterystyka zintegrowanych systemów informatycznych ERP wspomagających zarządzanie różnego typu przedsiębiorstw oraz znaczenie projektu wdrożeniowego i rewdrożeniowego dla przedsiębiorstw.

Rozdział został podzielony na dwie części. Pierwsza część przedstawia charakterystykę systemów ERP, ich architekturę i zastosowanie w zarządzaniu przedsiębiorstwami różnej wielkości i branż. Druga część zawiera definicje projektu wdrożeniowego oraz opisuje teoretyczny podział projektów wdrożeniowych systemów ERP na różne typy: Implementacja, Upgrade, Rollout oraz Reimplementacja.

2.1. Charakterystyka systemów ERP

Systemy ERP to komercyjne pakiety oprogramowania, które umożliwiają integrację danych i procesów biznesowych w całej organizacji [1]. Według Murraya [2] systemy ERP to systemy informatyczne, które integrują informacje i procesy, w ramach, i pomiędzy obszarami w organizacji. Ich celem jest integracja wszystkich danych i procesów biznesowych ze wszystkich działów w jednym systemie komputerowym, który kontroluje te procesy dzięki scentralizowanej bazie danych. System ERP ma wpływ na pracowników we wszystkich obszarach firmy, od produkcji, logistyki, magazynu, sprzedaży przez finanse, zakupy, aż po zasoby ludzkie. W literaturze możemy znaleźć inne definicje systemu ERP [3, 4]. ERP został zaprojektowany w celu wprowadzenia: zmian organizacyjnych, usprawnienia i ujednolicenia/standaryzacji procesów biznesowych oraz ułatwienia integracji danych.

System ERP koncentruje się na integracji większości działów organizacji, funkcji i procesów w ramach jednego komputerowego systemu informacyjnego, zdolnego do wspierania wszystkich tych obszarów, z ich indywidualnymi i specyficznymi wymaganiami. Każdy dział ma swój własny obszar roboczy zamknięty w odpowiedniej roli i w wielu przypadkach rozumiany jako podsystem informacyjny, zoptymalizowany zgodnie ze specyfiką prowadzonej działalności. System ERP łączy wszystkie cechy w ramach zintegrowanego

oprogramowania, które działa na centralnej bazie danych, dzięki czemu wszystkie organizacje mogą lepiej dzielić się informacjami. Co więcej, obecne wersje systemów ERP rozszerzają swoje działanie ponad granicami jednej organizacji [5]. W zależności od dostawcy, funkcjonalność systemu składa się z listy typowych komponentów (modułów) [6].



Rys. 2.1. Architektura modułowa systemu ERP [7]

Modułowa architektura systemu ERP obejmuje odrębne komponenty biznesowe, które są od siebie niezależne i mogą być wdrażane oddzielnie. Każdy moduł zajmuje się różnymi funkcjami określonego działu organizacji. Poszczególne systemy ERP mają swoją własną strukturę, jednak możemy wyróżnić następujące obszary jako moduły [7]:

- *Finanse (ang. Finance)* - moduł dla księgowości i finansów. Obejmuje rozliczenia, płatności i uzgadnianie kont, które mogą być wykonywane automatycznie. Ustawienia modułu muszą być zgodne z odpowiednimi przepisami obowiązującymi w danym kraju. Moduł przygotowuje również różne raporty, zwłaszcza zgodne z obowiązującymi wzorami (np. zestawienie zysków i strat).
- *Zarządzanie łańcuchem dostaw (ang. Supply Chain Management)* - moduł obejmuje wszystkie procesy, które przekształcają surowce w produkty końcowe. Obejmuje wiele

szczegółowych elementów w tym obszarze, jak przykładowo zaopatrzenie, przechowywanie i dostawa, itp.

- *Zarządzanie zapasami (ang. Inventory Management)* - moduł dostarcza informacji o bieżących zapasach i umożliwia przewidywanie przyszłego zapotrzebowania rynku. Może również obejmować narzędzia do efektywnego zarządzania przestrzenią magazynową w celu jej optymalnego wykorzystania.
- *Produkcja (ang. Manufacturing)* - moduł zarządza i optymalizuje zdolności produkcyjne, planowanie produkcji i harmonogramowanie produktów, pomaga w zarządzaniu zapasami i ilościami wytwarzanych produktów zgodnie z zapotrzebowaniem.
- *Zarządzanie relacjami z klientami CRM (ang. Customer Relationship Management)* - moduł organizuje wszystkie dane klientów, wysokowydajne kampanie i wydajną obsługę. Zaawansowane funkcje CRM zapewniają analizę danych i raporty umożliwiające ocenę zachowań klientów, wzorców zakupów i poziomów satysfakcji.
- *Zarządzanie zasobami ludzkimi (ang. Human Resources Management)* - moduł obsługuje wszelkie sprawy związanymi z pracownikami, takimi jak zatrudnianie, szkolenia, rozwój, listy płac, bezpieczeństwo, dobre samopoczucie, świadczenia, motywacja i administracja.
- *Analizy (ang. Business Intelligence)* - moduł obejmuje narzędzia do analizy bieżących i historycznych danych w celu wspierania bardziej efektywnego strategicznego, taktycznego i operacyjnego wglądu w sytuację organizacji i podejmowania decyzji. Zaawansowane funkcje zapewniają eksplorację danych, wizualizację danych i narzędzia danych, aby pomóc organizacjom w podejmowaniu lepszych decyzji bazujących na danych.
- *Zarządzanie serwisem (ang. Service Management)* - moduł wspiera procesy firmy usługowej. Koncentruje się na zespołach serwisowych, zarządzaniu ich czasem i produktywnością, planowaniu zadań i tras.
- *Zarządzanie Projektami (ang. Project Management)* - moduł zapewnia funkcjonalności, narzędzia i techniki wymagane do zarządzania i rozliczania projektów. Umożliwia codzienne monitorowanie wydatków i wykonanej pracy zgodnie z celami lub budżetami dla każdego projektu.

Moduły ERP automatyzują i wspierają znaczną część procesów biznesowych i administracyjnych. W związku z tym producenci standardowego oprogramowania ERP prezentują nie tylko zakres pokrycia systemem obszarów biznesowych, ale przedstawiają

również korzyści z wdrożenia z perspektywy procesów biznesowych End2End (E2E). Wielu z dostawców oferuje również wbudowane rozwiązania do integracji systemu z platformami zewnętrznymi poprzez tzw. szynę integracyjną. Tego typu narzędzia w połączeniu z systemowymi operacjami realizowanymi w czasie rzeczywistym i regularnymi aktualizacjami statusu zadań zewnętrznych oraz szybką realizacją wszystkich żądań (wejścia/wyjścia) do platform zewnętrznych wspierają ogromne możliwości optymalizacyjne na poziomie całej organizacji.

2.1.1. Definicja projektu wdrożeniowego

Wdrożenie systemu ERP jest unikalnym i ogromnym wyzwaniem, którego typowy czas realizacji wynosi od kilku miesięcy do nawet pięciu lat. Ze względu na skalę oraz obszar oddziaływania projekt wdrożenia systemu ERP nie powinien być postrzegany jako kolejny projekt działu IT, ale przedsięwzięcie całego zespołu pod opieką najwyższych władz organizacji. Jest bardzo istotne ponieważ realizowane przez system zmiany mają wpływ na wiele kluczowych aspektów organizacji, a ich akceptacja i rozumienie przez zespoły biznesowe sprzyja dynamice zmian na każdej płaszczyźnie projektu wdrożeniowego.

Typowy projekt wdrożenia ERP jest zorganizowany sekwencyjnie z etapami różniącymi się w zależności od wdrażanego systemu i przyjętej metodologii. Od początku istnienia systemów ERP opracowano różne podejścia i metodologie wdrożeniowe [8] [9] [10] [1] [11] [12] [13], które wraz z rozwojem systemów ERP są adaptowane do aktualnej rzeczywistości.

Systemy ERP muszą być wdrażane w taki sposób, aby zapewnić: wierne odzwierciedlenie i bezpieczne przetwarzanie informacji, integralność, spójność, niezawodność i użyteczność danych dla przedsiębiorstwa [2]. Wdrożenie systemu ERP jest realizowane w krokach zwanych fazami lub etapami kończącymi się odbiorem kolejnego kamienia milowego projektu. W skład każdej z faz wchodzi wszystkie działania projektowe [14], które powinny zostać wykonane przez zespół projektowy. Etapowe wdrażanie według modułów jest zdecydowanie najpopularniejszą techniką, w której komponenty ERP są wdrażane zgodnie z harmonogramem wg określonych priorytetów. Główne moduły ERP są zazwyczaj integrowane jako pierwsze, a następnie komponenty (mniej priorytetowe) realizowane są w kolejnych etapach [15]. Bardzo ważne, aby Faza I i II była realizowana wspólnie dla wszystkich obszarów, pozwoli to na uniknięcie błędów projektowych z perspektywy międzyobszarowej i zminimalizuje ewentualne zmiany w konfiguracji wcześniej wdrożonych modułów.

2.1.2. Znaczenie projektu wdrożeniowego dla przedsiębiorstw

Wdrożenie systemu ERP jest procesem kosztownym, długotrwałym, ryzykownym i pełnym złożonych czynników organizacyjnych, takich jak: początkowo nieznane wymagania,

poziom akceptacji użytkowników i szybko zmieniające się otoczenie IT. Doświadczenie wielu przedsiębiorstw jasno wykazuje, że nie jest realistycznym podejście do tych projektów, jako do projektów wyłącznie informatycznych (tylko z technicznego punktu widzenia). Są to projekty wpływające na firmę w wielu obszarach: procesów (reengineering procesów), struktury organizacyjnej, procedur, ról i zadań pracowników oraz samej kultury firmy. Wszystkie te implikacje mają głęboki wpływ na organizację, a w powiązaniu z zasobami finansowymi i ludzkimi zaangażowanymi w projekt determinują wiele powiązanych ze sobą ryzyk [16]. Wdrożenie systemu ERP dodatkowo znacznie zmniejszają możliwy margines błędów związanych z wprowadzaniem informacji do wielu systemów przez różnych pracowników w wielu różnych rodzajach baz danych [17].

Zmiana kultury organizacyjnej jest niezwykle potrzebna, a zarazem trudna do przeprowadzenia. Projekty ERP, to nie tylko wprowadzenie nowego oprogramowania do organizacji, ale także znacząca zmiana w zakresie przepływu informacji w ramach struktury organizacyjnej, co bezpośrednio wpływa na sposób pracy oraz interakcji między pracownikami. Niezwykle ważne dla przezwyciężenia tych trudności, jest zapewnienie pracownikom odpowiedniej, ukierunkowanej na ich obszar biznesowy opieki w ramach wdrożenia systemu. Jest to kluczowe dla zapewnienia wysokiego wskaźnika adaptacji nowo wdrożonego systemu ERP w firmie [18].

Organizacje, które osiągają pełne korzyści z wdrożenia nowych technologii to te, które dokonują niezbędnych zmian w swojej strukturze organizacyjnej, strategiach i procesach biznesowych, tak aby jak najlepiej wykorzystać możliwości systemu ERP [19]. Wdrożenie powinno koncentrować się na strategicznych celach biznesowych i uwzględniać złożoną integrację biznesową, a także wymagania z perspektywy organizacyjnej, technicznej i ludzkiej [14].

Systemy ERP są jedną z najbardziej kosztownych inicjatyw technologicznych, jakie może wdrożyć organizacja. Całkowity koszt posiadania systemu ERP obejmuje: licencje na standardowe oprogramowanie ERP, sprzęt (wydatki na środowisko serwerowe czy wymianę urządzeń peryferyjnych), usługi profesjonalne (doradztwo wdrożeniowe, bieżące utrzymanie/serwis, aktualizacje i optymalizację) oraz koszty wewnętrzne. Dlatego też kluczowe znaczenie ma określenie odpowiedniego budżetu i źródeł finansowania wdrożenia rozwiązania ERP, a także czasu trwania projektu i przewidywanego okresu zwrotu inwestycji [15].

Pomyślne wdrożenie systemu ERP to takie, które będzie realizowało zdefiniowane przed podjęciem projektu cele biznesowe i operacyjne. Wiele organizacji decyduje się na wdrożenie zintegrowanego systemu zarządzania, ponieważ poprzez optymalizację procesów i czynności operacyjnych może obniżyć koszty zapasów, produkcji, pracy i utrzymania systemów informatycznych IT oraz zapewnić lepszą przewagę konkurencyjną dzięki nowym funkcjom

[20]. System ERP może połączyć wszystkie obszary przedsiębiorstwa, w tym m. in. CRM - relacje z klientami, produkcję, zasoby ludzkie, zarządzanie finansami, sprzedaż (klienci) i zakupy (dostawcy), tworząc wysoce zintegrowany system ze współdzielonymi danymi. Potencjalne korzyści obejmują znaczny spadek zapasów, redukcję kapitału obrotowego, zebranie informacji na temat tego, czego klienci chcą i potrzebują oraz możliwością przeglądania i zarządzania analizą historyczną w zakresie klientów, dostawców, jako zintegrowaną całością.

Na podstawie doświadczeń projektowych, zarówno z wdrożeń zakończonych sukcesem, jak i porażką, można wskazać ogólne warunki udanego wdrożenia systemu ERP [19]. Zgodność między systemami ERP a kulturą organizacyjną jest warunkiem wstępnym udanego wdrożenia ERP. Często zespoły projektowe firm wdrażających systemy ERP nie rozumieją wymagań biznesowych, a tym samym korzyści, które systemy ERP mogą rozwiązać. Wiele organizacji utożsamia udane wdrożenie zintegrowanego systemu zarządzania z wyborem oprogramowania ERP, ale wskaźnik sukcesu projektów wdrożeniowych pokazuje, że od tego etapu do sukcesu jest jeszcze daleka droga [21]. Chociaż wybór oprogramowania powinien być bardzo staranny, oprogramowanie ERP nie powinno napędzać procesu podejmowania decyzji biznesowych. Wdrożenia ERP nigdy nie należy mylić z instalacją oprogramowania, która stanowi niewielki ułamek kosztów wdrożenia. Wdrożenie ERP wymaga dogłębnego strategicznego myślenia, które pozwala firmom lepiej zrozumieć własne procesy biznesowe. Dług technologiczny w zakresie IT i nacisk na infrastrukturę IT jest najmniej skorelowanym czynnikiem udanego wdrożenia ERP.

Pomyślne wdrożenie systemu ERP jest ściśle powiązane z zaangażowaniem kierownictwa najwyższego szczebla, które wyznacza strategiczny kierunek procesu wdrożenia. Ten czynnik koreluje najwyżej z wdrożeniem ERP spośród wszystkich czynników zidentyfikowanych w badaniu [18]. Istotne jest ciągłe wsparcie i monitorowanie procesu wdrażania, a brak lub niesystematyczne wsparcie ze strony najwyższego kierownictwa jest bardzo szkodliwe dla każdego wdrożenia ERP. Solidne i dogłębne zrozumienie zasad zarządzania projektem i jego zastosowania przez kluczowych członków zespołu jest krytycznie powiązane z udanym wdrożeniem ERP [21]. Zwiększenie szansy na powodzenie projektu osiąga się poprzez ustalenie racjonalnego zakresu projektu, ustalenie odpowiedniego zespołu projektowego z definicją ich obowiązków i miar do oceny efektów [10].

2.2. Charakterystyka typów projektów wdrożeniowych

Dostawcy systemów ERP opracowują podstawowe pakiety oprogramowania, które są aktualizowane okresowo (na przykład w zależności od zmieniających się wymogów prawnopodatkowych). Aktualizacje (tzw. łatki) mogą zawierać kilka drobnych

zmian/poprawek, tak aby system był zgodny z wymaganiami prawnymi i działał optymalnie. Cyklicznie (np. 2 razy w roku) dostawcy zapewniają zestaw zmian, które mają na celu rozwój i ewolucję funkcjonalności. Realizują ciągły rozwój interfejsów użytkownika i narzędzi ułatwiających budowę własnych rozwiązań w zakresie tworzenia pól, ekranów użytkownika, menu i raportów bez konieczności tworzenia kodu. Duże cykliczne aktualizacje zawierają nowe funkcje modułów lub narzędzia do obsługi kolejnych procesów biznesowych. Ze względu na znaczne nasycenie rynku ERP, dostawcy stale rozwijają oprogramowanie i dodają moduły, które starają się odpowiedzieć na oczekiwania klientów w zakresie przyszłych trendów w gospodarce. Eliminacja długu technologicznego, a co za tym idzie posiadanie aktualnego systemu ERP jest niezbędne zarówno dla małych i średnich przedsiębiorstw (MŚP), jak i dla globalnych organizacji. Jest to spowodowane ewolucją otoczenia rynkowego lub samej organizacji w zakresie rozwoju funkcjonalności systemu ERP do obsługi zmieniających się wymagań biznesowych. Co więcej, systemy ERP są podstawą dla Factory of the Future (FoF) [22]. FoF opiera się na solidnych podstawach w zakresie modularyzacji, masowej personalizacji, rozproszonej kontroli, IoT, modelowania i prognozowania, współpracy, rozproszonej produkcji i przejrzystych procesów.

Dodatkowo, zmianą w kilku ostatnich latach jest przenoszenie wdrożeń systemów ERP do chmury. Wynika to zarówno ze względu na potrzeby klientów oraz wymagań ciągle zmieniającego się rynku. Kluczowa zmiana z perspektywy klienta polega na tym, że organizacja wykupuje dostęp do wybranych usług i funkcjonalności systemu ERP. Jednym z głównych powodów takiego podejścia jest łatwiejszy sposób utrzymania, bycie na bieżąco z funkcjonalnościami i potencjalna gotowość do rozszerzenia ERP o pojawiające się potrzeby biznesowe. Jest to główna zaleta w porównaniu z rozwiązaniem on-premise i ERP zainstalowanym w chmurze prywatnej. Rozwiązanie on-premise jest często wybierane w przypadku znacznego dostosowania systemu ERP i integracji z innymi systemami w firmie. Tego typu projekty wdrożeniowe wiążą się z wysokim ryzykiem oraz długotrwałym procesem wdrożeniowym.

Wdrożenia systemów ERP można podzielić na różne typy projektów, które różnią się swoją specyfikacją, etapami i podejmowanymi działaniami. Odpowiedni typ projektu jest realizowany w zależności od sytuacji i potrzeb klienta (przedsiębiorstwa, instytucji, itp.), tak aby przygotowane rozwiązanie, które najlepiej wpasuje się w charakter jednostki i spełni wszystkie potrzebne funkcjonalności. Biorąc pod uwagę charakterystykę projektów wdrożeniowych, zakres, realizowane cele oraz wszystkie powyższe kwestie, zaproponowany został podział na następujące kategorie projektu:

- Implementacja
- Upgrade

- Roll’out

- Reimplementacja

W kolejnych podpunktach każdy z wyszczególnionych typów projektów zostanie szczegółowo opisany.

2.2.1. Implementacja

Implementacja to pierwszy cykl procesu wdrożenia systemu ERP w danej organizacji, której celem jest przekształcenie jej w bardziej wydajną i skuteczną organizację.

Decyzja o wdrożeniu systemu ERP często motywowana jest koniecznością zarządzania rozproszoną, złożoną i zróżnicowaną infrastrukturą organizacji [16]. Systemy ERP są aplikacjami modułowymi, które posiadają standardowy zestaw działań i konfiguracji systemu. Dostosowanie aplikacji jest możliwe przez wybór odpowiednich modułów, które zachowują integrację pomiędzy danymi w różnych modułach systemów [23], lub dodanie modyfikacji oprogramowania w celu dopasowania go do istniejących procesów biznesowych organizacji [24]. Aby zaproponować najlepsze rozwiązanie dla danej organizacji, niezbędne jest przeprowadzenie szczegółowej analizy istniejących procesów biznesowych i znalezienie sposobu na dostosowanie ich do systemu. Im większa jest personalizacja systemu, tym większe są jego koszty wdrożenia i późniejszego utrzymania. Dodatkowo może to zmniejszyć płynność komunikacji w porównaniu do standardowych rozwiązań [23]. Dlatego też, implementacja systemu ERP to niezwykle trudny i złożony proces [20].

Szczegóły procesów biznesowych, ich relacje z elementami danych oraz sposób, w jaki te relacje są powiązane w przedsiębiorstwie, muszą zostać zrozumiane i zmapowane do, w pełni zintegrowanego, systemu ERP [2]. Wymagania systemowe powinny koncentrować się na relacjach danych, zależnościach, regułach biznesowych, przepływach procesów i przepływach pracy. Główne rozbieżności z produktem i istniejącymi procesami biznesowymi powinny zostać udokumentowane i ocenione, tak aby można je było obejść przy minimalnych kosztach.

Aby wdrożenie ERP zakończyło się sukcesem, niezwykle ważna jest umiejętność adaptacji i dostosowanie się organizacji jej zespołu do zmian operacyjnych, które może nieść za sobą integrowany w przedsiębiorstwie systemem [25]. Projekt wdrożenia systemu ERP składa się zazwyczaj z następujących etapów:[15]:

- Etap 1 - odkrywania i planowania, który obejmuje badanie i wybór systemu, utworzenie zespołu projektowego i zdefiniowanie szczegółowych wymagań systemowych.

- Etap 2 - projektowania, który opiera się na szczegółowych wymaganiach i zrozumieniu bieżących przepływów pracy w celu opracowania szczegółowego projektu systemu ERP.

Obejmuje to projektowanie nowych, bardziej wydajnych przepływów pracy i innych procesów biznesowych, które wykorzystują system.

- Etap 3 - rozwoju, który obejmuje konfigurację i dostosowanie oprogramowania do obsługi *przeprojektowanych* procesów. Może to również obejmować opracowanie integracji z innymi istniejącymi aplikacjami biznesowymi organizacji, które nie zostaną zastąpione przez system ERP. W przypadku korzystania z lokalnego systemu ERP (on-premise) to także instalacja niezbędnego sprzętu i oprogramowania. Równolegle z rozwojem oprogramowania następuje opracowanie materiałów szkoleniowych, które pomogą użytkownikom dostosować się do nowego systemu. Planowana jest migracja danych, która obejmuje wyodrębnianie, przekształcanie i ładowanie danych z wielu systemów, z których każdy może używać różnych formatów i może przechowywać zduplikowane lub niespójne informacje.
- Etap 4 - testowania, który obejmuje wstępne testowanie określonych modułów i funkcji, testy pełnych możliwości systemu, a także testowanie zmigrowanych danych.
- Etap 5 - uruchomienia systemu.
- Etap 6 - wsparcia i aktualizacji systemu.

Przedstawiony podział etapów oraz zakresu realizowanych zadań różni się przykładowo w zależności od przyjętej metodyki, wdrażanego systemu czy rodzaju instalacji (on-premise/cloud).

Niektóre organizacje dążą do jednoczesnego wdrożenia wszystkich modułów systemu ERP, podczas gdy inne koncentrują się najpierw na określonych modułach lub procesach o wysokim priorytecie, a pozostałe integrują później w ramach kolejnych etapów.

2.2.2. Upgrade

Upgrade systemu ERP to proces dużej zmiany wynikający z wdrożenia nowej wersji już zainstalowanego systemu ERP, którego głównym celem jest dodanie aktualizacji funkcjonalnych i umożliwienie korzystania z nowych technologii i strategii biznesowych. Termin ten nie obejmuje drobnych zmian w ramach wersji systemu ERP, takich jak nowe zmiany, które zapewniają jedynie aktualizacje techniczne w zakresie poprawy błędów czy dostosowania do zmian prawnych. Ze względu na dużą złożoność systemów ERP, projekty upgrade mogą być przeprowadzane tylko w ramach kompleksowych projektów i wymagają znacznych zasobów osobowych i finansowych, a także wysokiego poziomu wiedzy w zakresie ERP. Zazwyczaj polecane jest przeprowadzenie, co najmniej co 2 lata, dużej aktualizacji ERP i kilku małych aktualizacji pomiędzy, aby zagwarantować płynne działanie systemu [24].

Podjęcie decyzji o projekcie upgrade'u systemu ERP może mieć różne przyczyny. Jedną z nich jest chęć korzystania z najnowszych technologii ERP i wykorzystania wszystkich możliwości systemu, a także wprowadzenie nowych funkcjonalności do systemu. Aktualizacje systemów ERP mają głównie na celu wykorzystanie nowych technologii i strategii biznesowych, aby zapewnić, że organizacja nadąża za najnowszymi trendami rozwoju biznesu. W związku z tym decyzja o aktualizacji systemu ERP, nie jest zwykle spowodowana pogorszeniem jakości kodu lub przewidywaną redukcją kosztów utrzymania, ale innymi celami. Aktualizacje ERP zapewniają organizacjom przyszłe ulepszenia od dostawców, aby dać przedsiębiorstwom lepsze możliwości nadążania za bieżącym rozwojem biznesu, ulepszania ich procesów i budowania bardziej wydajnych modeli biznesowych dzięki nowym funkcjom i nowym stylom przetwarzania dostarczonym w zaktualizowanych wersjach ERP [18]. Innym powodem może być brak kontynuacji wsparcia dla posiadanej wersji systemu lub przestarzałość posiadanych technologii i trudność w znalezieniu specjalistów do utrzymywania systemu [20]. Kolejną czynnikiem, jest czynnik ekonomiczny, czyli przykładowo fuzja lub przejęcie firmy, albo wprowadzenie nowych regulacji prawnych, a także zmniejszenie kosztów utrzymania systemu. Nowe aktualizacje umożliwiają kompatybilność z innymi nowoczesnymi narzędziami. Szczególnie ważne może być też pragnienie wydajności i odpowiedniego wykorzystania zasobów lub rosnąca presja na efektywność w celu wzmocnienia konkurencyjności, a nawet konsolidacja systemów pochodzących od różnych producentów [26].

W celu przeprowadzenia udanego wdrożenia upgrade'u systemu ERP, bardzo ważne są takie czynniki jak: jasne określenie celu, dobra koordynacja i komunikacja z partnerem wdrożeniowym, dokładne zarządzanie historią dostosowywania, przygotowanie do wymagań systemowych oraz redefinicja ról między działami biznesowymi i IT [27]. Dodatkowo należy rozważyć, gdzie możliwa jest standaryzacja systemu i poprawienie procesów biznesowych z wcześniejszej implementacji. Proces musi być starannie zaplanowany przez Kierownika Projektu i zgodnie z założeniami realizowany. Przed przeprowadzeniem upgrade'u systemu niezwykle ważne jest przeanalizowanie różnic w zakresie obecnych i możliwych nowych funkcjonalności oraz podjęcie decyzji, które z nich są używane, a także czy jest możliwość zmiany indywidualnych rozwiązań na te standardowe. Niezwykle ważne jest zebranie wszystkich wymagań w sposób szczegółowy i wyciągnięcie z nich wniosków, tak, aby nie przenosić wcześniejszych błędów do kolejnych wersji systemu. Upgrade systemu to proces niezwykle skomplikowany i w trakcie wdrożenia go można napotkać wiele trudności. Pierwszą z nich jest przeniesienie wcześniejszych modyfikacji kodu i wcześniejszych dostosowań systemu do nowej wersji. Szczególnie trudne i nie zawsze możliwe jest wtedy zapewnienie tych samych funkcjonalności oraz integracji pomiędzy różnymi danymi. Należy także pamiętać o tym, aby nie edytować części kodu źródłowego szablonu, co może spowodować

dalsze komplikacje [24]. Innym problemem może być brak dokumentacji dotyczącej dostosowań i raportów. Utrudnia to proces aktualizacji ze względu na brak potrzebnych danych i potrzebę przeprowadzenia pewnych części analiz jeszcze raz [27]. Ważne jest też zapoznanie użytkowników z nowym systemem i zapewnienie im ciągłości w zakresie wiedzy i kompletności informacji w ramach wykonywanych zadań. Dodatkowo trzeba także rozważyć jaką strategię upgrade'u systemu wykorzystać w zależności od wymagań organizacji, czy skoncentrować się na minimalizacji czasu przestoju, czy maksymalizację dostępnych zasobów [28]. Strategia minimalizacji czasu skutkuje krótszym czasem przestoju i zmniejsza ryzyko w zakresie działania nowych procesów, ale ma większe zapotrzebowanie na zasoby systemowe do równoległego działania zarówno systemu produkcyjnego, jak i systemu równoległego. Strategia maksymalizacji zasobów oznacza, że podczas aktualizacji nie są potrzebne dodatkowe zasoby systemowe, ale skutkuje dłuższym czasem przestoju, a po aktualizacji wymagana jest większa skala testów dodatkowych. Wraz z rozwojem organizacji bardzo ważny jest też jej rozwój technologiczny, co oznacza, że w pewnym momencie system zostanie zastąpiony nową wersją lub innym systemem [29]. Jeżeli aktualizacje systemu będą przeprowadzone w odpowiednim czasie, to może to przynieść większe zyski i lepszą wydajność organizacji. System ERP nie jest narzędziem, które przestanie być używane, ale będzie ewoluowało wraz z organizacją i jest nieuniknioną częścią zmiany podczas tej ewolucji.

2.2.3. Rollout

To proces wdrożenia systemu ERP na podstawie wcześniejszej udanej implementacji projektu w innej lokalizacji firmy lub kolejnej spółce. Najczęściej ten scenariusz jest wykorzystywany w dużych przedsiębiorstwach, grupach kapitałowych lub międzynarodowych korporacjach, które posiadają swoje siedziby w wielu lokalizacjach i różnych krajach.

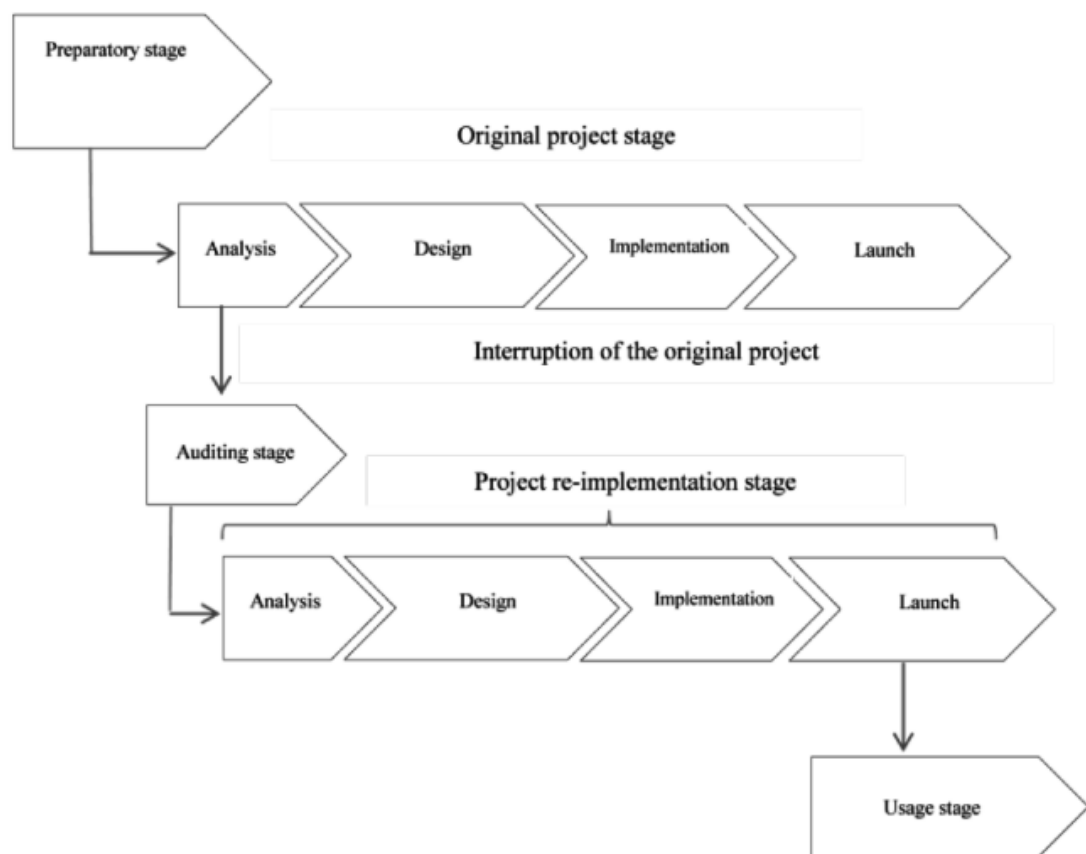
Zwykle proces wdrożeniowy odbywa się według uzgodnionego scenariusza w kolejnych lokalizacjach, a spółki zakładane są na podstawie szablonu lub planu przygotowanego przez główną siedzibę organizacji. Przygotowany wcześniej przez centralną spółkę decyzyjną zakres i wytyczne definiują możliwości potencjalnych zmian i adaptacji względem rozwiązania grupowego, co może pomóc lokalnemu oddziałowi/spółce skutecznie i wydajnie wdrożyć nowy system [5]. Jednakże standaryzacja procesów biznesowych może być stosunkowo trudna dla poszczególnych lokalizacji organizacji, które mają wypracowane własne rozwiązania i metody [23]. Często mają one sporą autonomię, fundusze, uprawnienia do zatrudniania zasobów w całej jednostce biznesowej i ponoszą bezpośrednią odpowiedzialność przed swoimi interesariuszami. Dlatego też niektóre firmy tworzą zespół, który podróżuje między każdą jednostką biznesową i lokalizacją w celu wsparcia procesu wdrożenia. W miarę jak zespół zdobywa większe doświadczenie przy każdym wdrożeniu, kolejne wdrożenia stają się

bardziej wydajne. Ta metoda osiąga lepszy poziom akceptacji przez pracowników i pomimo dłuższego czasu trwania procesu, ryzyko jest niższe. Zdarza się, że ze względu na różnice w przepisach lub lokalizacji geograficznej, wymagane są dostosowania wprowadzane w części rozwiązań specyficznych dla danej siedziby [15].

Prawidłowo przeprowadzony proces wdrożenia rolloutów w całej organizacji znacząco zwiększa jej potencjał i wartość. Ujednoczenie daje możliwość zarządzania całością danych firmy, skrócenie czasu realizacji procesów pomiędzy spółkami, obniżenie kosztów w łańcuchach dostaw, zmniejszenie poziomu zapasów, poprawa terminowości dostaw i widoczności asortymentu produktów w całej organizacji.

2.2.4. Reimplementacja

To proces powtórnego wdrożenia systemu ERP w firmie ze względu na kompletnie lub częściowo nieudaną pierwszą implementację (Rys.2.2). Decyzja o przeprowadzeniu reimplementacji jest najczęściej podejmowana w momencie, w którym na drodze do implementacji pojawiają się nierozwiązywalne problemy i porażki lub koszt kontynuacji implementacji jest o wiele większy niż przeprowadzenie procesu od początku. W momencie podjęcia decyzji, o przerwaniu procesu implementacji w celu minimalizacji całości kosztów wdrożenia systemu ERP, niezwykle ważne jest przeprowadzenie audytu. Realizacja przeglądu ma na celu zidentyfikowanie przyczyn zakończenia projektu i jego niepowodzeń, a także rekomendację i zalecenia dotyczące ponownego wdrożenia. Następnie wdrożenie jest przeprowadzane jeszcze raz, przez nowy zespół wdrożeniowy, który realizuje je zgodnie z założeniami audytu [30]. Jeżeli pojawiają się przesłanki o możliwej potrzebie reimplementacji projektu, to dobrym momentem na przeprowadzenie audytu i podjęcie decyzji o reimplementacji jest koniec fazy analizy wymagań biznesowych klienta.



Rys. 2.2. Porównanie projektu implementacyjnego i reimplementacyjnego [30]

3. Przegląd metodyk projektów informatycznych

W rozdziale przedstawiony został przegląd i charakterystyka wybranych metodyk projektowych systemów informatycznych. Użycie metodyk w trakcie realizowania projektu ma na celu strukturyzację i optymalizację procesu poprzez wytworzenie powtarzalnych schematów działania, a także optymalizację zarządzania różnymi aspektami projektu takimi jak optymalizacja zasobów, kontrola harmonogramu i budżetu, nadzór jakości oraz zmiany w projekcie.

Każdy projekt ma indywidualne ramy, takie jak: zakres i jakość rozwiązania, czas trwania projektu, zasoby i ryzyka [31] [32], dlatego powinien być traktowany jako nowe przedsięwzięcie z odrębnymi aspektami do rozważenia w ramach jego realizacji [21]. Każdy projekt jest wyjątkowy, a bardzo ważną kompetencją kierowników projektów jest elastyczne dostosowywanie się i reagowanie na różne wyzwania, które pojawiają się w codziennej pracy związanej z zarządzaniem projektami [33]. Pod uwagę powinny być brane takie aspekty jak: charakterystyka branży w której działa klient, a także systemy, narzędzia i struktury używane do realizacji projektu [34]. Projekt zakończony sukcesem to taki, który został sfinalizowany w przewidzianym koszcie, czasie i zakresie, a także z ogólną satysfakcją użytkowników końcowych z działania systemu i odwzorowania w nim procesów biznesowych organizacji [31]. Z tego powodu, wybór odpowiedniej metodyki projektowej do projektu ma kluczowe znaczenie dla pomyślnego zakończenia procesu [31] [35] [21][34]. Metodyki projektowe można podzielić na klasyczne i zwinne [36], przy czym główna różnica polega na sposobie planowania pracy w projekcie. W metodykach klasycznych, praca i zakres jest zaplanowany na początku realizacji projektu i następnie wykonywana zgodnie z założeniami planu. Natomiast w metodyce zwinnej stosowane jest podejście iteracyjne, a projekt ewoluuje zgodnie ze zmieniającymi się wymaganiami.

3.1. Model kaskadowy

Model kaskadowy (Waterfall) jest uznawany za jedną z klasycznych metodyk projektowych dla procesów wytwarzania oprogramowania. Metodyka ta została zaprezentowana w 1970 roku przez Winstona W. Royce'a. Zakłada ona liniową realizację projektu poprzez podział projektu

na poszczególne fazy lub etapy i następnie wykonywanie ich zgodnie z ustaloną kolejnością [36][35]. Wszystkie wymagania muszą być znane przed rozpoczęciem prac, a każdy z kolejnych etapów może być rozpoczęty tylko po zakończeniu wcześniejszego [37].

Metodyka Waterfall składa się z 5 faz (Rys.3.1), takich jak [35] [37] [38]:

1. **Analiza biznesowa:** Celem jest zdefiniowanie wymagań funkcjonalnych i нефункциональных systemu. Wymagania funkcjonalne definiują cel, zakres i konkretne działania systemu, możliwości użytkownika, cechy oprogramowania i wymagania bazy danych. Wymagania нефункциональные opisują takie wartości jak niezawodność, skalowalność, wydajność i jakość dostarczania usług. Faza kończy się utworzeniem dokumentów z założeniami wstępnymi i specyfikacją wymagań systemowych.
2. **Projektowanie:** Ta faza ma na celu zaprojektowanie rozwiązania na podstawie wcześniejszych wymagań. Obejmuje ona wybór odpowiedniej architektury oprogramowania, struktur danych w bazy danych, a także zaprojektowanie funkcjonalności systemu.
3. **Implementacja:** W tym etapie powstaje rozwiązanie techniczne (np. aplikacja, system) na podstawie wcześniej zaprojektowanych rozwiązań poprzez wytworzenie kodu aplikacji.
4. **Testowanie:** Obejmuje testowanie aplikacji utworzonej w fazie implementacji. Testy mogą obejmować różne typy wymagań, zarówno sprawdzanie, czy aplikacja działa poprawnie jak i na podstawie wymagań biznesowych, ale także testy wydajności. Znalezione błędy są poprawiane, tak aby zapewnić poprawne działanie aplikacji według wymagań.
5. **Serwisowanie:** Następuje po oddaniu aplikacji do użycia, obejmuje kontrolę nad odpowiednim działaniem aplikacji, naprawę błędów, poprawę działania przy pomocy modyfikowania rozwiązania w celu zwiększenia efektywności działania systemu.



Rys. 3.1. Model kaskadowy

Największą zaletą wykorzystania metodyki kaskadowej w projekcie jest korzystanie ze struktury projektowej, która odpowiada za organizację i kontrolę projektu. Przejrzysta

struktura projektowa ułatwia podział zadań i obowiązków poprzez departamentalizację i usprawnienie kontroli nad przebiegiem projektu [35]. Kolejną wartością dodaną jest utworzenie pełnej dokumentacji projektowej na samym początku procesu powstawania projektu, co wpływa pozytywnie na zwiększenie stopnia wiedzy klienta na temat systemu, ułatwia dołączenie nowej osoby do projektu, a także ułatwia dalszy rozwój i serwisowanie po zakończeniu realizacji projektu. W przypadku realizacji projektu, zgodnie z wcześniej ustalonymi założeniami, możliwe jest dokładne przewidzenie faktycznego kosztu przeprowadzenia projektu, a także czasu jego zakończenia. Metoda kaskadowa jest idealnym rozwiązaniem w przypadku projektów o jasnej wizji produktu końcowego, jak i pewności, o braku zmian w wymaganiach projektowych.

Ze względu na swoją charakterystykę użycie metodyki kaskadowej w procesie realizacji projektu posiada pewne mankamenty. Definicja wszystkich wymagań funkcjonalnych na początku projektu często jest trudna w rzeczywistych sytuacjach ze względu na możliwe zmiany wymagań w trakcie procesu wytwarzania rozwiązania [39]. Kolejną trudnością jest brak uwzględnienia w ramach projektowych ryzyk i niepewności, które często występują w praktycznych operacjach. Brak uwzględnienia ryzyk w projekcie jeszcze bardziej podwyższa ich poziom [35].

3.2. Agile

Metodyka Agile została utworzona jako odpowiedź na wcześniejsze nieelastyczne metodyki projektowe, które nie potrafiły sprostać dynamicznie zmieniającym się wymaganiom w trakcie procesu wytwarzania oprogramowania [40] [41]. Zasadniczym celem używania metodyki Agile jest możliwość adaptacji do często zmieniających się czynników zewnętrznych podczas procesu realizacji projektu [35]. Na początku projektu wyznaczana jest tylko ogólna wizja produktu, jednakże szczegółowe wymagania funkcjonalne są rozwiązywane krok po kroku. Wyznaczenie jakościowych celów i wizji projektu jest podstawą powodzenia projektów prowadzonych metodyką Agile [42]. Najlepsze rozwiązanie jest tworzone poprzez bardzo dobrą komunikację zarówno pomiędzy członkami zespołu jak i z klientem, wspólną współpracę i dobrą samodzielną organizację pracy [43]. Wymaga to uczestniczenia w projekcie wykwalifikowanych programistów, którzy chcą pracować jako zespół, komunikować się, zarządzać ciągłymi zmianami i rozwiązywać zaistniałe problemy. W związku z tym zespół jest kluczem do sukcesu zwinnego tworzenia oprogramowania, ponieważ każda zwinna metoda tworzenia oprogramowania w znacznym stopniu opiera się na współpracy i komunikacji [44].

Koncepcja wytwarzania oprogramowania Agile została zebrana w 2001 roku przez czołowych ekspertów technologicznych w dokumencie *Agile Manifesto*. Manifest składa się z 4 wartości i 12 zasad [45], które zostały przedstawione poniżej.

Wartości [46] [42]:

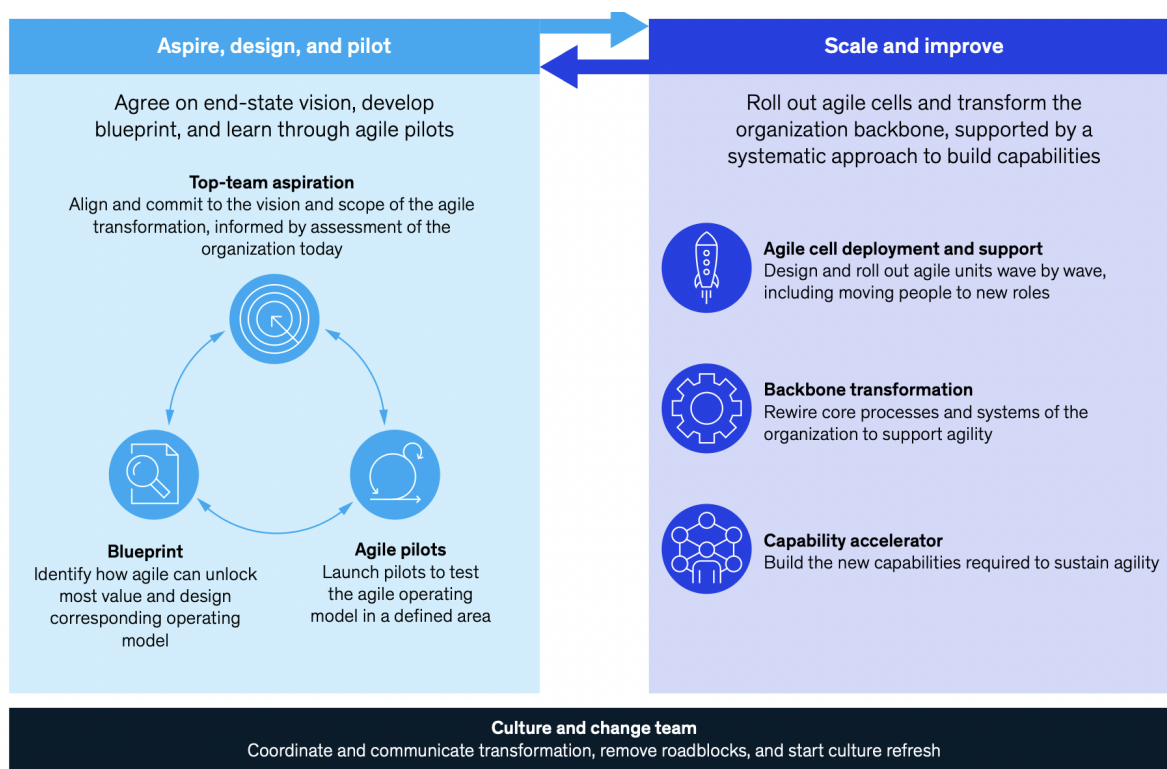
1. Osoby i interakcje ponad procesami i narzędziami.
2. Działające oprogramowanie jest ważniejsze niż obszerna dokumentacja.
3. Współpraca z klientem ponad zapisami i negocjacjami umowy.
4. Odpowiedź na zmianę zamiast kontynuowania planu.

Wprowadzone zostały również następujące zasady [46]:

1. Naszym najwyższym priorytetem jest zadowolenie klienta poprzez wczesne i ciągle dostarczanie wartościowego oprogramowania.
2. Zmieniające się wymagania są mile widziane, nawet na późnym etapie rozwoju.
3. Częste dostarczanie działającego oprogramowania, od kilku tygodni do kilku miesięcy, z preferencją krótszych terminów.
4. Członkowie zespołu ze strony biznesu i technicznej muszą codziennie współpracować podczas całego projektu.
5. Budowa projektu wokół zmotywowanych osób. Zapewnienie im środowiska i wsparcia, którego potrzebują, a także zaufanie, że wykonają swoją pracę.
6. Najbardziej wydajną i skuteczną metodą przekazywania informacji zespołowi ds. rozwoju jest komunikacja bezpośrednia.
7. Działające oprogramowanie jest główną miarą postępu.
8. Zwinne procesy promują zrównoważony rozwój.
9. Ciągła dbałość o doskonałość techniczną i dobry projekt.
10. Prostota - sztuka maksymalizacji ilości niewykonanej pracy.
11. Najlepsze architektury, wymagania i projekty powstają w samo-organizujących się zespołach.
12. W regularnych odstępach czasu zespół zastanawia się nad tym, jak stać się bardziej efektywnym, a następnie odpowiednio dostosowuje swoje zachowanie.

Realizacja zadań w metodyce Agile jest przeprowadzana w sposób iteracyjny, przez co wymagania i rozwiązania ewoluują w procesie rozwoju rozwiązania. Projekty są dostarczane w procesie seryjnym, a nie w momencie ich ukończenia [47]. Dobry zespół projektowy Agile, który jest gotowy do rozpoczęcia pracy w projekcie, powinien charakteryzować się następującymi cechami [48]:

- koncentracja na ważnej okazji biznesowej z dużą stawką stawką,
- odpowiedzialność za konkretne wyniki,
- zaufanie do autonomicznej pracy - kierowanie przez jasne prawa decyzyjne, dysponowanie odpowiednimi zasobami i składający się z niewielkiej grupy multidyscyplinarnych ekspertów,
- zaangażowanie w stosowanie zwinnych wartości, zasad i praktyk,
- upoważnienie do ścisłej współpracy z klientami,
- wsparcie przez kierownictwo wyższego szczebla, które zajmie się rozwiązywaniem przeszkód i napędzaniem pracy zespołu.

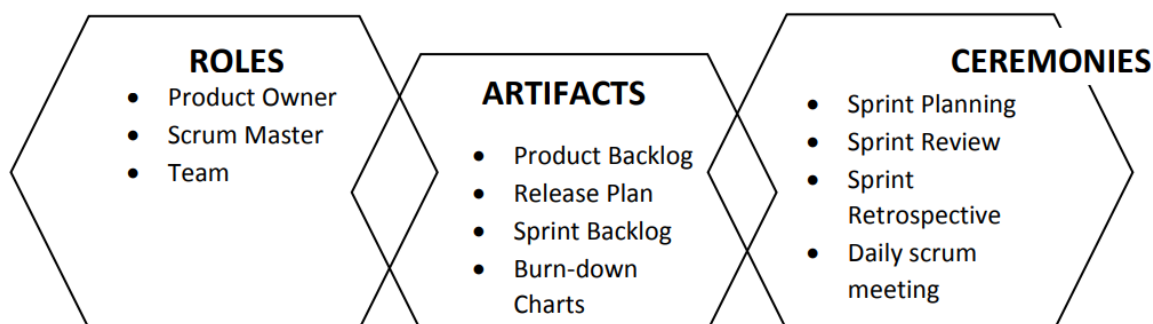


Rys. 3.2. Agile jako metodyka pracy w organizacji [49]

Zalety stosowania technologii Agile to między innymi zwiększenie pozytywnego odbioru projektu przez klienta poprzez stałe zaangażowanie i wsparcia z jego strony [39]. Kolejnym argumentem przemawiającym za używaniem tej metodyki, jest doprowadzenie projektu do lepszego wyniku, poprzez niezamykanie się w sztywnych ramach projektu, a postawienie na rozwój produktu w czasie procesu. Sukces projektu, w którym używana jest metodyka Agile, jest zależny od stałego i konsekwentnego udziału klienta w projekcie [39]. Jeżeli metodyka Agile jest prawidłowo wdrożona, to prawie zawsze skutkuje to wyższą produktywnością i poprawą morale zespołu, szybszym czasem wprowadzenia produktu na rynek, lepszą jakością i niższym ryzykiem niż w przypadku tradycyjnych metod [48]. Ryzykami związanymi z prowadzeniem projektu metodyką Agile mogą być też zmniejszone ograniczenia takich aspektów projektowych jak harmonogram i zakres, co może skutkować straceniem z widoku głównej wizji projektu. Pomimo tego, obecnie podejmowane są próby wprowadzenia metodyki Agile nie tylko jako sposobu prowadzenia projektu, ale skalowanie jej na całą organizację w celu nie tylko rozwiązania obecnych problemów, ale również osiągnięcia przewagi nad konkurencją w przyszłości [48] [49].

3.3. Scrum

To metodyka zwinna, która jest używana w projektach realizujących budowę i rozwój oprogramowania. Główną zasadą metodyki Scrum jest świadomość zależności, według której wraz z rozwojem produktu, klienci mogą zmienić zdanie na temat ich potrzeb i wymagań [50]. Strategia Scrum polega na elastycznym, holistycznym rozwoju produktu, w którym zespół programistów pracuje jako jednostka, aby osiągnąć wspólne cele [51]. Scrum składa się z zespołów, wydarzeń, artefaktów i zasad (Rys.3.3). Reguły są niezbędne do powiązania zespołów, wydarzeń i artefaktów podczas projektu. Zapewniają one również uzgodnioną strukturę rozwiązywania konfliktów w ramach projektu [52]. Wszystkie zasady zapisane są w *Scrum Guide* [53] i można podsumować je w następujący sposób. Praca jest realizowana w



Rys. 3.3. Scrum Framework [50]



Figure 4 Scrum Method

Rys. 3.4. Realizacja projektu z użyciem metodyki Scrum [35]

sposób iteracyjny i przyrostowy, poprzez stopniowe dostarczanie nowych fragmentów oprogramowania (Rys.3.4). Nowa funkcjonalność projektu jest realizowana w trakcie 2-4 tygodniowych okresów czasowych zwanych sprintami [50]. Metodyka Scrum charakteryzuje się silną komunikacją zarówno pomiędzy członkami zespołu projektowego jak i z klientami i docelowymi użytkownikami końcowymi, co umożliwi szybką zmianę w wytwarzanym oprogramowaniu na podstawie zmian w wymaganiach i celach biznesowych produktu. Początkowo wymagania dotyczące funkcjonalności produktu zostają zebrane w *Product Backlog* jako lista znanych wymagań i problemów do rozwiązania wraz z ich priorytetami [53]. Każdy okres sprintu jest poprzedzony spotkaniem *Sprint planning*, które ma na celu utworzenie listy zadań do rozwiązania w trakcie wyznaczonego okresu czasu. Dodatkowo po zakończeniu okresu organizowane jest spotkanie podsumowujące, podczas którego omawiane jest co udało się osiągnąć w czasie tego okresu i co można poprawić w kolejnych iteracjach procesu [46]. W ramach sprintu, codziennie organizowane jest spotkanie, podczas którego członkowie zespołu informują o tym, co udało się zrobić od ostatniego spotkania i co należy zrobić przed kolejnym spotkaniem [41]. Umożliwia to lepszą synchronizację działań pomiędzy członkami zespołu i identyfikację działań o najwyższym w danym okresie priorytecie.

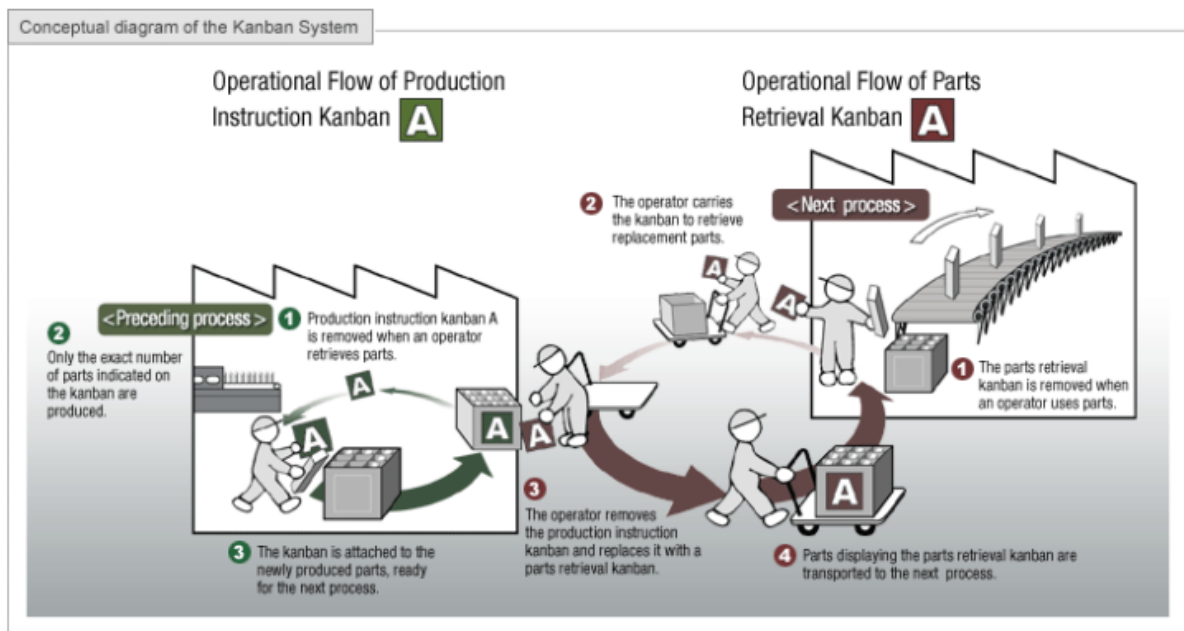
Za poprawne przeprowadzenie projektu zgodnie z metodyką Scrum odpowiada specjalnie wyznaczona za to osoba - Scrum Master. Jego rolą jest wsparcie członków zespołu w pracy zgodnie z zasadami metodyki, mierzenie procesu realizacji projektu, zarządzanie ryzykami i zakończeniu wyznaczonych zadań w wyznaczonych terminach. Od strony technicznej, projektem zarządza Product Owner, którego zadaniem jest identyfikacja wymagań, ich priorytetyzacja, a także podział na terminy i zadania do realizacji w danych okresach [53].

Metodyka Scrum narzuca również wielkość grupy realizującej projekt jak i jej skład. Standardowo liczy około 5-9 członków zespołów, w tym 3-4 deweloperów, tester, tech leader, Product Owner i Scrum Master[35]. Zaletami stosowania takiego trybu pracy są: widoczny od razu wynik pracy, zmniejszenie pracy nad wypracowaniem szczegółowych wymagań w specyfikacjach i podział zadań ze względu na umiejętności członków zespołu. Przykładowo programiści zajmują się tylko wytwarzaniem oprogramowania, a za komunikację z klientem odpowiedzialny jest Product Owner, który przekazuje programistom wypracowane z klientem ustalenia. Ze względu na fakt, że Scrum został zaprojektowany z myślą o małych zespołach, których członkowie są ze sobą w stanie ciągłej współpracy, ryzyka lub motywacja do zmiany metodyki mogą pojawić się w przypadku, gdy zespół projektowy pracuje z wielu lokalizacji i stref czasowych, komunikacja nie jest na najwyższym poziomie lub użycia metodyki dla większych projektów niż przewidziana wielkość zespołu [41]. Współczynnikiem zwiększającym pozytywną współpracę i komunikację jest dojrzałość zespołu, wyspecjalizowanie członków zespołu, mała rotacja i wcześniejsza praca według metodyki Scrum [44] [54].

Scrum nie definiuje formalnego procesu zarządzania ryzykiem ani nawet właściciela ryzyka. W celu rejestracji ryzyka można użyć na przykład prostego rejestru ryzyka lub risk burn-down chart [55].

3.4. Kanban

To metodyka zwinna, która powstała na podstawie procesów optymalizacji produkcji JIT (Just-In-Time) w firmie Toyota w latach 50 (Rys.3.5). Założenie JIT oznacza produkcję tylko tych produktów, które określono w danym momencie i sprecyzowania ich liczby [40]. Kanban zakłada realizację procesów biorąc pod uwagę takie aspekty jak: wizualizacja przepływu pracy, ograniczenia pracy w toku, zarządzanie przepływem pracy, uczynienia każdego kroku jednoznacznym i określenie ewolucji jako jednego, precyzyjnego mechanizmu [32] [56] [57]. Postępowanie zgodnie z takimi założeniami powoduje zmniejszenie ryzyk projektowych poprzez eliminację *wąskich gardeł* w procesie oraz zwiększenie jego wydajności i efektywności [47]. Ze względu na wysoką skuteczność projektów produkcyjnych



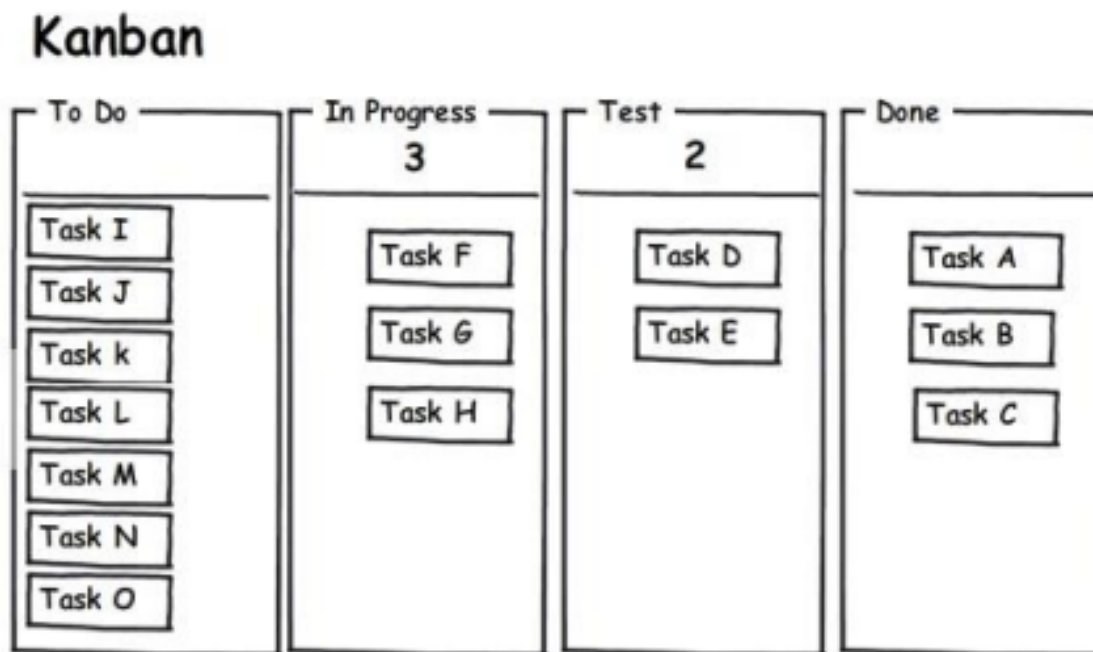
Rys. 3.5. Proces kanban [40]

zakończonych sukcesem przy użyciu założeń Kanban, przeprowadzono działania dostosowujące do stworzenia nowej metodyki projektowej [38].

Informacje dotyczące projektu są przechowywane w 4 płaszczyznach:

- Ideas - możliwe zadania do rozważenia, czy będą realizowane w ramach projektu
- To do - lista zadań do wykonania, które nie mają przydzielonych zasobów do ich realizacji
- In progress - lista zadań, które zostały już przydzielone i są w trakcie realizacji
- Done - lista zakończonych działań, które nie potrzebują już żadnych nakładów pracy do ich wykonania

Aby dobrze zwizualizować wszystkie wymiary projektu i jego zadania, wykorzystuje się do tego tablicę Kanban, która jest ogólnodostępna dla wszystkich członków projektu. Zazwyczaj tablica Kanban jest podzielona pionowo na różne kolumny lub etapy (Rys.3.6). Każdy etap odnosi się do stanu zadania, podczas gdy każde zadanie jest reprezentowane przez kartę przymocowaną do tablicy na etapie, aby reprezentować bieżący stan zadań. Na tablicy Kanban karty są przesuwane z lewej na prawą stronę w oparciu o zmiany stanu zadania [47]. Wizualizowany jest proces rozwoju, a także status postępu prac. Na początku stosowana była tablica analogowa, do której przyczepiano karty z zadaniami. Wraz z ogólnym rozwojem technologii, nastąpiła także automatyzacja tablicy Kanban [58]. Dane mogą być aktualizowane w czasie rzeczywistym, tablica Kanban może być też zintegrowana z innymi systemami wspomagającymi pracę przedsiębiorstwa, w tym między innymi systemem ERP.



Rys. 3.6. Tablica kanban [40]

Zastosowanie metodyki Kanban wpływa pozytywnie na komunikację w zespole, bardziej zrozumiałą wizualizację i zrozumienie przepływu procesów, zwiększenie produktywności i transparentności w czasie trwania projektu, co pozytywnie przekłada się na jego realizację. Kolejnymi zaletami są lepsza priorytetyzacja produktów i zadań, zmniejszenie liczby defektów i błędów, poprawa jakości i szybsze wprowadzanie produktów na rynek [59]. Możliwą trudnością w stosowaniu metodyki Kanban jest brak definicji takich wymiarów do zapisywania i późniejszego kontrolowania jak wartości, praktyki, techniki i narzędzia [56].

3.5. PRINCE2

PRINCE2 (Projects IN Controlled Environments, wersja 2) to standard metodyki zarządzania projektami, który w początkowej fazie był wykorzystywany przede wszystkim w Wielkiej Brytanii i Europie. Standard ten jest wymaganą metodą dla wszystkich projektów zleczanych przez rząd Wielkiej Brytanii. PRINCE2 znajduje się w domenie publicznej i oferuje wskazówki dotyczące najlepszych praktyk w zakresie zarządzania projektami. PRINCE2 został opracowany przez Centralną Agencję Komputerową i Telekomunikacyjną (CCTA) rządu brytyjskiego dla określonych projektów IT, ale analiza przeprowadzona w 1996 r. rozszerzyła zakres jego ważności dla każdego rodzaju projektu [60].

PRINCE2 można łatwo dostosować do różnych projektów i środowisk. Jest to jasno zdefiniowana struktura z rolami, obowiązkami, zasadami i procesami, które są łatwe do

nauczenia i zastosowania. PRINCE2 jest otwarty, elastyczny, a jego wykorzystanie szybko rośnie niezależnie od sfery gospodarczej czy lokalizacji geograficznej. W niektórych krajach posiadanie certyfikatu PRINCE2 jest niemal obowiązkowe, często ze względu na wymagania sektora publicznego [61].

PRINCE2 to tradycyjna metoda zwana „Process driven”, która odpowiada na pytania „co” i „dlaczego”, a w niewielkich ilościach na pytanie „jak” [32]. Opiera się ona na 4 filarach: Zasady (Principles) - przedstawione na rysunku Rys.3.7, Tematy (Themes), Procesy (Processes) i Adaptacja (Adaptation). Zasady są podstawowym czynnikiem gwarantującym



Source: Bentley, C. (2010). The Essence of the Project Management Method. England: INBOX SK.

Rys. 3.7. Wartości w metodyce Prince2 [62]

zarządzanie projektem, a tematy skupiają się na filozoficznych aspektach projektu, realizowanych poprzez procesy od początku do końca projektu. Zasady PRINCE2 to [62] [63]:

- ciągłe uzasadnienie biznesowe, uczenie się na podstawie doświadczeń,
- zdefiniowane role i obowiązki,
- zarządzanie etapami,

- zarządzanie przez wyjątki, skupienie się na produktach i dostosowanie do środowiska projektu.

Projekt PRINCE2 musi mieć ciągłe uzasadnienie biznesowe, aby być rozpoczęty i kontynuowany.

Projekt prowadzony tą metodyką składa się z następujących procesów [62]:

1. **Przygotowanie projektu** - ma na celu zebranie niezbędnych informacji do realizacji projektu, wyznaczenie kierownika i zespołu zarządzającego, zaplanowanie prac niezbędnych do przygotowania umowy między klientem a dostawcą, utworzeniu dziennika kierownika projektu.
2. **Inicjowanie projektu** - przygotowanie szczegółowych planów i strategii projektowych, a także dokumentacji, w tym dokumentu Project Initiation Documentation zawierającego najważniejsze ustalenia.
3. **Kierowanie projektem** - działanie kierownika projektu polegające na wdrażaniu przygotowania i zatwierdzenia planu projektu, zatwierdzenia jego realizacji, ustalenie standardów wykonywanych zadań i kontrolę ich wykonalności, monitorowanie postępów i pomyślne zakończenie projektu.
4. **Sterowanie etapem** - obejmuje stałe działania projektu, które monitorują postępy projektu i poszczególnych prac. Mogą to być takie czynności jak zatwierdzanie zadań, przegląd informacji, raportowanie i monitorowanie ryzyka, a także działania naprawcze.
5. **Zarządzanie dostarczaniem produktów** - obejmuje realizację prac projektowych poprzez planowanie, uzgadnianie i przydzielanie określonych zadań, ich raportowanie, a następnie ich akceptację poprzez odbierającego.
6. **Zarządzanie granicami etapów** - proces kończący określony etap w projekcie i raportujący jego koniec, a także aktualizacja planu projektowego i rejestru ryzyk, przygotowanie i rozpoczęcie następnego etapu.
7. **Zamykanie projektu** - postępowanie kończące przeprowadzony projekt. Rozpoczyna się po osiągnięciu celów projektu poprzez skierowanie prośby kierownika projektu o zakończenie projektu. Podejmowane są takie czynności jak zarejestrowanie zakresu celów projektu, potwierdzenie zadowolenia klienta z produktu, zapewnienie środków odpowiednich do utrzymania i wsparcia.

Projekt PRINCE2 jest również zarządzany w sześciu tematach: zarządzania zakresem, zarządzania czasem, zarządzania kosztami, zarządzania jakością, zarządzania ryzykiem i

korzyściami z projektu [63]. Zarządzanie ryzykiem w ramach metodyki PRINCE2 obejmuje trzy wymiary:

- strategię zarządzania ryzykiem (w jaki sposób zarządzanie ryzykiem zostanie osadzone w działaniach związanych z zarządzaniem projektem, jaka jest tolerancja na ryzyko i kiedy uruchamiany jest wyjątek);
- rejestr ryzyka jako narzędzie do przechwytywania i utrzymywania informacji o zidentyfikowanych wątkach i możliwościach (wsparcie projektu zazwyczaj prowadzi rejestr ryzyka w imieniu kierownika projektu);
- procedurę zarządzania ryzykiem. Proces zarządzania ryzykiem w projekcie zaleca 5 kroków: identyfikację, ocenę (oszacowanie i ewaluację), planowanie, wdrożenie i komunikację [55].

3.6. PMBOK

Project Management Body of Knowledge (PMBOK) to grupa procesów i obszarów wiedzy, które są powszechnie akceptowane jako najlepsze praktyki w ramach dyscypliny zarządzania projektami. PMBOK Guide jest również uznanym na całym świecie standardem, który zapewnia podstawy zarządzania projektami, ponieważ mają one zastosowanie do szerokiego zakresu projektów [60]. PMBOK został utworzony przez Project Management Institute, który jako międzynarodowe stowarzyszenie zrzeszające kierowników projektów, ma na celu wyposażenie ich w najlepsze praktyki wspierające zakończenie prowadzonych projektów sukcesem. Pierwsze dokumenty powstały już w 1987 roku, a pierwsza wersja przewodnika powstała w 1996 roku, natomiast obecnie obowiązująca jest wersja PMBOK 7.0 [64], która wprowadza stosunkowo duże zmiany w porównaniu do wcześniejszych wersji, gdzie każda z nich jest osobną metodyką. Początkowo PMBOK był publikacją referencyjną, działającą jedynie jako ogólne wytyczne dla Kierownika Projektu i nie definiował szczegółowo metod i narzędzi [65]. Pomijał także warstwę, jak projekt powinien być prowadzony, co obejmowałoby zdefiniowane role i obowiązki, etapy, opisy produktów i kolejność działań w procesach.[61].

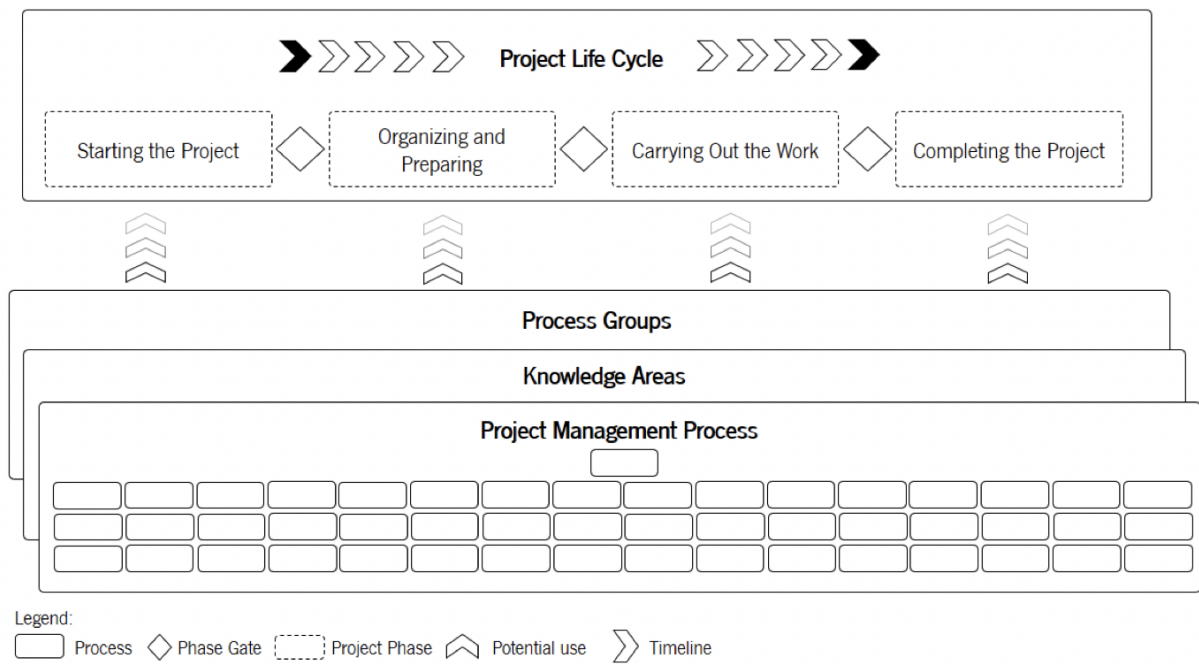
Ze względu na stanowisko PMI mówiące o tym, że PMBOK 7 nie przekreśla wcześniejszych wersji, to połączenie dwóch najnowszych metodyk PMBOK 6 i PMBOK 7, może mieć pozytywny wpływ na przeprowadzenie projektu [66]. W PMBOK 7, wydanym w 2021 r., nastąpiła przerwa w zwykłym formacie publikacji, której towarzyszyło przełamanie niektórych paradygmatów: obszary wiedzy i grupy procesów zostały usunięte, aby zrobić miejsce dla zasad i domen wydajności. Natomiast najlepsze praktyki zostały przeniesione do cyfrowego kompendium o nazwie PMI Standard Plus [67].

PMBOK 6 opisuje procesy, które powinny zostać przeprowadzone podczas realizacji projektu i które można podzielić na 5 głównych grup, oraz na obszary wiedzy. Procesy występujące w PMBOK można podzielić na następujące grupy [60] [68]:

- **Inicjowanie** - opracowanie planu projektu, jego definicji celu, utworzenie opisu i serii działań, ich harmonogramu, a także przewidywanie czasu ich działania. Przygotowany zostaje plan kontroli jakości i zarządzania ryzykiem.
- **Planowanie** - szczegółowe opracowanie planu realizacji projektu, w tym zakresu, harmonogramu, zasobów, budżetu, sposobów kontroli jakości i ryzyka.
- **Realizacja** - wykonywanie wcześniej zaplanowanych prac projektowych, a także zarządzanie zespołem i komunikacją.
- **Monitorowanie i kontrola** - bieżący nadzór nad realizacją prac projektowych, ich harmonogramem i zakresem, a także kontrola jakości i zarządzanie ryzykiem w projekcie.
- **Zamykanie** - działania związane z oficjalnym zamknięciem projektu, uzupełnienie dokumentacji i przekazanie projektu do interesariuszy.

W ramach procesów wykorzystywane są obszary wiedzy, które można podzielić na [69] [68]:

- **Zarządzanie integracją procesów projektowych** - zebranie wszystkich pozostałych obszarów wiedzy w jeden.
- **Zakres** - kontrola ram funkcjonalności rozwiązania.
- **Harmonogram** - przedział czasowy, podczas którego realizowany jest projekt.
- **Koszt** - wartość wszystkich działań realizowanych w ramach projektu.
- **Jakość** - wysoki poziom dostarczanego rozwiązania.
- **Zasoby** - zapewnienie odpowiednich zasobów w odpowiednim terminie i ilości.
- **Komunikacja** - planowany sposób współpracy pomiędzy członkami zespołu.
- **Ryzyka** - minimalizacja zdarzeń mogących mieć negatywny wpływ na realizację projektu.
- **Zamówienia** - odpowiedzialne za planowanie i realizację wszystkich zakupów potrzebnych do realizacji projektu.



Rys. 3.8. PMBOK - Cykl życia projektu [69]

- **Interesariusze** - kroki mające na celu zwiększenie zaangażowania wszystkich jednostek zaangażowanych w projekt.

W PMBOK 7 głównymi metodami podziału projektu jest jego podział na 12 zasad i 8 domen wydajności (Rys.3.9) [64]. Dodatkowo PMBOK 7, jako nadrzędny cel projektu uznaje rezultat (Outcome) zamiast wcześniejszego wyniku (Output). Oznacza to, że projekt oprócz uzyskanego wyniku, musi brać także pod uwagę szerszą perspektywę. Zasady w PMBOK 7 to [70] [64] :

- **Zarządzanie** (ang. Stewardship)
- **Zespół** (ang. Team)
- **Interesariusze** (ang. Stakeholders)
- **Wartość** (ang. Value)
- **Myślenie systemowe** (ang. Systems Thinking)
- **Przywództwo** (ang. Leadership)
- **Dostosowanie** (ang. Tailoring)
- **Jakość** (ang. Quality)
- **Złożoność** (ang. Complexity)



Rys. 3.9. PMBOK 7.0 [70]

- **Ryzyko** (ang. Risk)
- **Zdolność adaptacji i odporność** (ang. Adaptability & Resilience)
- **Zmiana** (ang. Change)

PMBOK jest rozwijany na podstawie doświadczeń kierowników projektów i w odpowiedzi na dynamiczny rozwój przemysłu, zbierając najlepsze praktyki z wielu wymiarów.

3.7. Podsumowanie

Każda z powyżej wymienionych metod ma swoje zalety i wady. Jednakże należy podkreślić, że używanie metodyk projektowych w praktyce wpływa pozytywnie na przebieg projektu, ze względu na ustalony sposób pracy projektowej i planowanie działania według określonego schematu. Dostosowanie się do unikalnych celów, interesariuszy i złożoności środowiska to czynniki, które przyczyniają się do sukcesu projektu ponieważ ma na celu maksymalizację wartości, zarządzanie ograniczeniami i poprawę wydajności poprzez użycie odpowiednich procesów, metod, modeli i artefaktów w celu osiągnięcia pożądaných wyników projektu [67].

Kluczowe dla wyboru odpowiedniej metodyki jest znajomość środowiska projektowego i produktu, który ma być tworzony lub wdrażany, dzięki czemu wybrana zostanie odpowiednia do niego metodyka projektowa. W obecnych czasach stosowane są również metodyki hybrydowe [39], które składają się z najlepszych praktyk wybranych z innych metodyk [36].

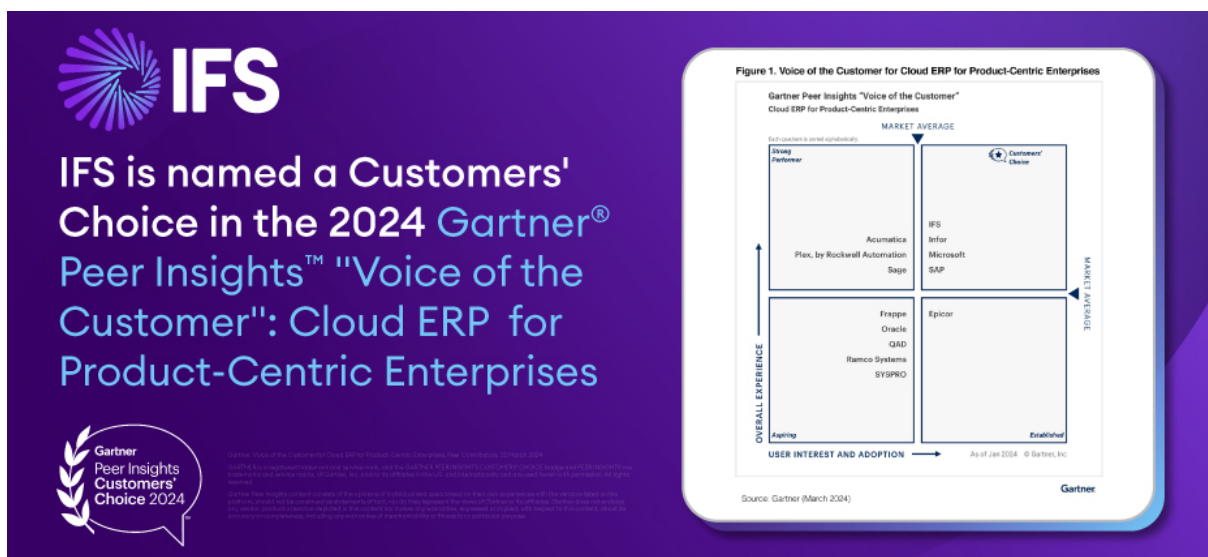
Organizacje zrzeszające kierowników projektów takie jak PMI i PRINCE2 aktualizują swoje metodyki i produkty, gromadząc nowe trendy, które mogą przyczynić się do usprawnienia realizacji projektów [60].

Ze względu na skomplikowanie procesu wdrożeniowego i samego produktu, jakim jest system ERP, metodyki projektów informatycznych nie oferują wystarczającego wsparcia w procesie realizacji i prowadzenia projektu. Trudnością w realizacji projektu przy pomocy metodyki kaskadowej jest charakterystyka procesu wdrażania systemu ERP, który w rzeczywistości nie ma liniowego przyrostu ze względu na możliwe zmieniające się okoliczności i wymagania [71]. W przypadku metodyk zwinnych, ogromną przeszkodą jest zdecydowanie inne podejście do projektu oraz sposobu zarządzania zakresem. Metodyki zwinne charakteryzują się podziałem pracy na krótkie interwały czasowe. Wdrożenie systemu ERP to decyzja strategiczna, mająca sztywno określone ramy i wysoce kosztowna, która obejmuje wiele różnych zależności w całej strukturze organizacyjnej przedsiębiorstwa. Gdzie, każda z funkcjonalności wytworzonych w czasie długotrwałego okresu wdrażania ma wpływ na jego kształt końcowy. Jednakże częściowe włączenie metodyk zwinnych w proces wdrażania systemu ERP może mieć pozytywny wpływ na zakończenie projektu sukcesem [71].

Dlatego też, większość dostawców systemów ERP, stworzyła własne metodyki i wzorce projektowe. Zostały one utworzone biorąc pod uwagę zarówno najlepsze praktyki projektowe, a także konkretne cechy, funkcjonalności i architekturę systemu ERP. Z drugiej strony kluczowi dostawcy oprogramowania ERP w znacznym stopniu definiują swoje metodyki na standardach rynkowych z zakresu zarządzania projektami. W zależności od potrzeb i możliwości adaptacji poszczególnych systemów ERP do zmian i rozwoju oprogramowania w większym stopniu wykorzystali elementy metodyk zwinnych. Natomiast producenci, którzy dążą do wykorzystania systemu w pełnym jego zakresie przy jednocześnie minimalnej skali zmian, adaptują i rekomendują we własnej metodyce kaskadowe podejście do realizacji wdrożenia. Dlatego też, zagadnienia dotyczące metodyki producenta i architektura systemów kluczowych dostawców zostały wyodrębnione w ramach 4 rozdziału.

4. Architektura systemu i metodyka projektowa kluczowych dostawców ERP

W rozdziale przedstawione zostały wyniki badań rynku, dotyczące obecnie używanych metodyk projektowych, a także rozwiązania systemowe i wdrożeniowe systemów ERP. Rozdział obejmuje przegląd architektury najbardziej znanych systemów ERP i metodyk wdrożeniowych pochodzących od dostawców tych systemów. Znajomość dostępnych rozwiązań oferowanych przez producentów systemów ERP jest niezbędnym czynnikiem w procesie optymalizacji procesów wdrożeniowych.



Rys. 4.1. Gartner Customer Choice 2024 [72]

W ramach badania rynku przeprowadzona została analiza obecnych liderów rynku, a także wykorzystywanych przez nich technologii, architektur i metodyk. Na podstawie raportu Gartnera [72] na temat rozwiązań CLOUD ERP dla firm produkcyjnych zdecydowanymi liderami rynku są firmy: IFS, SAP, Microsoft Dynamics 365 oraz Infor. Poniżej umieszczono również *Gartner Customer Choice* z 2024, w którym te firmy również zajmują najwyższe pozycje (Rys.4.1). W ramach przeprowadzonej analizy rynkowej, wykonano przegląd materiałów i dokumentacji technicznej wybranych producentów systemów ERP do których

producenci umożliwili wgląd lub miały one charakter publiczny. Ze względu na cel pracy, czyli optymalizację procesu prowadzenia projektu wdrożeniowego, szczególny nacisk został położony na metodyki projektowe liderów rynku. Dokonana została szczegółowa analiza zasad prowadzenia projektu wdrożeniowego oraz architektury systemów ERP. Głównym celem była analiza i porównanie metodyki IFS Application do pozostałych najpopularniejszych rozwiązań rynkowych, co było podstawą do zaproponowania w kolejnych rozdziałach modyfikacji i rozwinięcia metodyki wdrożeniowej oraz narzędzi informatycznych wspomagających projekt wdrożeniowy.

4.1. SAP

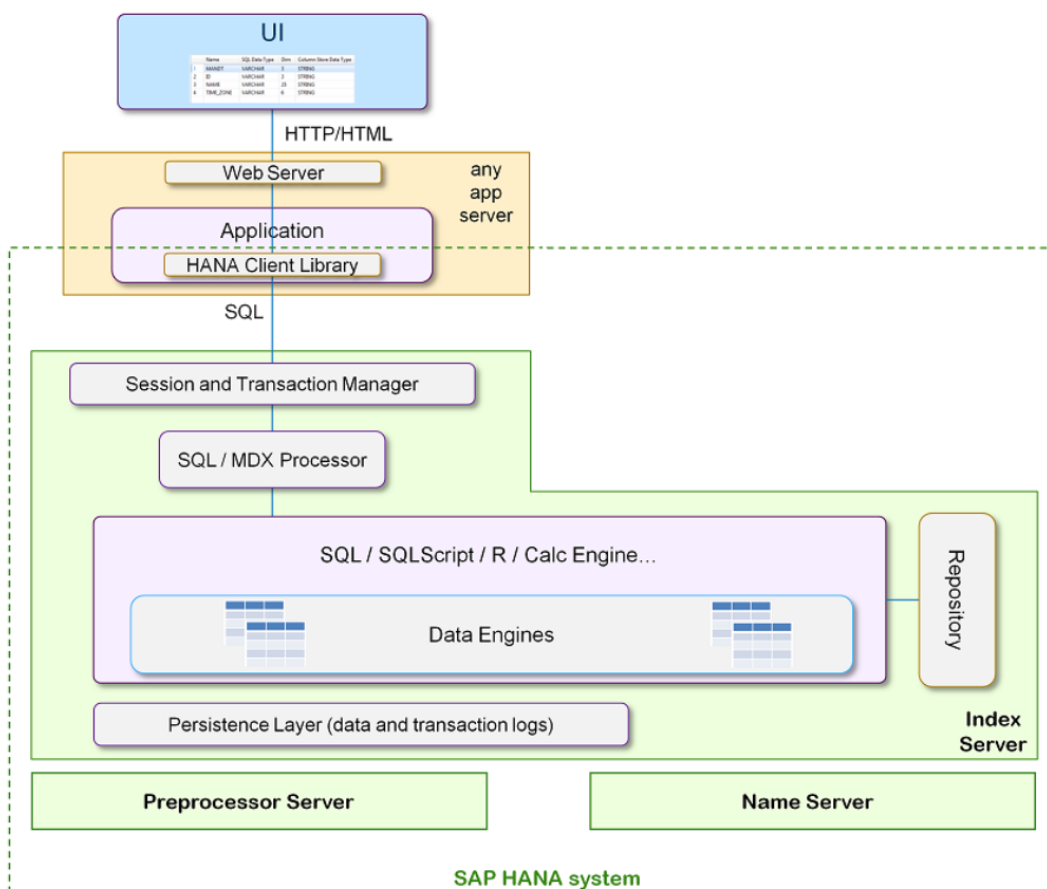
SAP jest najbardziej znanym systemem ERP na globalnym rynku. Firma powstała w 1972 r. w Niemczech i zajmuje się tworzeniem systemu ERP dla dużych i bardzo dużych przedsiębiorstw. Najnowszym produktem jest system SAP S/4HANA, który powstał w 2015 roku. Wcześniejszy produkt SAP Business Suite 7.0 przestanie być wspierany w grudniu 2025 według stanu z dnia analizy.

4.1.1. Architektura

Kluczowym elementem systemu SAP S/4HANA jest wbudowana baza danych (in-memory) SAP HANA. SAP S/4HANA to aplikacja modułowa o modelu klient-serwer, podzielona na warstwy posiadające własne funkcjonalności i zakresy odpowiedzialności. Klient odpowiada za wyświetlanie informacji użytkownikowi, pobieranie od niego danych i przesyłanie ich do serwera. Serwer odpowiada za przechowywanie informacji, odbieranie żądań od klienta, przetwarzanie ich, wykonywanie obliczeń i centralne zarządzanie procesami w aplikacji, a także pobieranie informacji przechowywanych w bazie danych. W ramach SAP S/4HANA, rolę klienta pełni zestaw aplikacji SAP Fiori, natomiast rolę serwera i bazy danych pełni SAP HANA. Architektura wysokopoziomowa, z uwzględnieniem podziału pomiędzy poszczególne warstwy została przedstawiona na rysunku Rys.4.2.

SAP HANA (High-performance Analytic Appliance), to innowacyjna platforma przechowywania danych, która ze względu na swoją budowę umożliwia przyspieszony proces przetwarzania danych i ich analizę w czasie rzeczywistym w porównaniu do innych tradycyjnych baz danych. Oprócz bazy danych, pełni również rolę serwera aplikacji. SAP HANA to baza in-memory, oznacza to, że dane przechowywane są w pamięci RAM zamiast na dyskach twardych, co zwiększa szybkość wykonywania operacji [74].

Dodatkowo baza danych ma strukturę wielo-modelową, czyli zamiast przechowywania kolejnych danych w wierszach, są one przechowywane kolumnowo. SAP HANA spełnia



SAP HANA Database High-Level Architecture

Rys. 4.2. SAP Architektura wysokopoziomowa [73]

również wymagania ACID (atomicity, consistency, isolation, durability) dotyczące właściwości baz danych potwierdzające możliwości poprawnego przetwarzania transakcji bazodanowych. Zaletą tego rozwiązania jest możliwość tworzenia skomplikowanych zapytań bezpośrednio w bazie danych. Baza danych SAP HANA posiada jako część swojego rozwiązania Online Analytical Processing (OLAP) i Online Transaction Processing (OLTP), co umożliwia w jednym czasie wykonywanie analizy i transakcji przez wielu użytkowników. Umożliwia również przechowywanie różnego typu danych: strukturyzowanych i niestrukturyzowanych.

SAP HANA jest napisana w standardowym języku programowania SAP, czyli ABAP. SAP oferuje również zestaw bibliotek, endpointów API i szablonów integracji dla klientów, które umożliwiają połączenie z bazą danych dla innych platform, a także języków programowania takich jak JavaScript, Python, R, Java, Go i inne.

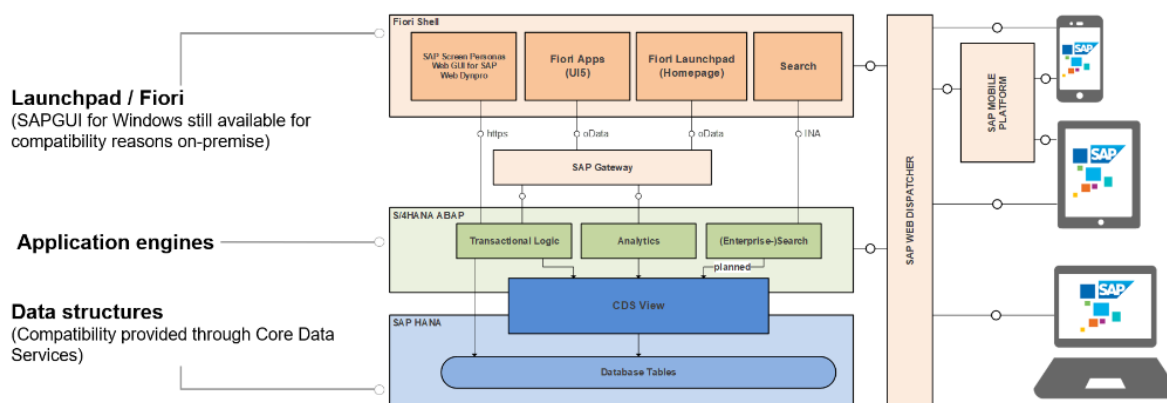
SAP Fiori, to rozwiązanie, umożliwiające tworzenie aplikacji i interfejsów (UI), które mogą połączyć się z SAP HANA i być następnie używane przez końcowych użytkowników systemu. W ramach SAP Fiori dostępny jest zestaw aplikacji bazowych, które odpowiadają na standardowe potrzeby organizacji i wspierają jej kluczowe procesy biznesowe. W ramach rozwiązania powstał zestaw aplikacji, które używane są nie tylko w systemie SAP S/4 HANA, ale także w *SuccessFactors Employee Central solution*, *SAP Ariba mobile solutions*, *SAP Hybris Cloud for Customer solution* itd.

SAP Fiori jest także narzędziem umożliwiającym budowę nowych aplikacji, które pozwalają na personalizację systemu i zapewnienie wypełnienia niestandardowych wymagań funkcjonalnych. SAP Fiori wspiera proces budowy nowych aplikacji poprzez objęcie całego zakresu w procesie budowy nowych aplikacji, dzieląc go na następujące płaszczyzny:

- **SAP Fiori Concept** - schemat działania aplikacji w ramach opracowania zestawu jak najlepszych doświadczeń dla użytkowników aplikacji końcowych. Jest to uproszczenie standardowych rozwiązań i dostosowanie do specyfiki systemu ERP i ogólnych nowoczesnych zasad projektowania aplikacji.
- **SAP Fiori Apps (design)** - to zestaw narzędzi odpowiadający za aplikację od strony wizualnej, ogólnie przyjęte wytyczne oraz wzorce architektury informacji, kolorystyki i interakcji.
- **SAP Fiori technology** - to technologie, infrastruktury i modele od strony technicznej [73].

W systemie SAP S/4HANA szczególnie ważną rolę pełni SAP Web Dispatcher (SWD), który działa zarówno jako reverse proxy, load balancer oraz warstwa bezpieczeństwa i routowania. SAP Web Dispatcher może połączyć się między innymi z SAP S/4HANA, SAP HANA i SAP Fiori, co zostało przedstawione na rysunku Rys.4.3. SWD umożliwia skalowanie systemów, zwiększa wydajność działania oraz zarządzanie przepływem informacji w sieci [75].

Od strony infrastrukturalnej SAP oferuje i rekomenduje wdrożenie swoich systemów w usłudze chmurowej. W przypadku wdrożenia chmurowego możliwe jest skorzystanie albo z własnej infrastruktury chmurowej SAP - SAP HANA Enterprise Cloud lub SAP HANA Cloud, lub wybranie jednego z głównych dostawców technologii chmurowych takich jak Amazon, Azure i Google Cloud. SAP w swoim ekosystemie zapewnia też narzędzie do integracji, analizy i zarządzania danymi, takie jak SAP Data Intelligence Cloud, SAP Datasphere oraz SAP Analytics Cloud. SAP rozwija także ogólnodostępną społeczność ekspertów i użytkowników systemu, w ramach której możliwa jest wspólna dyskusja na temat różnych zagadnień, nowych funkcjonalności itd.



Rys. 4.3. SAP Web Dispatcher [76]

4.1.2. Metodyka wdrożeniowa

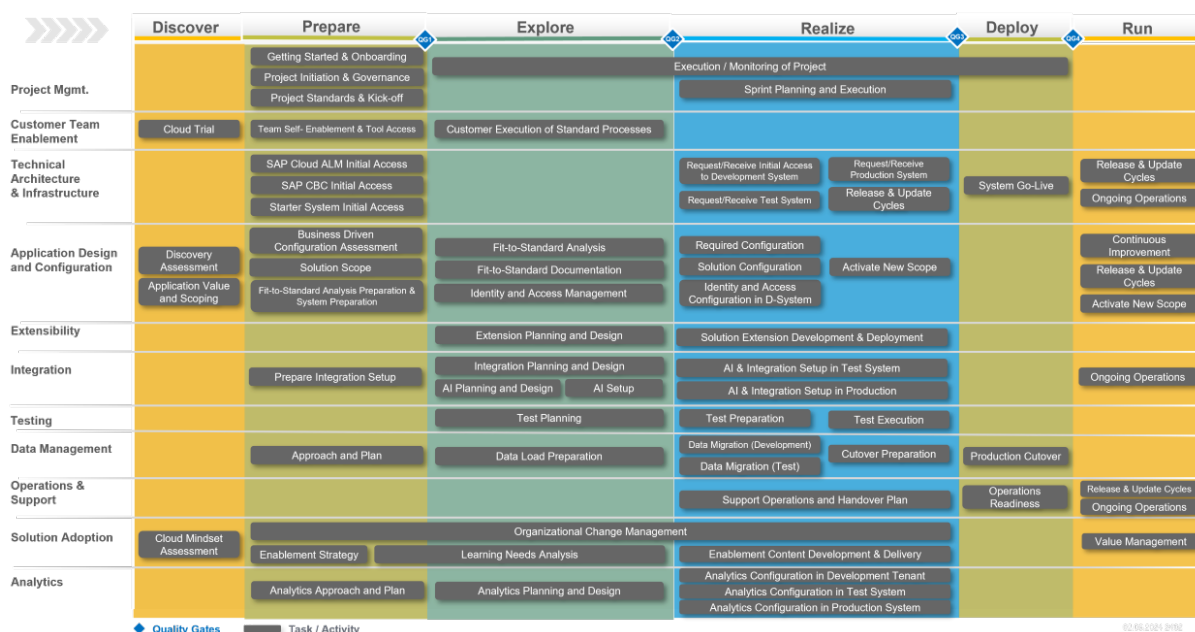
Obecnie rekomendowaną metodyką wdrożeniową przez SAP jest metodyka SAP Activate Methodology, która wywodzi się z wcześniejszych metodyk SAP-a takich jak Accelerated SAP (ASAP). W porównaniu do wcześniejszych metodyk, które skupiały się na optymalizacji procesu wdrożenia on-premise, SAP Activate jest rozwiązaniem dedykowanym dla wdrożeń systemu SAP S/4HANA w chmurze. Wraz z rozwojem technologicznym pojawiła się również istotna zmiana w metodyce. Obecnie zamiast tworzenia zestawu indywidualnych modyfikacji pod klienta SAP Activate Methodology skupia się na stworzeniu standardowych rozwiązań dla konkretnego typu lub branży klienta. W drugiej kolejności realizowane są zmiany w czynnościach operacyjnych organizacji polegające na dopasowaniu procesów biznesowych do tych rozwiązań [77]. Dlatego też, SAP Activate Methodology jest częścią pakietu SAP Activate, w którego skład wchodzi jeszcze równolegle SAP Best Practices i Guided Configurations. Są one dodatkowymi narzędziami wspomagającymi proces wdrożeniowy. **SAP Best Practices**, to zebrany zestaw standardowych, wstępnie skonfigurowanych rozwiązań stworzonych dla różnych branż w celu wdrożenia rozwiązań SAP. Narzędzia SAP są wdrażane i rozwijane od 1972 roku w zakresie najlepszych praktyk instalowania i wdrażania swoich produktów [78]. **Guided Configurations**, to zestaw narzędzi, zasobów i akceleratorów zaprojektowanych z myślą o łatwej konfiguracji systemów SAP i procesów biznesowych. Dodatkowo w swojej dokumentacji SAP udostępnia roadmapy w zakresie metodyki SAP Activate, dla różnego typu projektów i specyfiki klienta [79]. Niezależnie od wyspecyfikowanych wariantów dla poszczególnych rozwiązań o wysokim stopniu szczegółowości, w każdym z nich producent rekomenduje podział na następujące fazy projektu [80]:

- **DISCOVER** Na początku procesu wdrożenia najważniejsza jest decyzja klienta o wdrożeniu systemu ERP lub zmianie nowe rozwiązanie. Klienci dokonują wyboru narzędzi do rozwiązania swoich problemów i starają się wybrać odpowiedni system i roadmapę odpowiadającą ich potrzebom. Dodatkowo następuje rozpoznanie potencjalnych ryzyk, kluczowych wartości biznesowych i możliwości. Proces kończy się wyborem konkretnego rozwiązania SAP, które będzie wdrażane w następnych fazach projektu.
- **PREPARE** Podczas tego etapu następuje zaplanowanie i przygotowanie projektu, a także analiza potencjalnych ryzyk i wyznaczenie mierników jakości projektu. Dodatkowo zostaje przeprowadzany podział na role i odpowiedzialności, a także ogólna akceptacja procedur zarządzania projektem. Przygotowane zostaje dedykowane środowisko klienta wraz z gotowymi podstawowymi procesami. Podczas realizacji etapu analizowane są zagadnienia dotyczące migracji danych ze starych systemów do nowego rozwiązania.
- **EXPLORE** W czasie tej fazy dokonywana jest szczegółowa analiza biznesowa klienta, wyszczególnione zostają kluczowe procesy biznesowe, które następnie w czasie serii sesji SAP Fit-to-Standard Analysis są porównywane z procesami biznesowymi SAP Best Practises i dopasowywane do nich. Możliwa jest również optymalizacja rozwiązania i zaplanowanie nowych procesów biznesowych. Ustalane są kluczowe rezultaty i korzyści, standardowe szablony migracji i potwierdzane są ustalenia dot. kosztów i terminu realizacji w zakresie specyfikacji dodatkowych obiektów i wymagań klienta. Faza kończy się potwierdzeniem dokumentu Backlog, który będzie realizowany w następnej fazie.
- **REALIZE** W czasie tego etapu następuje konfiguracja systemu na podstawie dokumentu Backlog. Kolejne części systemu będą dostarczane i testowane na podstawie harmonogramu. Każde z rozwiązań musi zostać zatwierdzone przez osoby odpowiedzialne za dane obszary w organizacji klienta. Po zakończeniu tego procesu, powstaje plan przeniesienia przygotowanych konfiguracji, obiektów, danych podstawowych i transakcyjnych do finalnie wykorzystywanego w przyszłości systemu.
- **DEPLOY** To faza, w czasie której następuje wyłączenie starych systemów i przejście na nowy system SAP. Odbywa się zwykle w czasie zaplanowanej przerwy pracy w przedsiębiorstwie lub okresie systemowej bezczynności organizacji. Po uruchomieniu systemu organizacja nadal jest wspierana w ramach problemów lub wątpliwości.

- **RUN** To faza końcowa, w czasie której kończy się cykl wdrożenia. Celem tej fazy jest kontynuacja działania systemu, a także systematyczna aktualizacja środowiska ERP o najnowsze konfiguracje i funkcjonalności w ramach filozofii *Continued Learning*.

Na rysunku Rys.4.4 przedstawiona została podstawowa wersja roadmapy z SAP Activate dotycząca wdrożenia systemu w chmurze. Jest to jedna z wielu wersji do wyboru w ramach wersji podstawowej, a w dokumentacji, można znaleźć kilkanaście szablonów, w tym takie dotyczące wdrożenia chmurowego, on-premise, upgrade'u czy skoncentrowanego na danej branży przemysłowej.

SAP Activate for SAP S/4HANA Cloud, public edition (3-system landscape)



Rys. 4.4. SAP Activate [79]

Oprócz metodyk wdrożeniowych SAP rozwinął metodyki dotyczące konkretnych aspektów procesu. Przykładem może być framework *The Secure Software Development Lifecycle at SAP* [81], który koncentruje się na bezpieczeństwie aplikacji zarówno w trakcie jak i po wdrożeniu. Obejmuje on procedury szkoleń użytkowników, metody analizy ryzyka i kontrole bezpieczeństwa. Metodyka SAP, analogicznie do opracowanych przez pozostałych dostawców oprogramowania jest zbiorem reguł, które każdy z partnerów wdrożeniowych powinien stosować i przestrzegać. Analiza rynku wdrożeniowego, literatury z zakresu wdrożeń SAP ERP pokazuje, że wielu integratorów podejmuje się dostosowania standardów producenta do potrzeb rzeczywistych projektów wdrożeniowych. Producent definiuje co najmniej dwie płaszczyzny projektu wdrożeniowego: konfiguracja instalowanego u klienta środowiska w ramach realizowanego projektu oraz skorelowana z tym środowiskiem metodyka wdrożeniowa. Dla obecnego klienta SAP proponowana jest instalacja SAP4/HANA Cloud

Private Edition z możliwością przeniesienia założeń z dotychczasowej instalacji. W przypadku tego typu projektu, rozumianego jako upgrade, są proponowane dedykowane rozwiązania w zakresie metodyki wdrożeniowej np *Rise with SAP - Tailored-To-Fit* [82]. Dla nowego klienta proponowana jest instalacja z konfiguracją zależną od branży, potrzeb i możliwości adaptacyjnych do rozwiązań standardowych systemu. Tego typu instalacja określona jest przez producenta jako SAP4/HANA Cloud Private Edition. W przypadku wdrożeń od podstaw integratorzy rekomendują dedykowane pod ten typ projektu narzędzia i metodykę wdrożeniową np. *GROW with SAP: READY-TO-RUN* [83]

4.2. Microsoft Dynamics 365

Microsoft to jedna z najbardziej znanych firm informatycznych, która powstała w USA w 1975 roku. Microsoft oferuje paletę produktów i rozwiązań, takich jak systemy operacyjne, usługi chmurowe, programy do edycji danych, platformy komunikacyjne itd. W zakresie systemów zintegrowanego zarządzania organizacją jednym z najpopularniejszych rozwiązań rynkowych jest Microsoft Dynamics 365. Jego dużą zaletą jest możliwość elastycznej i bardziej rozbudowanej integracji z innymi produktami firmy Microsoft.

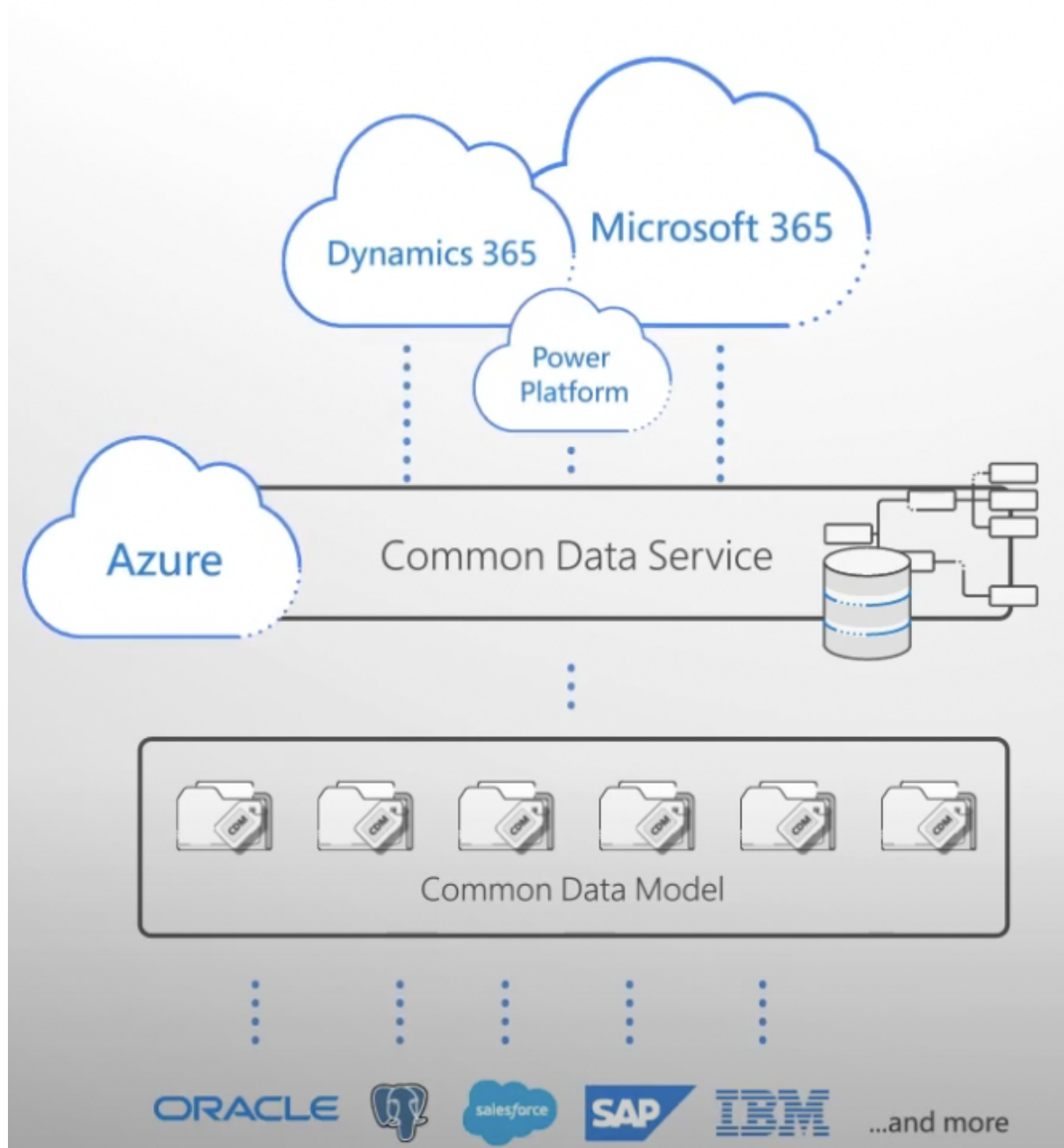
4.2.1. Architektura

Microsoft Dynamics to pakiet aplikacji do planowania zasobów przedsiębiorstwa (ERP) i zarządzania relacjami z klientami (CRM). Jego architektura jest modułowa, skalowalna i zaprojektowana do integracji z usługami chmurowymi i lokalnymi serwerami. Ze względu na to, że Microsoft zapewnia zarówno rozwiązanie ERP, jak i jest jednym z największych dostawców przestrzeni chmurowej jako właściciel Azure, wiele rozwiązań jest ze sobą bezpośrednio powiązanych i dostępnych w ramach kompleksowej oferty systemu na system ERP jako usługi (SaaS). [84] Narzędzia użytkownika Microsoft Dynamics, które osadzone są w chmurze Azure działają natywnie. W skład MS Dynamics 365 wchodzi zbiór aplikacji, które mogą działać osobno, lub mogą być ze sobą połączone w zależności od potrzeb docelowego klienta. Jak w każdym innym modelu SaaS, klient podejmuje decyzję, które procesy będzie realizował przy wsparciu systemu i na tej podstawie dobierane są odpowiednie aplikacje/komponenty Microsoft Dynamics 365. Oprócz realizacji podstawowych procesów dobrze znanych z innych rozwiązań ERP, takich jak: Finanse, Sprzedaż, HR, Dystrybucja, Projekty, FSM i inne obszary otrzymujemy narzędzia i możliwość:

- możliwości integracji z office365, jako element rozwiązania *Business Central*,
- dedykowane narzędzia dla obszaru Sales z zakresu sztucznej inteligencji, dostępne jako Copilot w połączeniu z Azure OpenAI,

- power platform - zestaw narzędzi chmurowych, które umożliwiają organizacjom tworzenie aplikacji, automatyzowanie procesów, analizowanie danych i budowanie rozwiązań w oparciu o rozwiązania low-code.

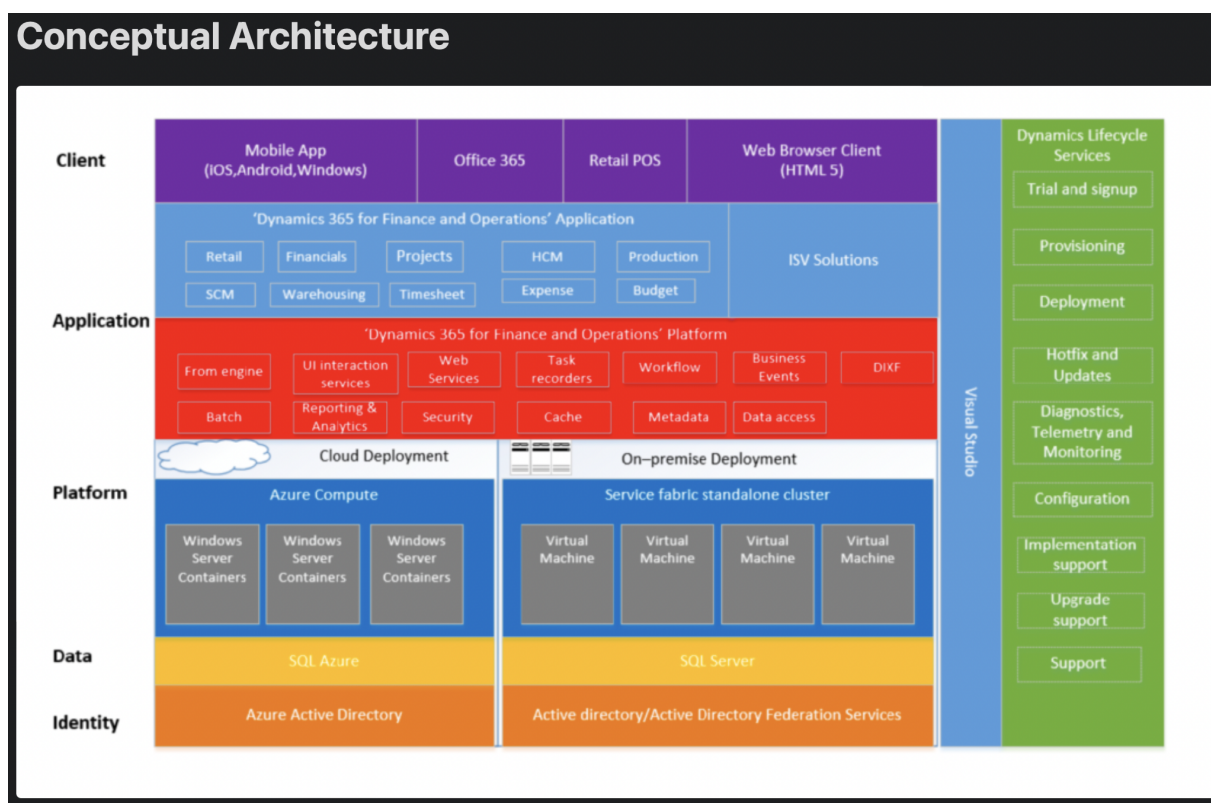
Architektura Microsoft Dynamics 365 w oparciu o usługi chmurowe w znacznym uproszeniu jest przedstawiona na rysunku Rys.4.5, dla której przedstawiony został sposób interakcji pomiędzy aplikacjami, narzędziami power platform, bazą danych oraz integracją z innymi rozwiązaniami klienta.



Rys. 4.5. Microsoft Dynamics 365 - Architektura wysokopoziomowa [84]

Architektura dla rozwiązań ERP różni się w zależności od wdrożenia: w chmurze (cloud) lub on-premise (lokalne centrum danych). Na przykładzie Microsoft Dynamics 365 opisane zostały obie te architektury na rysunku Rys.4.6 oraz w tabeli 4.1. **Klient** Interfejs użytkownika (UI): łączy się z Microsoft Dynamics 365 poprzez przeglądarkę internetową (web), aplikację mobilną lub dedykowane aplikacje. Komunikacja odbywa się za pomocą HTTPS, co zapewnia szyfrowanie transmisji danych. **Aplikacja** Dynamics 365 jako SaaS (Software as a Service):

działa w środowisku chmurowym Microsoft Azure. Klient nie zarządza bezpośrednio serwerami ani infrastrukturą. **Interfejs API** Aplikacja udostępnia RESTful API, które umożliwia integrację z innymi systemami klienta, np. CRM, narzędzia BI, i inne. **Baza danych** Microsoft Azure SQL Database: Dane aplikacji są przechowywane w rozproszonych bazach danych w chmurze Microsoft Azure. Microsoft zarządza kopiami zapasowymi, odtwarzaniem danych oraz ich replikacją. **Uwierzytelnianie** w środowisku Microsoft: Uwierzytelnianie w Dynamics 365 opiera się na Azure Active Directory (AAD), gdzie użytkownicy muszą przejść autoryzację za pomocą OAuth 2.0. W zależności od ustawień organizacji, klient/użytkownik może być zobowiązany do użycia wieloskładnikowego uwierzytelniania Multi-Factor Authentication (MFA), np. przez SMS, aplikację mobilną lub klucz sprzętowy.



Rys. 4.6. Microsoft Dynamics 365 - Architektura [84]

Przedstawione poniżej porównanie rozwiązań chmurowych (Azure) z instalacją on-premise ma na celu zobrazować różnice w podejściu do bezpieczeństwa i możliwości rozwojowych dotychczasowych systemów ERP z obecnie oferowanymi rozwiązaniami.

Zgodnie z porównaniem informacji dotyczących instalacji ERP w chmurze Azure i on-premise można wyciągnąć wnioski, że architektura Azure oferuje większą elastyczność, automatyzację oraz bezpieczeństwo zarządzane przez Microsoft, podczas gdy wdrożenie

Tabela 4.1. Architektura Dynamics 365 vs instalacja

Rodzaj instalacji	Chmura	On-premise/chmura sprzętowa
Dostępność	Zdalny dostęp (internet)	sieć lokalna lub VPN
Infrastruktura	Zarządzana przez Microsoft Azure	Zarządzana przez wewnętrzny zespół IT
Skalowalność	Dynamiczne skalowanie w chmurze	zgodne z infrastrukturą lokalną
Baza danych	Azure SQL Database	MS SQL Server (zarządzany lokalnie)
Uwierzytelnianie	Azure Active Directory	Active Directory (op. LDAP, MFA)
Bezpieczeństwo	Microsoft	po stronie klienta

on-premise daje większą kontrolę, ale wymaga znacznie większego zaangażowania zasobów IT organizacji.

Architektura integracji i możliwości rozwoju dodatkowych aplikacji klienta z Microsoft Dynamics 365 zapewnia bezpieczną wymianę danych. Komunikacja opiera się na protokole HTTPS (Hypertext Transfer Protocol Secure), który szyfruje przesyłane dane za pomocą TLS (Transport Layer Security). Klient może nawiązywać połączenie za pomocą żądań GET, POST, PUT lub DELETE, które są szyfrowane, zapewniając integralność i poufność danych. Aby aplikacja klienta mogła komunikować się z Dynamics 365, konieczna jest autoryzacja za pomocą Azure Active Directory (AAD) przy użyciu standardu OAuth 2.0 4.7. Po uzyskaniu tokenu dostępu, aplikacja może przysyłać żądania poprzez https do interfejsów API Dynamics 365, uzyskując dostęp do danych i usług w bezpieczny sposób.

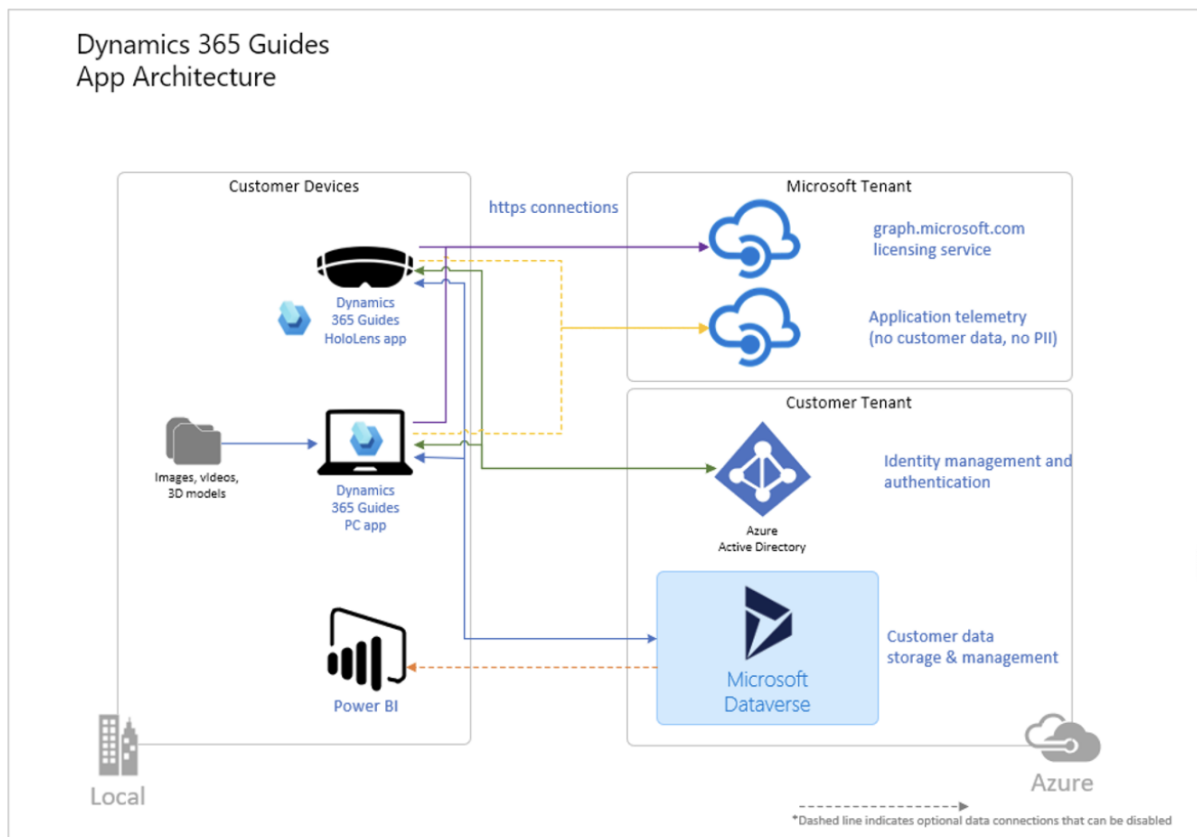
4.2.2. Metodyka wdrożeniowa

W przypadku Microsoft Dynamics 365 można wyróżnić dwie metodyki takie jak Microsoft Sure Step i Microsoft Success by Design. Pierwsza z nich jest używaną od wielu lat metodyką wdrażania systemów ERP, natomiast druga z nich to nowsza metodologia, skupiona na wdrożeniach chmurowych. Wielu partnerów nadal bazuje lub korzysta z Microsoft Sure Step, dlatego w poniższym rozdziale przybliżone zostały obie metodyki.

Metodyka Microsoft Sure Step

Microsoft Sure Step można podzielić na następujące fazy projektu:

- **Diagnostic** W tej fazie następuje ocena procesów biznesowych i infrastruktury klienta, a także zasobów i możliwości dopasowania do rozwiązania. Przygotowany zostaje plan projektu, propozycja rozwiązania i dokument "Statement of Work".
- **Analysis** Celem tego etapu jest dokładna analiza obecnie występujących procesów biznesowych, przeprowadzenie analizy fit-gap i przygotowanie dokumentu z wymaganiami funkcjonalnymi, a także specyfikacji środowiska.



Rys. 4.7. Microsoft Dynamics - architektura wysokopoziomowa
[84]

- **Design** Tworzony jest projekt wdrożenia i dokumenty dotyczące projektu funkcjonalnego, technicznego i całego rozwiązania. Przygotowana zostaje strategia migracji, a także ustalone zostaną kryteria testowe.
- **Development** W tej fazie tworzone i kończone są konfiguracje, fragmenty kodu z rozwiązaniem użytkownika. Przeprowadzane są testy funkcjonalności, przygotowana jest również dokumentacja dla użytkowników końcowych.
- **Deployment** W tym etapie tworzone jest środowisko produkcyjne, do którego następnie migrowane są dane klienta. Przeprowadzone są kompleksowe testy akceptacyjne rozwiązania. Przeprowadzane są szkolenia użytkowników końcowych, po których następuje zakończenie procesu tworzenia instrukcji stanowiskowych i dokumentacji. Przeprowadzone zostaje sprawdzenie go-live i uruchomienie czynności związanych ze startem produkcyjnym systemu.
- **Operation** W ostatnim etapie wdrożenia, zostają wykonane takie czynności, jak rozwiązywanie otwartych zagadnień, zakończenie procesu transferu wiedzy do organizacji klienta, a także przeprowadzenie procesu lessons learnt lub post-mortem

projektu. W ramach kontynuowania współpracy, klient przechodzi na support, który ma na celu wsparcie systemu w poprawnym działaniu i pracy.

Microsoft Sure Step ma również dwa podwarianty: 1) Proces optymalizacji i 2) Upgrade. W procesie optymalizacji na początku następuje ocena systemu, czy jest zaprojektowany i oddany tak, żeby spełnić kluczowe potrzeby i wymagania klienta, a także analiza systemu pod kątem możliwej optymalizacji. W projekcie Upgrade również na początku następuje ocena istniejących procesów biznesowych. Następnie tworzone są wymagania dla nowego zakresu funkcjonalności.

Metodyka Microsoft Success by Design

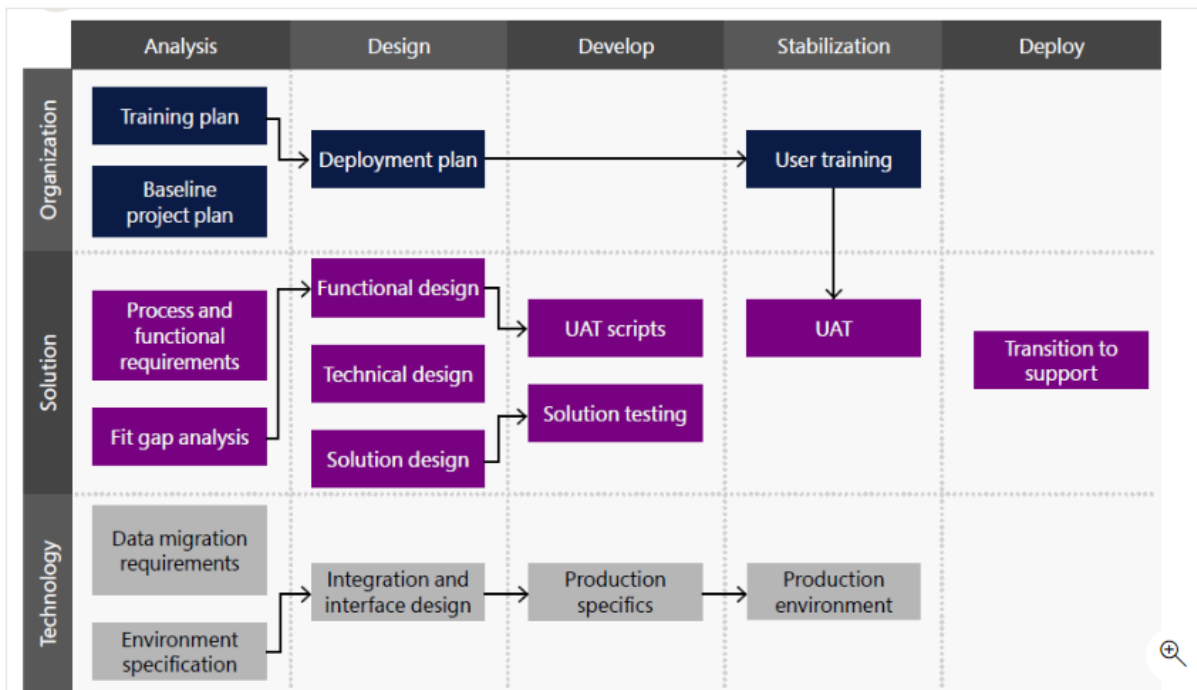
Microsoft Success by Design to framework i zestaw praktyk pomagających we wdrożeniu systemu Dynamics 365. Jego zadaniem jest wsparcie przedsiębiorstwa niezależnie od tego czy wybierze do wdrożenia model kaskadowy, zwinny czy hybrydowy.

- **Agile** To elastyczna metodyka, która opiera się na współpracy oraz szybkim reagowaniu na zmieniające się wymagania i informacje zwrotne. Celem tego działania jest połączenie standardowych procedur z backlogu wraz ze zmianami systemu w zależności od przypadku, wspólnej współpracy z użytkownikami i dostosowywaniu się do zmiany (Rys.4.8).



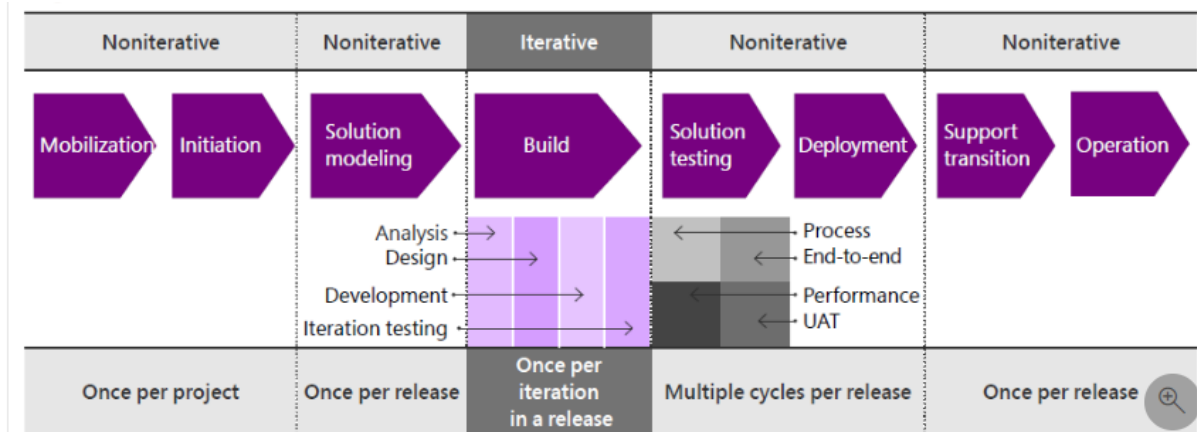
Rys. 4.8. Microsoft Agile

- **Waterfall** To ustrukturyzowana metodyka, która opiera się na podziale na poszczególne etapy i działania, a także definiuje z góry kolejność wykonywanych działań i kamienie milowe. Rozpoczęcie kolejnej fazy jest możliwe tylko po zakończeniu wszystkich działań z wcześniejszej fazy (Rys.4.9).
- **Hybrid** To połączenie najlepszych praktyk z Agile i Waterfall. Działania są podzielone na iteracyjne i nieiteracyjne. Działania nieiteracyjne w tym przypadku, to czynności, które mają miejsce w projekcie tylko raz i mają znaczenie decyzyjne. Przykładem może być inicjacja projektu, testy akceptacyjne i całkowite wdrożenie. Działania iteracyjne, to



Rys. 4.9. Microsoft Waterfall

takie, które mogą być tworzone stopniowo i udoskonalane w procesie tworzenia rozwiązania. Przykładem może być uszczegóławianie wymagań, projektowanie i analiza rozwiązania, a także rozwój funkcjonalności systemu (Rys.4.10).



Rys. 4.10. Microsoft Hybrid

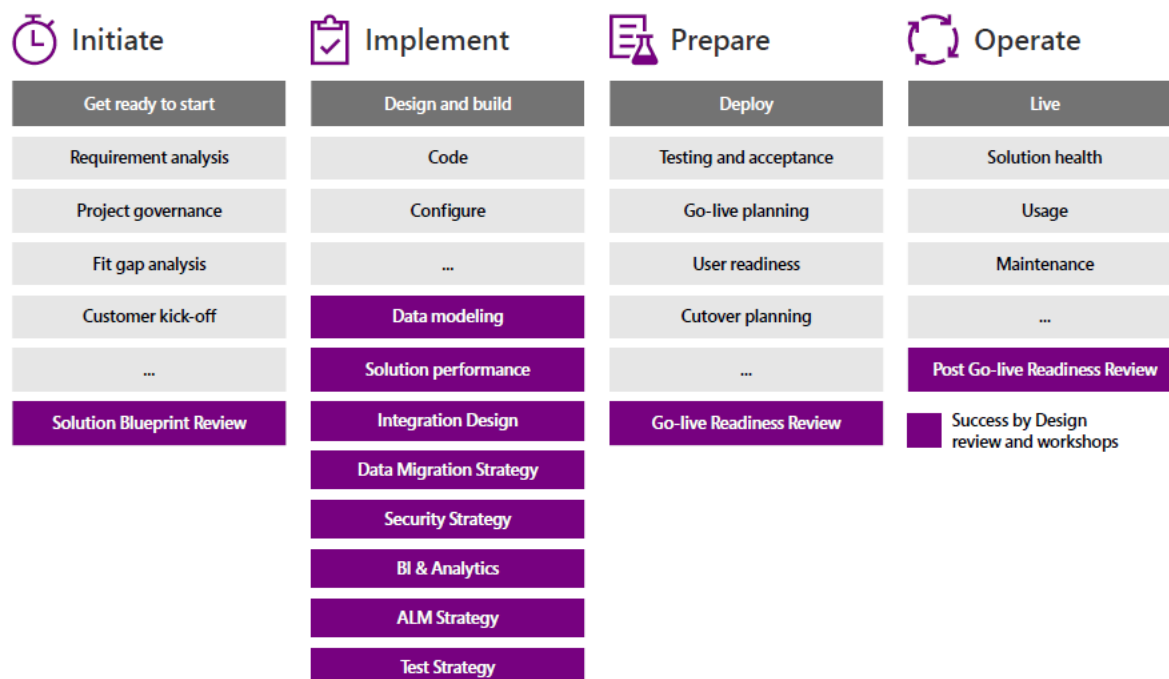
Dodatkowo w dokumentacji Microsoft znajdują się takie propozycje uruchomienia aplikacji jak Big Bang, wdrożenie stopniowe i wdrożenie równoległe. W przypadku Big Bang, w jednym momencie następuje zmiana ze starego systemu na nowy. We wdrażaniu stopniowym zmiana z nowego systemu na stary jest podzielona na fazy lub release części oprogramowania. Może być podzielona ze względu na lokalizacje, moduły, priorytet itd. Wdrożenie równoległe, to wariant

wdrożenia stopniowego, w którym w czasie wdrażania nowego systemu, stary nadal działa. Celem tego jest zachowanie operacyjności przedsiębiorstwa w czasie, w którym użytkownicy końcowy uczą się nowego systemu.

Microsoft Success by Design składa się z następujących faz (Rys.4.11):

- **Initiate** To faza, w której zespół projektowy odkrywa, gromadzi i weryfikuje wiedzę o kliencie, taką jak wymagania biznesowe, tworzy wysokopoziomowy projekt rozwiązania, analizuje i aktualizuje zakres i plan projektu.
- **Implement** W tym etapie budowane jest rozwiązanie zgodnie z uzgodnionym zakresem i planem projektu. Dodatkowo projekt jest analizowany także pod kątem Solution Blueprint Review, czyli zestawu ustaleń i zaleceń, przygotowanych na podstawie wcześniejszych implementacji projektów, które skupiają się na różnych aspektach projektu, takich jak przykładowo model danych, bezpieczeństwo, integracja i strategia testowa. Na podstawie danych z wcześniejszych wdrożeń, łatwiejsze jest zidentyfikowanie ryzyk, które mogą powstać w czasie wdrożenia. Ważne jest wykonanie przeglądu ryzyk w tym etapie, aby rozwiązanie nie było na zbyt wysokim poziomie złożoności i nie posiadało znacznego zakresu modyfikacji kodu na potrzeby projektu.
- **Prepare** W tej fazie następuje zakończenie budowy rozwiązania i przeprowadzone są testy sprawdzające jego działanie, w tym także ostatnie testy akceptacyjne użytkowników (UAT) i szkolenia. Zakończone są przeglądy bezpieczeństwa, ustalony zostaje plan przejścia ze starego systemu na nowy i uruchomienia nowego systemu. Zakończona jest budowa modelu wsparcia i dokumentacji dla klienta. Dodatkowo przeprowadzona jest procedura *Success by Design Go live Readiness Review*, która ma na celu identyfikację pozostających do realizacji zagadnień i problemów w systemie.
- **Operate** W ostatnim etapie, nowy system z rozwiązaniem klienta rozpoczyna działanie. Celem tej fazy jest stabilizacja i przejście na działanie przedsiębiorstwa w nowym systemie zgodnie z zakładanym scenariuszem. Następnymi krokami może być rozbudowa funkcjonalności i dodanie nowych ulepszeń do działającego już systemu.

Dodatkowo Microsoft dla potencjalnych klientów spełniających wymagania stawiane metodyką, oferuje program wdrożenia Fast Track for Dynamics 365, które umożliwiły przyspieszone wdrożenie systemu i rozwiązań w chmurze. Podejście to, możliwe jest do zastosowania nie tylko dla Dynamics 365, ale także dla innych usług chmurowych.



Rys. 4.11. Microsoft Success by design Methodology

4.3. IFS

Firma IFS została założona w 1983 roku w Szwecji. Od tego czasu powstało kilka wersji systemu ERP oferowanego przez tego dostawcę oprogramowania. Najnowsza wersja to IFS Cloud, która miała swoją premierę w marcu 2021 roku i do chwili obecnej jest jedyną oferowaną przez producenta wersją w sprzedaży.

4.3.1. Architektura

IFS Cloud to aplikacja 'evergreen', która jest na bieżąco rozwijana, aktualizowana i poprawiana. Architekturę systemu IFS można podzielić według kilku kryteriów, producent proponuje następujące podziały: podstawowy wysokopoziomowy opis architektury, rozbudowany opis architektury logicznej z wyjaśnieniem podziału na warstwy i odpowiedzialności między nimi oraz architekturę fizyczną, gdzie aplikacja jest podzielona na części według używanych technologii.

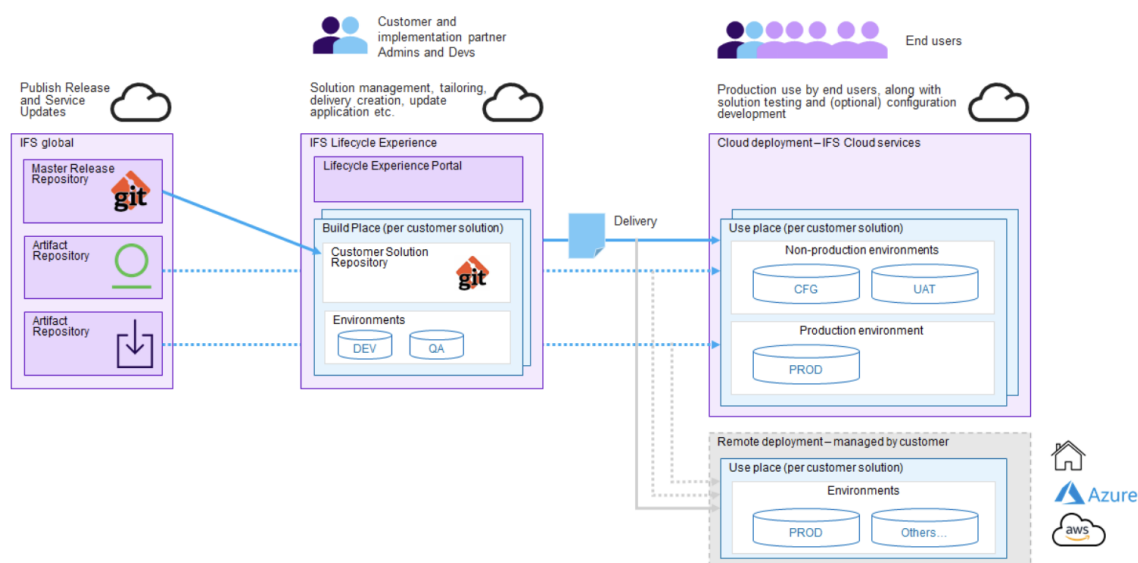
Według podziału wysokopoziomowego aplikacja IFS, składa się z następujących warstw architektury [85]:

- **Warstwa kodu źródłowego** Zawiera standardowy początkowy kod aplikacji, jest to czysty szablon rozwiązania, w którym znajdują się niezbędne moduły i funkcjonalności, dla rozpoczęcia pracy wdrożeniowej. Kod źródłowy jest przechowywany w Master

Release Repository, gdzie na bieżąco dodawane są nowe funkcjonalności i zmiany aktualnego kodu.

- **Warstwa z indywidualnym rozwiązaniem biznesowym** W tej warstwie standardowe rozwiązanie IFS Cloud jest dopasowywane do konkretnego rozwiązania biznesowego. W ramach konkretnej instalacji klienta wybierane są moduły, które będą wykorzystane w docelowym rozwiązaniu. W dedykowanym Build Place tworzone są dostosowania aplikacji do potrzeb danego klienta. Możliwe są zmiany i dostosowanie obecnego systemu lub napisanie nowych funkcjonalności, a także konfiguracja i budowa połączeń z systemami zewnętrznymi.
- **Warstwa deploymentu aplikacji** Aplikacja IFS może być oferowana jako aplikacja Cloudowa lub jako aplikacja dostępna on-premise, która jest instalowana w dowolnej lokalizacji chmury sprzętowej lub na serwerach klienta. Wdrożenie w chmurze jest bardziej zautomatyzowane, a część funkcjonalności może być od razu zainstalowana automatycznie.

Taka architektura rozwiązania jest jedną z przyczyn, które powoduje przeprowadzenie procesu budowy rozwiązania dla klienta (Bild Place) tak jak przedstawiono na rysunku Rys.4.12. Z globalnego repozytorium pobierany jest szablon na repozytorium rozwiązania użytkownika. Do repozytorium użytkownika dodawane są nowe części rozwiązania. Producent niezależnie od przyjętego modelu wdrożeniowego, w chmurze (np. Azure) lub zdalnie w lokalizacji wybranej przez klienta, kody źródłowe przechowuje we własnej chmurze.



Rys. 4.12. IFS Wysokopoziomowa architektura rozwiązania

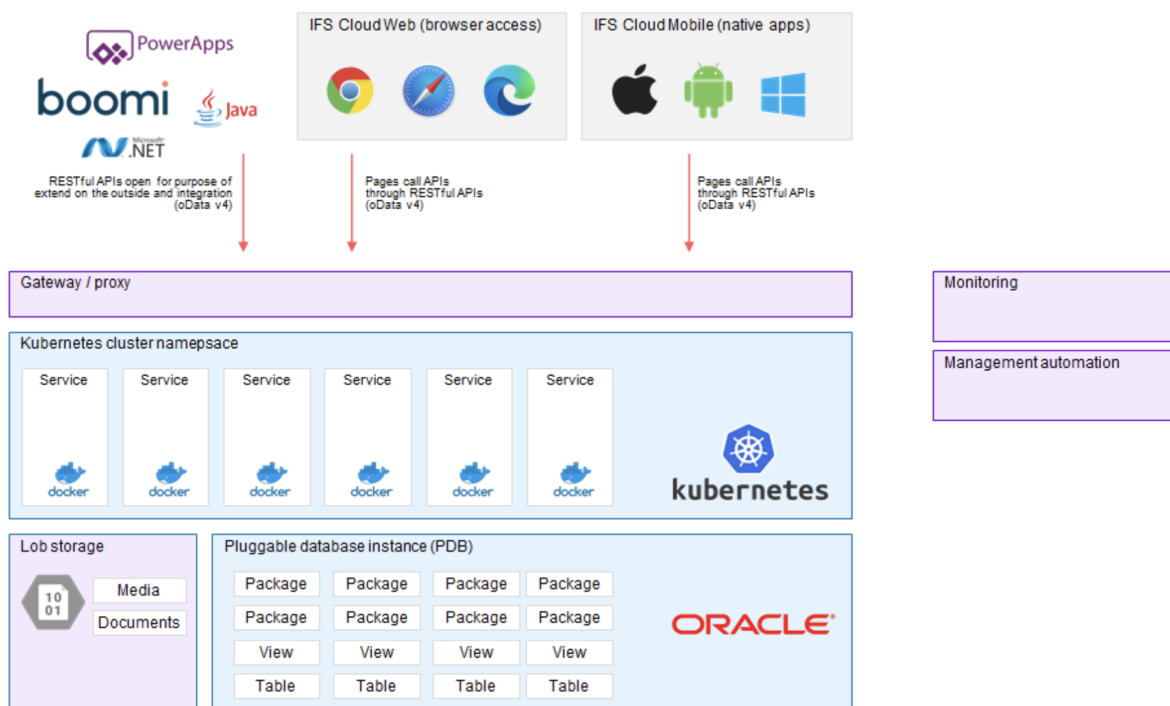
Z perspektywy aplikacji i możliwości rozwiązania konkretnego problemu lub procesu biznesowego opracowano bardziej szczegółowy podział architektury na warstwy. IFS w swojej architekturze korzysta z zasad Layered Application Architecture (LAA), gdzie każda wersja jest zbudowana na poprzedniej. Dzięki temu w aplikacji powstaje podział odpowiedzialności za zasoby na poszczególne warstwy, które zostały przedstawione w tabeli 4.2. W takim modelu aplikacji, następuje podział odpowiedzialności pomiędzy dostawcę technologicznego i klienta, w którym każda ze stron jest odpowiedzialna za określone części środowiska systemu.

Tabela 4.2. Architektura aplikacji IFS Cloud

Warstwa	Cel	Użytkownik	Zawartość
Personalizacja	Własne ustawienia aplikacji, zakładki, zapisane wyszukiwania	Użytkownicy końcowi	Ustawienia zapisane w profilu użytkownika
Konfiguracje	Elementy rozwiązania klienta, nowe projekty ekranów	Klient	Meta dane przechowywane w IFS CCloud i repozytorium rozwiązania klienta
Customizacje	Elementy rozwiązania klienta wymagające dodania nowych funkcjonalności do systemu, zmiana podstawowych funkcjonalności, logiki biznesowej	Klient	Kod rozwiązania użytkownika, przechowywany na jego repozytorium
Core application	Standardowe oprogramowanie IFS	IFS	Główny kod aplikacji, przechowywany w Master Release Repository
Core platform	Standardowe oprogramowanie IFS	IFS	Kontenery, template kodu, narzędzia, przechowywany w Master Release Repository

Architekturę systemu można również podzielić na warstwy fizyczne systemu, wtedy podział następuje według używanych technologii. Fizyczna architektura systemu oraz wykorzystane technologie różnią się częściowo ze względu na wybrany sposób wdrożenia - wdrożenie systemu w chmurze (np. Azure) czy on-premise (w tym chmurze sprzętowej). Dla obydwóch typów wdrożeń istnieje warstwa interfejsu, która umożliwia użytkownikom końcowych aplikacji kilka możliwych opcji korzystania z systemu (Rys.4.13). Istnieje kilka rodzajów interfejsów, które są odpowiedzialne za komunikację z użytkownikiem, przekazywanie i pobieranie od niego informacji. Możliwe jest ustawienie zestawów uprawnień

dla użytkowników aplikacji, a także personalizacja ich ekranów. Głównym interfejsem jest IFS aplikacja webowa Cloud Web, która zapewnia dostęp do systemu przy pomocy przeglądarek internetowych (Chrome, Edge, Safari). Dodatkowo dla komunikacji B2B możliwe jest użycie IFS Cloud Web B2B, który zawiera tylko funkcjonalności niezbędne dla tego typu użytkowników. Z myślą o osobach pracujących na dedykowanych urządzeniach mobilnych (produkcja, magazyn, logistyka), powstała również aplikacja mobilna IFS Cloud Mobile. W ramach tego rozwiązania końcowi użytkownicy systemu mają dostęp do potrzebnych im funkcjonalności systemu w uproszczony sposób. Kolejną częścią architektury równoległą do interfejsów dla użytkowników jest warstwa węzłów komunikacyjnych (endpointów) wychodzących z aplikacji, które umożliwiają integrację z innymi platformami. Przykładem takiej platformy może być platforma integracyjna Boomi. Następnie użytkownicy łączą się przez serwer proxy z usługami aplikacji, które są zarządzane i działają w oparciu o platformę Kubernetes. Sercem systemu w zakresie danych jest Oracle, gdzie przechowywane są wszystkie dane. Jeżeli wdrożenie realizowane jest w chmurze, wtedy oferowane jest rozwiązanie SaaS (Software as a Service), a dane przechowywane są w chmurze np. Microsoft Azure i odpowiedzialny za nie jest IFS. Klient może wybrać też opcję wdrożenia on-premise, co jest dobrą opcją dla klientów, którzy chcą samodzielnie obsługiwać swoje środowisko IFS Cloud oraz system zarządzania relacyjną bazą danych Oracle (RDBMS).



Rys. 4.13. IFS Stack technologiczny

4.3.2. Metodyka wdrożeniowa

Metodyka projektowa IFS jest bazą, która posłużyła do rozwoju metodyki wdrożeniowej opracowanej w ramach rozdziału 6. Natomiast sama metodyka IFS bazuje na zasadach i pryncypiach, które są podstawą metodyk kaskadowych. Metodyka IFS dzieli projekt wdrożeniowy na 5 faz:

- **Rozpoczęcie projektu:** Na początku projektu następuje przekazanie informacji od sprzedaży do grupy wdrażającej. Przygotowywane są pierwsze dokumenty i proponowany zakres rozwiązania przy pomocy narzędzia Scope Tool.
- **Potwierdzenie prototypu:** W tej fazie stworzony zostaje prototyp docelowego planowanego do uruchomienia realnego rozwiązania. Mapowane są również szczegółowo wymagania biznesowe na rozwiązania systemowe w obszarach, które zostaną wykorzystane w docelowym rozwiązaniu.
- **Przygotowanie rozwiązania:** Przygotowywane zostają rozwiązania, szczegółowe scenariusze testowe oraz testy rozwiązania, które zostały przygotowane na podstawie konsultacji z klientem w ramach Fazy II. Dodatkowo planowane i przygotowywane są CRIM (customizacje, raporty, interfejsy i modyfikacje) oraz import danych zgodnie z opracowaną strategią migracji danych.
- **Implementacja:** W tej fazie system jest przygotowywany do uruchomienia oraz finalnie konfigurowany zgodnie z wymaganiami klienta. Tworzone są konta użytkowników i ustawiane są zakresy uprawnień i zabezpieczenia. Rozpoczyna się proces uruchomienia i migracji danych zgodnie ze scenariuszem rozruchu produkcyjnego. Przygotowywane są instrukcje stanowiskowe i szkoleni są użytkownicy końcowi systemu.
- **Go Live:** W tym etapie zakończone zostają wszystkie przygotowania do startu, zaległe zadania i testy. Odbywa się przejście ze starego systemu na nowy system IFS, po którym następuje okres wsparcia po uruchomieniu i przekazanie rozwiązania do dalszego serwisu.

Metodyka IFS zostanie bardziej szczegółowo opisana i rozwinięta w dalszych rozdziałach pracy.

4.4. Podsumowanie przeglądu systemów

Główni dostawcy systemów ERP mają bardzo ograniczone zbiory dokumentacji dostępne publicznie, większość materiałów i informacji dostępna jest poprzez udział w szkoleniach lub

Tabela 4.3. Architektura producentów ERP Cloud

Obszar System	SAP S/4HANA	Microsoft Dynamics 365	IFS Cloud
Platforma	SAP Cloud Platform, AWS, Azure, Google Cloud	Microsoft Azure	Microsoft Azure
Baza danych	In-memory SAP HANA	Common Data Service (CDS) + Azure SQL	Oracle DB
Architektura	Mikrouslugi, in-memory database	Mikrouslugi na platformie Azure	Mikrouslugi, Kubernetes
Integracja	SAP Cloud Integration, API	Power Platform, Azure Services	IFS RESTful API
Warstwa aplikacyjna	SAP Fiori (UX/UI)	Web UI, integracja z Microsoft Power Apps	Web UI

certyfikacji, część dokumentacji szczególnie związanej z architekturą i metodyką jest udostępniana zespołowi wdrożeniowemu przed rozpoczęciem realizacji projektu wdrożeniowego. Architektura **SAP S/4HANA**, **Microsoft Dynamics 365**, oraz **IFS Cloud** z perspektywy technicznej dla instalacji w chmurze pokazuje jak różne zastosowano technologie i przyjęte rozwiązania. Każdy z tych systemów wykorzystuje technologię chmurową, ale inaczej ją implementuje. Pomimo, że IFS Cloud korzysta z chmury np. Microsoft Azure w zakresie instalacji dla środowiska klienta to bazy osadzone są w środowisku Oracle. Każdy z powyższych systemów ma inną architekturę oraz zupełnie różne podejście do zarządzania danymi, integracji czy skalowalności. Porównanie wybranych zagadnień w zakresie architektury i obszarów technicznych zostało przedstawione w tabeli 4.3.

Porównanie metodyki wdrożeniowej systemów ERP, takich jak **SAP**, **Dynamics 365** oraz **IFS**, jest ważne, aby zrozumieć podobieństwa i różnice w podejściu do projektów wdrożeniowych tych systemów. Każdy z dostawców ma własne narzędzia, roadmapy, wzorce i wytyczne dla wdrożeń, które mają na celu wsparcie i efektywną pomoc w procesie implementacji, ograniczając ryzyko i koszty projektu.

- **SAP S/4HANA** (SAP Activate) oferuje dobrze ustrukturyzowaną metodykę łączącą podejście kaskadowe i Agile z rozwiązaniami o charakterze 'best practices', co wspiera wdrożenia projektów w rozproszonej infrastrukturze chmurowej z mocnym wsparciem przede wszystkim dla dużych organizacji. W dużej mierze SAP rekomenduje narzędzia, szablony, czy wytyczne dla partnerów, którzy na potrzeby realizacji własnych wdrożeń adaptują metodykę Activate do swojego środowiska projektowego.

Tabela 4.4. Metodyka wdrożeniowa producentów ERP - porównanie

Aspekt	SAP S/4HANA	Microsoft Dynamics 365	IFS Cloud
Metodyka wdrożeniowa	SAP Activate	Sure Step	IFS Implementation Methodology
Podejście	Hybrydowe (Agile + Best Practices)	Elastyczne (Agile / Waterfall)	Iteracyjne wdrożenia
Fazy wdrożeniowe	6 faz	6 faz	5 faz
Best Practices	Tak, gotowe modele, scenariusze dla branż	Tak, gotowe szablony dla procesów	Tak, gotowe pakiety standardowych procesów
Podejście Agile	Zintegrowane z elementami Agile	Wsparcie zarówno dla Agile lub Waterfall	Zintegrowane z elementami Agile
Dostępność Cloud/On-prem	Cloud jako podstawowe rozw. rynkowe	Cloud, on-premise i hybrydowe	Cloud i on-premise
Integracja z innymi systemami	Tak, zwłaszcza z innymi narzędziami SAP	TAK, Rozwiązania Microsoft i Power Platform	TAK

- **Microsoft Dynamics 365** (Sure Step) proponuje elastyczne podejście zarówno dla projektów o podejściu Agile, jak i tradycyjnym kaskadowym (Waterfall). Dynamics 365 posiada dobrze przygotowaną dokumentację w zakresie integracji ERP z ekosystemem Microsoft i wsparciem dla rozwiązań osadzonych chmurze Azure.
- **IFS Cloud** (IFS Implementation methodology) koncentruje się wokół metodyki kaskadowej i opiera się na realizacji 5 faz projektowych. Od samego początku rekomenduje bardzo duże zaangażowanie użytkowników końcowych. W przeciwieństwie do pozostałych porównywanych metodyk jest zbudowana w oparciu o wdrożenia on-premis z adaptacją do rozwiązań chmurowych, a nie odwrotnie.

Synteza podstawowych zagadnień dotyczących metodyk wdrożeniowych ERP została przedstawiona w tabeli 4.4.

5. Zaproponowana metodyka i fazy projektu wdrożeniowego systemu ERP

W tym rozdziale przedstawiona zostanie autorska propozycja zmian w zakresie działań realizowanych w ramach poszczególnych faz i ich wykorzystania w zależności od typu projektu. Identyfikacja działań projektowych uzależnionych od rodzaju projektu ma na celu usprawnienie procesu wdrożeniowego poprzez:

- automatyczne ograniczenie wymaganych do realizacji zadań projektowych,
- przygotowanie gotowych skryptów zarządczych dla każdego typu projektu,
- opracowanie szablonów projektowych zależnych od typu projektu w zakresie harmonogramu projektowego, podziału obowiązków projektowych oraz rodzaju prac w projekcie,
- optymalizację kosztów wdrożeniowych i czasu trwania projektu.

Ze względu na fakt, że doktorat wdrożeniowy realizowany jest w organizacji, która większość projektów realizuje w oparciu o system IFS, do dalszych prac rozwojowych została wybrana metodyka *IFS Implementation methodology*, która dalej jest nazywana metodyką IFS. Kolejnym bardzo ważnym argumentem jest również fakt, że system IFS Cloud według Gartnera (Rys.4.1) jest obecnie jednym z liderów w klasie systemów ERP dla firm produkcyjnych. Metodyka IFS zaprojektowana została na potrzeby dostarczania IFS Applications - IFS Cloud na podstawie zasad ogólnie obowiązujących standardów dla modeli kaskadowych. Na bazie opracowanej przez producenta oprogramowania metodyki, w ramach tej pracy doktorskiej, została opracowana metodyka wdrożeniowa uwzględniająca wymagane aktywności zależne od typu realizowanego projektu.

Potrzeba dostosowania metodyki do typu projektu wdrożeniowego jest wynikiem doświadczeń wdrożeniowych oraz błędów, które zostały popełnione przez zespół wdrożeniowy [86]. Mocną i jednocześnie słabą stroną standardowych metodyk wdrożeniowych jest ich uniwersalność. Niezależnie czy jest to metodyka producenta wdrażanego oprogramowania, jak metodyka IFS, czy metodyka klasyczna oparta na produkcji

jak Prince2 przedstawia ona w dużej mierze ramy dla projektu. W związku z tym, aby usprawnić prace projektowe, zidentyfikowano działania zarówno z perspektywy operacyjnej jak i zarządczej dla każdego z typów projektu. Działanie to daje możliwość opracowania skryptów zarządczych, które ujednoczą działania zarządcze na projekcie niezależnie od osoby kierownika projektu. Identyfikacja działań zależnych od typu projektu umożliwia również przygotowanie uniwersalnych szablonów projektowych, które pozwolą na standaryzację procesu i sprawniejsze wymiarowanie kosztów oraz czasu projektowego.

5.1. Fazy projektu wdrożeniowego

Standardowa metodyka zakłada, że klasyczne wdrożenie systemu IFS składa się z 5 następujących po sobie faz projektu:

- Przygotowanie Projektu
- Projektowanie Rozwiązania - Przygotowanie Prototypu
- Projektowanie Rozwiązania - Potwierdzenie Prototypu
- Przygotowanie Uruchomienia
- Uruchomienie Produkcyjne

Decyzja o zmianie metodyki według typu projektu wynika z konieczności przygotowania podstawy gwarantującej jednolity charakter realizowanych w organizacji projektów, tak aby był on niezależny od zastosowanych narzędzi oraz zespołu, jaki zostanie przydzielony do jego realizacji. Kluczowym aspektem motywującym do wprowadzenia zmian w standardach metodyki jest konieczność optymalizacji zasobów poświęcanych na wdrożenie. Rzetelne określenie planowanych do wykonania działań, pozwala na świadome zarządzanie portfelem projektowym. Proces ten odbywa się poprzez właściwą alokację zasobów ludzkich, wymaganych narzędzi oraz czasu dla zespołu dostawcy i klienta. **W ramach niniejszych badań wykazano, że przy zastosowaniu dostosowanej do typu projektu metodyki wdrożeniowej łączne nakłady zasobów ludzkich, czas realizacji i ryzyko uległy zmniejszeniu [86]** - jest to powodowane realizacją prac projektowych w oparciu o opracowaną dedykowaną do typu projektu adaptacją metodyki projektowej. Poprzez świadome pominięcie działań, które mają wyłącznie kontrolny charakter, ograniczamy czas na zbędną administrację oraz obniżamy ryzyko ewentualnych nieporozumień z klientem w zakresie podejścia do zarządzania projektem. W ramach pracy badawczej przeprowadzona została analiza danych na przykładzie pięciu firm. Dane zostały zebrane poprzez obserwację i raporty z Komitetów Sterujących, okresowych raportów z postępu prac, dzienników projektowych, badanie

formalnej dokumentacji (minutki, protokoły, CR'ki) oraz wywiady z Kierownikami Projektów klientów, Konsultantami ERP integratorów systemów i Dyrektorami ds. Informatyki (CIO). Wykorzystując głównie informacje zebrane podczas projektów reimplementacji lub upgrade. Ze względu na zróżnicowane zakresy projektów, przeprowadzone badania nie mogły zostać wykonane w oparciu o jednolitą standardową ankietę.

W pozostałych analizowanych projektach harmonogram i zakres wszystkich działań powdrożeniowych został zaplanowany zgodnie ze strategiami CIO. Mechanizmy wdrożeniowe, reimplementacji lub upgrade były dostarczane przez dostawcę systemu ERP, a w analizowanych firmach zastosowano różne podejścia ze względu na brak dedykowanych standardów w momencie wdrożenia. Informacje zostały zebrane przez autora podczas kilku wdrożeń, upgrade, reimplementacji i rolloutu u wszystkich badanej grupy klientów. w odniesieniu do branży, trzy z pięciu firm zajmują się produkcją, a dwie z nich działają w sektorze użyteczności publicznej. Wszystkie badane firmy to duże polskie przedsiębiorstwa korzystające z tego samego systemu ERP o różnej wersji i architekturze. IFS zgodnie z opisem producenta dedykowany jest dla dużych i średnich przedsiębiorstw. Prace badawcze zostały uzupełnione o projekt wdrożeniowy typu rollout funkcjonalny, który został zrealizowany dla 3 przedsiębiorstw oraz rollout techniczny dla dwóch kolejnych przedsiębiorstw. Wszystkie projekty zostały zrealizowane dla jednej grupy kapitałowej.

Metodyka wdrożeniowa IFS niezależnie od zaproponowanych i opracowanych w ramach badań narzędzi, realizowanych tematów czy dostarczanych produktów zarządczych nie ulega zmianie. W zależności od wielkości lub specyfiki danego projektu wdrożeniowego zaproponowane zostały dedykowane scenariusze działań dla poszczególnych typów projektów. Jest to spowodowane tym, że jak każde uniwersalne rozwiązanie metodyka IFS może być zastosowana we wszystkich typach projektów wdrożeniowych, ale ze względu na brak specjalnych scenariuszy do różnych typów projektów wdrożeniowych niesie za sobą wiele ryzyk m. in.:

- niedoświadczony zespół będzie realizował lub podejmował próbę realizacji większości działań projektowych niezależnie od typu projektu,
- niejednoznaczna definicja produktów i działań projektowych może prowadzić do utrudnionej analizy porównawczej tego samego typu projektów,
- w projektach wykonywane są produkty i działania, których zasadność jest nieznająca, a koszty za każdym razem poniesione,
- może prowadzić do wydłużenia prac projektowych, a tym samym obniżenia efektywności projektowej oraz ograniczenia dostępności zespołu w innych projektach.

Klasyfikacja wymaganych do realizacji działań według typu projektu wdrożeniowego została przygotowana w oparciu o metodykę IFS implementation. Producent w proponowanej metodyce zakłada realizację projektu w ramach V faz projektowych niezależnie od rodzaju realizowanego projektu wdrożeniowego.

W tabeli 5.1 przedstawiony został podział na poszczególne fazy projektu wg zaproponowanej klasyfikacji typu projektu, która częściowo została zaproponowana w ramach publikacji [86]. Tabela ta przedstawia kluczowe różnice w zakresie realizowanych fazy i ma za zadanie zobrazować skalę zróżnicowania w zależności od typu realizowanego projektu. Szczegółowy opis definiujący zakres działań dla faz i odstępstwa w różnych typach projektów został opisany w rozdziałach 5.2-5.6.

Tabela 5.1. Fazy projektu według typu wdrożenia. Symbol (+) oznacza, że wszystkie czynności wykonywane są analogicznie jak dla poprzedniej pozycji w wiersza

Fazy	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
Faza I: Przygotowanie Projektu	(+)*	(+)	(+)	(+)	(+)	(+)
Faza II: Projektowanie Rozwiązania - Przygotowanie Prototypu	(+)	ograniczony zakres	nd	ograniczony zakres	nd	(+)
Faza III: Projektowanie Rozwiązania - Potwierdzenie Prototypu	(+)	ograniczony zakres	nd	ograniczony zakres	nd	(+)
FAZA IV: Przygotowanie Uruchomienia - Przygotowanie Rozwiązania	(+)	(+)	(+)	(+)	(+)	(+)
FAZA V: Uruchomienie Produkcyjne	(+)	(+)	(+)	(+)	(+)	(+)

5.2. Faza I - Przygotowanie projektu

Przygotowanie projektu jest fazą obowiązkową niezależnie od skali czy typu realizowanego wdrożenia. Podczas tej fazy realizowane są prace związane z przygotowaniem i rozplanowaniem projektu, kompletowana jest obsada, weryfikowane są zasoby techniczne oraz przygotowywane są szczegółowe plany. Na podstawie zaakceptowanych planów następuje potwierdzenie i uruchomienie zespołu projektowego oraz przygotowanie środowiska projektowego. Powołany Kierownik Projektu opracowuje Dokument Definicji Projektu, w ramach którego potwierdzany jest m.in. zakres, plan i uszczegółowiona metodyka realizowanego wdrożenia. Faza I - Przygotowanie projektu, w głównej mierze ma charakter zarządczy i angażuje przede wszystkim zasoby w roli Kierownika Projektu oraz techników wymaganych do przygotowania infrastruktury oraz instalacji środowiska projektowego i systemowego. Na tym etapie każde z działań ma charakter obligatoryjny i jest realizowane według wymagań. Tabela 5.2 zawiera zestawienie działań występujących w Fazie I dla różnych typów projektów, kolejno uwzględnione są typy: implementacja, upgrade funkcjonalny, upgrade techniczny, rollout funkcjonalny, rollout techniczny oraz reimplementacja. Symbol (+) oznacza, że wszystkie czynności wykonywane są analogicznie jak dla poprzedniej pozycji w wierszu. Użyte w tabeli skróty oznaczają kolejno: PROD - baza produkcyjna, CFG - CONFIG - baza konfiguracyjna, UAT - USER ACCEPTANCE TESTING - baza testowa.

Na podstawie analizy działań występujących w Fazie I projektu można stwierdzić, że dla działań F1.Z1 Zaplanowanie i przygotowanie projektu oraz F1.Z2 Przygotowanie środowiska i infrastruktury faza ta jest identyczna w zakresie zarządczym dla każdego z realizowanych projektów. Zasadnicze różnice dotyczą działania F1.Z3 Przygotowanie i instalacja systemu IFS. Aktywności te można podzielić na trzy grupy:

1. Pełna instalacja ERP w zakresie projektów:

- implementacja
- upgrade funkcjonalny
- reimplementacja (nie zaleca się przeniesienia technicznego bazy na potrzeby prototypowania)

2. Pełna instalacja ERP z przeniesieniem technicznym danych na UAT

- upgrade techniczny
- upgrade funkcjonalny (opcjonalne przeniesienie techniczne bazy na potrzeby prototypowania)

3. Instalacja dodatkowego środowiska UAT w zakresie:

- rollout funkcjonalny
- rollout techniczny

Zakończenie tej fazy w każdym z typowych projektów kończy się spotkaniem otwierającym projekt, odbiorem Dokumentu Definicji Projektu oraz przygotowanym środowiskiem projektowym, umożliwiającym rozpoczęcie kolejnej fazy projektowej.

Tabela 5.2. Faza 1

Działanie	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
F1.Z1 Zaplanowanie i przygotowanie projektu	1. Przygotowanie zespołu wdrożeniowego i przekazanie umowy/zamówienia) 2. Opracowanie i wdrożenie Dokumentu Definicji Projektu	(+)	(+) oraz Kick-off projektu	(+)	(+) oraz Kick-off projektu	(+)
F1.Z2 Przygotowanie środowiska i infrastruktury	1. Przygotowanie infrastruktury (sprzęt/chmura/łącza) 2. Konfiguracja infrastruktury sprzętowej (serwerów) 3. Utworzenie połączenia VPN lub konfiguracja tunelu IP-Sec	(+)	(+)	(+)	(+)	(+)
F1.Z3 Przygotowanie i instalacja systemu IFS	1. Instalacja baz danych Oracle 2. Instalacja IFS Cloud PROD, CFG oraz UAT 3. Aktualizacja IFS Cloud do najnowszej wersji	(+) z ew. możliwością wykonania UAT na bazie poprzedniej wersji	3. UAT aktualizowane z poprzedniej wersji, CFG – REF, PROD.	2. Instalacja dodatkowa UAT	2. Instalacja dodatkowa UAT	(+)

5.3. Faza II - Projektowanie i prototypownie rozwiązania/ Przygotowanie koncepcji biznesowej

Na podstawie przeprowadzonych prac analitycznych, w tej fazie projektowej następuje mapowanie zakresu rozwiązania wymaganego umową lub zamówieniem z listą funkcjonalności standardowych i zidentyfikowanych rozszerzeń funkcjonalnych. Prototyp rozwiązania będzie podstawą dla przygotowania rozwiązania aplikacyjnego planowanego do wykonania w następnej fazie „Przygotowanie Rozwiązania” [8]. W rozdziale została podjęta analiza zmian i uszczegółowienie metodyki dla Fazy II. Zdefiniowane zostały działania i zadania, które muszą zostać wykonane, a których pominięcie nie będzie miało wpływu na zakres i końcowe koszty realizowanego wdrożenia.

5.3.1. Kluczowe różnice według rodzaju projektu

Implementacja:

Na podstawie zrealizowanych projektów, zidentyfikowano nieefektywność związaną z dotychczasową realizacją zaawansowanych szkoleń obszarowych przed przystąpieniem do specyfikacji prototypu docelowego rozwiązania. We wszystkich zrealizowanych projektach wdrożeniowych konieczne było powtórzenie szkoleń w III Fazy projektowej. Ze względu na brak docelowego rezultatu pierwszej tury szkoleń podstawowych zaleca się ich realizację wyłącznie w zakresie przygotowania zespołu do pracy z systemem (podstawowe funkcje systemu, nawigacja i możliwości rozbudowy). Pozostałe działania Fazy II dla projektu wdrożeniowego realizowane są w pełnym zakresie zdefiniowanym w metodyce wdrożeniowej producenta.

Upgrade funkcjonalny:

Dla projektu upgrade funkcjonalny szkolenia wprowadzające mają strategiczne znaczenie, ponieważ traktowane są jako uzupełnienie posiadanej przez użytkowników kluczowych wiedzy, przedstawienie nowych możliwości w zakresie już wykorzystywanych funkcjonalności/procesów. Realizacja zaawansowanych warsztatów szkoleniowych na tym etapie ma na celu wesprzeć zespół projektowy do sprawniejszego przeprowadzenia ewentualnych zmian w zakresie systemowego wsparcia procesów biznesowych oraz działań związanych z benchmarkingiem wybranych procesów biznesowych. Na tym etapie można dokonać zgody na wyłączenie lub uzupełnienie listy procesów, które będą przygotowywane w ramach prototypu zgodnie z ustalonymi zasadami zarządzania zmianą. Istotą tego działania jest wyłączenie z prototypowania procesów o charakterze administracyjnym lub tych, które nie podlegają istotnym zmianom w ramach realizowanego projektu.

Upgrade techniczny, rollout techniczny:

W ramach tego typu projektów rozwiązanie zostało przygotowane i zaprojektowane w ramach innego (wcześniej zrealizowanego) projektu wdrożeniowego. Zespół projektowy nie bierze udziału w definiowaniu zakresu, projektowaniu i przygotowaniu rozwiązania. W tej sytuacji następuje automatyczne przejście do działań w zakresie adaptacji rozwiązania na potrzeby projektu, szkoleń zespołu wdrożeniowego, testów poprawności wykonania upgrade'u oraz przygotowania do uruchomienia.

Rollout funkcjonalny:

Dla projektu rollout funkcjonalny Faza II ma bardzo ograniczony charakter i skupia się wyłącznie na modułach lub procesach, które podlegają uzupełnieniu wobec projektu bazowego lub pozostałych projektów rolloutowych. Przykładem takiego działania jest konieczność realizacji prototypu w zakresie obszaru finansowego (lokalizacja krajowa), rozbudowy systemu o moduł produkcyjny, planistyczny, magazynowy, itp. Każdorazowe odstępstwo od przyjętego wzorca dla rolloutu technicznego w zakresie realizowanych procesów systemowych wymaga prac zgodnie ze zdefiniowanym zakresem Fazy II. Powyższe wnioski zostały zebrane na podstawie zrealizowanych projektów typu rollout techniczny / funkcjonalny. Nomenklatura dzieląca rollouty na dwie grupy projektów (rollout techniczny i rollout funkcjonalny) nie była dotychczas w organizacji wykorzystywana i została wydzielona w oparciu o wnioski ze zrealizowanego projektu wdrożeniowego wykonanego w trakcie prac nad rozprawą.

Reimplementacja:

Projekt reimplementacyjny ma charakter naprawczy względem pierwotnie wykonanego projektu [86]. Ze względu na ten fakt, zakres Fazy II i działania w ramach tego typu projektu są analogiczne do klasycznego wdrożenia z jedną różnicą w zakresie realizowanych szkoleń. Zespół wdrożeniowy powinien zostać przeszkolony w pełnym zakresie funkcjonalnym systemu planowanego do wdrożenia. Ma to kluczowe znaczenie dla właściwego przygotowania zespołu, który w kolejnych krokach fazy będzie odpowiadał za przygotowanie specyfikacji i docelowego prototypu. Na tym etapie identyfikowane są również potrzeby w zakresie migracji danych (czyszczenie, uzupełnienie, naprawa). Prace w zakresie przygotowania danych do migracji mają charakter strategiczny, a ich naprawa prowadzi do znaczącego usprawnienia procesów biznesowych.

Tabela 5.3. Faza 2

Działanie	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
F2.Z1 Rozpoczęcie i szkolenie z podstaw aplikacji (look & feel)	1. Wykonanie spotkania Kick-off 2. Zaplanowanie i wykonanie szkoleń Look and Feel	(+)	nd	(+)	nd	(+) szkolenia w pełnym zakresie dla wdrażanych obszarów
F2.Z2 Przygotowanie Dokumentów Projektu Rozwiązania i Specyfikacja prototypu	(+)	ograniczony zakres - dodatkowe i rozwijane procesy / moduły	nd	ograniczony zakres - dodatkowe i rozwijane procesy / moduły	nd	(+)
F2.Z3 Specyfikowanie zakresu Rozszerzeń Funkcjonalnych	(+)	ograniczony zakres	nd	ograniczony zakres	nd	(+)
F2.Z4 Zdefiniowanie Zakresu Danych	(+)	ograniczony zakres	nd	ograniczony zakres	nd	(+)

5.3.2. Działania projektowe Fazy II

Na Fazę II składają się następujące działania: F2.Z1 Rozpoczęcie i szkolenie z podstaw aplikacji (look & feel), F2.Z2 Przygotowanie Dokumentów Projektu Rozwiązania i Specyfikacja prototypu, F2.Z3 Specyfikowanie zakresu Rozszerzeń Funkcjonalnych oraz F2.Z4 Zdefiniowanie Zakresu Danych. Tabela 5.3 zawiera zestawienie działań występujących w Fazie I dla różnych typów projektów.

F2.Z1 Rozpoczęcie i szkolenie z podstaw aplikacji (look & feel) - W ramach tego działania projektowego realizowane są zadania zgodne z Dokumentem Definicji Projektu. Uniwersalną i obowiązkową aktywnością, która jest wykonywana na każdym projekcie to Kick-off. Nasze doświadczenia projektowe pokazują, że warto realizować spotkanie w dwóch krokach. Pierwszy z nich dotyczy spotkania wewnętrznego i w głównej mierze realizuje cele:

- przekazanie wiedzy pomiędzy zespołami sprzedaży i presales do zespołu wdrożeń,

- przygotowanie zespołu wdrożeniowego w zakresie specyfiki klienta oraz unikalnych cech realizowanego projektu,
- zakres wdrożenia oraz podział obowiązków - odchylenia od standardowej metodyki,
- wewnętrzny model odpowiedzialności RACI w ramach zespołu wdrożeniowego i grup wspierających.

W drugim kroku realizujemy formalne spotkanie z klientem, podczas którego zakładamy realizację:

- przedstawienie ról i zespołu projektowego,
- zakres i harmonogram wdrożenia,
- szkolenie zespołu z metodyki z dostosowaniem do właściwego typu projektu,
- zasady komunikacji i struktura projektowa,
- sposób zgłaszania zagadnień i zarządzania zmianą.

Działanie F2.Z1 Kick-off dla projektów związanych z upgrade i rollout technicznym jest działaniem wykonywanym podczas I fazy projektowej. W przypadku pozostałych typów projektów realizujemy szkolenia o charakterze wstępnym, mającym na celu zapoznanie się zespołu z wdrażanym narzędziem. Z zebranych doświadczeń z realizowanych projektów wynika, że początkowe szkolenia mają charakter wyłącznie poglądowy, a realizacja szkoleń zaawansowanych została przeniesiona na koniec Fazy III.

Działanie F2.Z2 Przygotowanie Dokumentu Projektu Rozwiązania i specyfikacja prototypu jest aktywnością, która rozpoczyna kluczowy dla organizacji krok, polegający na definicji lub doprecyzowaniu zapisów umowy w zakresie:

- mapowanie wymagań klienta na rozwiązania systemowe,
- przygotowanie prototypu kluczowego dla organizacji procesu/procesów E2E(end-2-end) z przykładowymi dla organizacji danymi,
- potwierdzenie uzgodnionego rozwiązania E2E oraz procesów obszarowych w ramach Dokumentu Projektu Rozwiązania.

Każdy zatwierdzony do realizacji projekt wdrożeniowy powinien zawierać zakres prac do wykonania, najlepiej w formie odrębnego załącznika. Ze względu na fakt, że każdy proces pozyskiwania projektu jest unikalny, a poziom szczegółowości zapisów umownych lub zamówienia jest niejednolity, to konieczne jest wykonanie mapowania wymagań

funkcjonalnych na standardowe rozwiązania systemowe. Istotne jest również uświadomienie zespołu wdrożeniowego, że wykonane mapowanie staje się wiążące w zakresie przygotowania docelowego rozwiązania. Realizowane równoległe prace dotyczące analizy stanu obecnego i projektowanego rozwiązania mają charakter wielotorowy, a nadzór nad ich realizacją sprawuje Architekt Rozwiązania. Odpowiada on również za koncepcję z perspektywy między-obszarowej, spójność całego rozwiązania w odniesieniu do procesów wieloodziałowych i między-spółkowych. Przygotowana przez architekta dokumentacja projektowa ma charakter nadrzędny nad dokumentacją obszarową i spaja całość przygotowanego przez zespół wdrożeniowy rozwiązania. W trakcie przygotowania rozwiązania i mapowania wymagań identyfikowane są rozszerzenia funkcjonalne wynikające z braku możliwości zaadaptowania standardowych funkcjonalności zgodnie z wymaganiami użytkownika kluczowego. Na tym etapie zbierane są wszystkie potrzeby użytkowników kluczowych w zakresie CRIM (customizacje, raporty, integracje oraz modyfikacje), których wymagania nie są spełnione przez standardowe rozwiązania systemowe.

Działanie F2.Z3 polega na zdefiniowaniu specyfikacji wysokiego poziomu - HLS (High Level Specification), która ma na celu zgrubne zdefiniowanie zakresu zmiany oraz przyjęcia szacunkowego kosztu robocizny. W projektach wdrożeniowych warto uwzględnić podział w zakresie opieki i odpowiedzialności nad rozszerzeniami:

- customizacje (obiekty własne, eventy, lobby) oraz raporty - zarządzane przez klienta,
- modyfikacje systemowe i integracje (interfejsy) - zarządzane przez integratora.

Powyższy podział jest podyktowany architekturą systemu ERP, która w ramach prac administratora umożliwia realizację dodatkowych pól, zdarzeń systemowych, kokpitów zarządczych i innych obiektów. Jest to obszar działalności rozwojowej, który z założenia nie wpływa na kod źródłowy programu, a jest jedynie jego nadbudową. Natomiast w przypadku modyfikacji wymagana jest ingerencja w kod programu, a zastosowane zmiany mogą mieć wpływ na kluczowe funkcje systemu. W związku z tym zdefiniowany zakres rozszerzeń tworzy nie tylko listę zmian do rozwiązania standardowego, ale również podział odpowiedzialności w zakresie ich ewentualnego wykonania. Po stronie integratora pozostaje odpowiedzialność w zakresie analizy kosztowej i zgrubnego opisu potencjalnych, ale według użytkowników kluczowych koniecznych zmian do wykonania. Każda ewentualna zmiana podlega weryfikacji odnośnie zasadności biznesowej [87] przez architekta projektu i zespół decyzyjny klienta. Komitet Sterujący i sponsor projektu na koniec fazy II dokonuje oceny zasadności biznesowej, wpływu na budżet i harmonogram ramowy projektu zidentyfikowanych zmian do zakresu.

W trakcie II fazy projektowej przygotowywana jest również Strategia migracji danych (F2.Z4), która ma na celu usystematyzowanie zakresu i struktury docelowych danych.

Działanie ukierunkowane jest na zebranie informacji odnośnie stopnia możliwości przygotowania i użycia danych z dotychczas wykorzystywanych systemów, sposobu ich przygotowania oraz metod transferu do docelowej bazy. Wszystkie powyższe działania w różnym zakresie procesowym realizowane są we wszystkich typach projektów z wyjątkiem tych o charakterze technicznym. W pozostałych projektach skala realizacji działań dla Fazy II zależy od ilości przemodelowanych procesów. W przypadku projektów implementacji od podstaw oraz związanych z ponownym wdrożeniem systemu ERP (reimplementacyjnych) wszystkie procesy i działania podlegają wykonaniu w całości. Natomiast w sytuacji realizacji upgrade'u czy rolloutu funkcjonalnego realizujemy koncepcję biznesową, identyfikację zmian i mapowanie danych wyłącznie w zakresie procesów, które zostały wyznaczone do dalszego ewentualnego reengineeringu.

5.4. Faza III - Przygotowanie Rozwiązania / Potwierdzenie Prototypu

Podstawowym celem Fazy III jest przygotowanie i weryfikacja rozwiązania aplikacyjnego. Faza III dla projektu wdrożeniowego systemu ERP jest kluczowa z punktu widzenia kontroli harmonogramu i zakresu projektu. Powstające na tym etapie błędy, pominięte działania lub wygenerowane opóźnienia mają ogromny wpływ na dalsze przygotowanie uruchomienia i start produkcyjny systemu. Na podstawie analizy zakończonych projektów pojawiają się wnioski, że jest to najdłuższa faza z największym zaangażowaniem zasobów ludzkich. Czas trwania waha się od 3 miesięcy dla niewielkich projektów upgrade do 16 miesięcy dla projektów o dużej skali i złożoności rozwiązania. Koszty w zakresie robocizny fazy mieszczą się w zakresie 200-2000 robocizny i są skorelowane z zakresem realizowanych na start produkcyjny CRIM. Podczas realizacji tej fazy uruchamiane są m.in. działania w ramach:

- Parametryzacja/konfiguracja systemu,
- Programowanie zmian,
- Migracja danych,
- Szkolenia zespołów kluczowych,
- Testy jednostkowe w zakresie zmian/rozwoju aplikacji,
- Testy procesowe w obszarach,
- Testy end-to-end i inne wymagane umową lub przyjętym do realizacji rozwiązaniem.

Na tym etapie projektu następuje uruchomienie dodatkowych zasobów zarówno ze strony klienta (wsparcie techniczne dla migracji danych, testów, realizacji szkoleń i dokumentacji, czy przygotowania dodatkowej infrastruktury i środowiska) oraz integratora (programiści, testerzy, wsparcie techniczne dla migracji). Faza III w zakresie metodyki i narzędzi wspierających wdrożenie ERP została w trakcie badań zidentyfikowana jako kluczowa do dalszej optymalizacji, rozbudowy i zmian rozwojowych. Wpływ na konieczność rozwoju zarówno od strony narzędzi jak i metodyki są wynikiem audytu działań organizacyjnych w zakresie:

- Analizy jakościowej dostarczonych w dotychczasowych projektach zmian w oprogramowaniu,
- Pracochłonności w zakresie przygotowania zmian oraz ich podobieństwa pod względem modyfikacji z wcześniej wdrożonych projektów,
- Terminowości prac projektowych w podziale na Fazy i typy projektów,
- Analizy porównawczej działań pod względem typu realizowanego wdrożenia.

Prace optymalizacyjne w zakresie narzędzi wspierających są omówione w rozdziale 6. Podstawowym, oczywistym zadaniem jest właściwe przygotowanie planu prac pod realizację tego etapu. W zależności od wielkości i skali projektu odpowiednie zaplanowanie działań etapu jest krytyczne dla powodzenia projektu wdrożeniowego.

Dla projektów dużych i bardzo dużych, dla których planowany budżet przekracza 1000 roboczodni, głównym wyzwaniem Fazy III jest przygotowanie właściwego podejścia w zakresie realizacji zmian do standardowej wersji systemu oraz odpowiedniego zarządzania realizacją równoległych działań projektowych. W takim przypadku w pierwszej kolejności Architekt Projektu identyfikuje, które zmiany i zadania projektowe muszą być wykonywane w odpowiedniej kolejności ponieważ są uzależnione od siebie nawzajem. Na tej podstawie definiowana jest ścieżka krytyczna dla realizacji zadań projektowych Fazy III. Szacunkowe koszty dla realizacji zadań umożliwiają oszacowanie terminu realizacji w tym zadań na ścieżce krytycznej. Uzyskany w ten sposób szacunkowy termin wykonania najdłuższego łańcucha prac pozwala zweryfikować możliwość realizacji Fazy III w zakładanym terminie. W drugim kroku Kierownik Projektu rozkłada prace na osi czasu dla pozostałych łańcuchów prac przy założeniu:

- średnia ilość poprawek do przygotowanego rozwiązania to 1.6 dla próbki około 500 modyfikacji,
- maksymalna miesięczna ilość realizowanych prac programistycznych to około 100 roboczodni,

- każdy krok wytwarzania zmiany jest oszacowany, a jego estymacja zamieszczona w systemie.

Poniżej przedstawione zostały kluczowe różnice zidentyfikowane przez autora w ramach prac porównawczych poszczególnych działań dla różnych rodzajów projektu wdrożeniowego ERP. Na tym etapie projektu działania związane z realizacją prac nad specyfikacjami, programowaniem modyfikacji, parametryzacją systemu i migracją danych mogą być realizowane równolegle. Powyższe działania w ramach zadań cząstkowych powinny być uruchamiane zgodnie z planem operacyjnym definiującym kroki przygotowania rozwiązania systemowego. Tak przygotowany plan powinien być uzgodniony przez architekta i zatwierdzony przez Kierownika Projektu. Właściwie uzgodniony i monitorowany plan zadań jest podstawą do dalszego właściwego zarządzania zakresem zmian oraz identyfikacją ryzyk projektowych mogących mieć wpływ na finalne terminowe zakończenie prac nad Fazą III. Właściwe zaplanowanie prac w ramach tej fazy jest podstawą dla rozpoczęcia działań, a dalsze prace są uzależnione od jakości przygotowanego planu.

Pierwsza grupa działań Fazy III, które realizowane są równolegle to:

- parametryzacja środowiska testowego,
- specyfikacje Rozszerzeń Funkcjonalnych,
- przygotowanie i testy jednostkowe rozszerzeń funkcjonalnych,
- przygotowanie ról i uprawnień,
- przygotowanie, migracja i testy w zakresie migracji danych.

Po zakończeniu uzgodnionych do realizacji powyższych działań w określonych procesach lub grupach procesów następuje weryfikacja ról i uprawnień oraz testy obszarowe rozwiązania. Przejście do kolejnego kroku powinno nastąpić po potwierdzeniu poprawności działania procesów w poszczególnych obszarach systemu w zakresie:

- danych,
- ról i uprawnień,
- zgodności działania procesów z zatwierdzoną Koncepcją Biznesową / Prototypem.

Następnym krokiem, który powinien zostać zrealizowany po potwierdzeniu poprawności rozwiązania w poszczególnych obszarach, jest finalne szkolenie dla użytkowników kluczowych oraz potwierdzenie scenariusza testów międzyobszarowych. Działania te pozwalają na rozpoczęcie prac w zakresie:

- przeprowadzenie testów akceptacyjnych rozwiązania,
- potwierdzenie ról i uprawnień na potrzeby szkoleń,
- przygotowania instrukcji stanowiskowych.

Kolejne działania związane z przygotowaniem zespołu operacyjnego organizacji, czyli przygotowanie instrukcji oraz przygotowania szkolenia użytkowników końcowych powinny być realizowane po zakończeniu testów akceptacyjnych rozwiązania na bazie wcześniej przygotowanych materiałów szkoleniowych, ról dostępowych i docelowych uprawnień. Szkolenia użytkowników końcowych (Faza IV) powinny być tak zaplanowane, aby nie odbyły się wcześniej niż kilka tygodni przed planowanym startem produkcyjnym systemu. Definicja następstw oraz rozumienie zależności jest podstawą do właściwego ich rozplanowania i kontroli w ramach stałego działania projektowego określanego jako sterowanie etapem, za które odpowiedzialny jest Kierownik Projektu. Przedstawiona poniżej lista działań projektowych Fazy III nie determinuje kolejności ich wykonania. Kolejność działań może zależeć od typu i złożoności projektu, a ich chronologia powinna być zdefiniowana w harmonogramie szczegółowym dla Fazy w ramach realizowanego projektu.

Wykonywane w bieżącej fazie zadania zostały pogrupowane według działań, w ramach których zostały zidentyfikowane różnice wynikające z typu realizowanego projektu. Działanie F3.Z1 Specyfikacje Rozszerzeń Funkcjonalnych jak przedstawia Tabela 5.4 może być realizowane przy założeniu, że w ramach Fazy II zostały zidentyfikowane ewentualne zmiany do standardu systemu i dotyczą projektów:

- implementacja,
- upgrade funkcjonalny,
- rollout funkcjonalny,
- reimplementacja.

Zmiany te dotyczą tak zwanego obszaru CRIM (customizacje, raporty, integracje, modyfikacje), który w zakresie odpowiedzialności jest realizowany przez integratora, z planowanym zakresem zadań do wykonania na start projektu. W ramach pozostałych typów projektów działanie to nie podlega planowaniu i realizacji, ponieważ według założeń w ramach upgrade i rollout technicznego nie planujemy żadnych zmian do wykonania. W tym miejscu nie są podejmowane zagadnienia związane z zarządzaniem zmianą oraz procesem decyzyjnym związanym z wyborem zadań dodatkowych do realizacji. W zakresie mechanizmów wspierających opracowanie i analizę specyfikacji zagadnienia zostaną omówione w rozdziale 6. Na podstawie opracowanego i zaakceptowanego przez klienta i

Architekta Projektu dokumentu specyfikacji modyfikacji realizowany jest finalny proces wyceny kosztowej i przekazanie zadania do realizacji przez zespół rozwoju oprogramowania.

Działanie F3.Z2 jest realizowane na podstawie szczegółowej specyfikacji techniczno-funkcjonalnej w oparciu o uzgodniony proces przepływu informacji pomiędzy: konsultantem, klientem, audytorem, architektem projektu, kierownikiem projektu, architektem technicznym, programistą i testerem. Szczegółowo przygotowany scenariusz zasad przepływu informacji definiuje obieg zadań i proces akceptacji prac i ewentualnych prac rozwojowych. Wykonane konfiguracje administratora, raporty dodatkowe, interfejsy i modyfikacje (CRIM) podlegają testom jednostkowym zarówno na poziomie technicznym jak i funkcjonalnym zgodnie z ustalonym wcześniej scenariuszem działania. Po zakończeniu prac nad wszystkimi zmianami w procesach danego obszaru w ramach tego działania system jest gotowy do szkoleń końcowych i weryfikacji rozwiązania aplikacyjnego (testów między-obszarowych). Równolegle realizowane są zadania związane z parametryzacją środowiska systemowego. Zadania realizowane są przez cały zespół wdrożeniowy zgodnie z księgą reguł. W wewnętrznych opracowaniach zadania związane z konfiguracją środowiska systemowego, których realizacja wykonywana jest przy współdziałaniu klienta nazywane są warsztatami parametryzacyjnymi systemu. Wykonywane zgodnie z harmonogramem operacyjnym Fazy III warsztaty są podstawą do dalszej konfiguracji słowników, ustawień czy procesów według potrzeb klienta. Działanie F3.Z3 Przeprowadzenie warsztatów szkoleniowo-parametryzacyjnych kończy się szkoleniem mającym na celu usystematyzowanie kompleksowej wiedzy i ustawień z zakresu opracowywanego przez zespół obszaru.

Pozostałe zadania w ramach działania F3.Z3 muszą być wykonywane według tego samego typu projektu, jak dla działań związanych ze specyfikowaniem i wykonywaniem CRIM co zostało usystematyzowane w Tabeli 5.4. Zgodnie z planem operacyjnym omawianej fazy równolegle względem powyższych aktywności i w ramach tego samego typu projektów (z wyłączeniem projektów o charakterze technicznym) realizowane są zadania związane z działaniem F3.Z4 Przygotowanie, realizacja i testy migracji. W pierwszym kroku zgodnie z opracowaną strategią przygotowywane są przez zespół klienta dane do migracji. Sposób przygotowania danych jest unikalny dla każdego projektu, ale warto zauważyć, że w ramach klasycznego wdrożenia w zupełności wystarczają standardowe szablony migracji, a największą trudnością dla tego zadania jest właściwe mapowanie obecnie wykorzystywanych i dostępnych danych. Sytuacja wygląda zupełnie inaczej w przypadku każdego innego typu projektu ponieważ dla każdego z nich można zidentyfikować zadania związane z przygotowaniem danych w zakresie:

- automatycznego przeniesienia danych z obecnej wersji systemu, dla procesów nie podlegających rozwojowi lub działających zgodnie z obecnymi wymaganiami,

- pełnego zakresu zgodnie ze standardowymi szablonami przygotowanymi przez integratora, dla procesów nowo wdrażanych lub nie dostępnych w obecnej wersji systemu,
- dopełnieniu dostępnych w systemie danych wynikających z rollout na nową spółkę,
- naprawie danych poprzez automatyczne lub ręczne pozasystemowe ich czyszczenie wynikające z niepoprawnych danych przechowywanych w obecnej wersji systemu.

Na podstawie zakończonych prac związanych z działaniami F3.Z1/F3.Z4 możliwe jest rozpoczęcie prac związanych z działaniem F3.Z5 - przeprowadzenie cykli testów związanych z weryfikacją przyjętego do uruchomienia w poszczególnych obszarach rozwiązania. W opracowanej metodyce dopuszcza się realizację działań F3.Z5 na podstawie zatwierdzonych wyników z działań F3.Z1/Z3 oraz próbki danych czy niepotwierdzonych ról i uprawnień. Wszystkie odstępstwa oraz szczegółowy zakres testów procesów obszarowych powinien zostać zamieszczony w scenariuszach testowych.

Potwierdzone scenariusze testowe są podstawą do przeprowadzenia testów i weryfikacji ich wyników. Zarówno opracowanie scenariuszy testowych oraz ich realizacja powinny być obowiązkowe niezależnie od tego jaki typ projektu wdrożeniowego systemu ERP realizujemy, przy założeniu, że prace testowe dla projektów technicznych realizowane są w ramach Fazy IV. Różnice w zakresie realizacji tego działania pomiędzy projektami dotyczą przede wszystkim skali i poziomu złożoności procesów, a te w dużym stopniu uzależnione są od ilości modyfikacji i integracji oraz złożoności obsługiwanych przez klienta procesów.

Po zakończeniu prac związanych z weryfikacją rozwiązania należy wykonać formalne kroki, które powinny doprowadzić do zakończenia prac związanych z działaniem F3.Z6 - Weryfikacji ról i uprawnień dostępowych. Zadanie dotyczące przygotowania dostępu rozpoczyna się już w Fazie I podczas przygotowania i instalacji środowiska. W zależności od typu projektu uprawnienia są generowane na podstawie szablonych uprawnień lub jeśli jest to zgodne z Koncepcją biznesową wykorzystuje się obecnie funkcjonujące role. W kolejnych etapach pracy nad projektem wdrożeniowym zakłada się imienne konta dostępowe i przypisuje do wcześniej zdefiniowanych ról. Zespół wdrożeniowy klienta przeprowadza weryfikację czy role biznesowe, uprawnienia i profile są właściwie przygotowane. Bardzo ważne, aby proces tworzenia i akceptacji uprawnień został zakończony przed rozpoczęciem szkoleń dla użytkowników końcowych i przygotowaniem dokumentacji użytkownika.

Zwieńczeniem prac nad przygotowaniem rozwiązania obszarowego w każdej płaszczyźnie projektu jest przejście do działania F3.Z7 - Przeprowadzenie globalnych testów akceptacyjnych rozwiązania. Przygotowane i potwierdzone obszarowe rozwiązanie jest podstawą do przeprowadzenia testów między-obszarowych w ramach środowiska UAT. Przed przystąpieniem do tego zadania należy potwierdzić scenariusz testów międzyobszarowych, na

którym zweryfikowana zostanie spójność całego łańcucha zdarzeń end-2-end(E2E). Zespół testowy klienta ma za zadanie zrealizować czynności testowe zgodne z potwierdzonym scenariuszem. Z drugiej strony zespół ma za zadanie zweryfikować rozwiązanie pod względem spełnienia potwierdzonych przez obie strony projektu zakresu wymagań. Po zakończeniu weryfikacji przyjętego rozwiązania zespół wdrożeniowy przechodzi do działań związanych z przygotowaniem użytkowników końcowych do pracy w systemie.

Tabela 5.4. Faza 3

Działanie	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
F3.Z1 Specyfikowanie Rozszerzeń Funkcjonalnych	1. Wykonanie specyfikacji modyfikacji 2. Ustalenie Custom Objects, Quick Reports, Lobby do wykonania	(+)	nd	(+)	nd	(+)
F3.Z2 Przygotowanie i testowanie Rozszerzeń Funkcjonalnych	1. Programowanie uzgodnionych specyfikacji 2. Testowanie wewnętrzne i zewnętrzne modyfikacji oraz instalacja na środowiskach 3. Tworzenie Custom Objects, Quick Reports, Lobby	(+)	nd	(+)	nd	(+)
F3.Z3 Przeprowadzanie warsztatów szkoleniowych dla zespołów	1. Przeprowadzenie szkoleń dla użytkowników kluczowych i administratorów	(+)	nd	(+)	nd	(+)
F3.Z4 Przygotowanie i Testy Migracji	1. Przygotowanie danych do migracji 2. Ustalenie zadań migracji 3. Wykonanie wprowadzenia danych ręcznie 4. Wykonanie 2 iteracji migracji testowej	(+)	nd	(+)	nd	(+)
F3.Z5 Przeprowadzanie cykli testów do Weryfikacji Rozwiązania Aplikacyjnego	1. Opracowanie scenariuszy testowych 2. Realizacja testów	(+)	nd	(+)	nd	(+)

F3.Z6 Weryfikowanie ról i danych	1. Utworzenie pakietów uprawnień testowych 2. Przypisanie uprawnień 3. Testy uprawnień	(+)	nd	(+)	nd	(+)
F3.Z7 Przeprowadzanie testów Akceptacyjnych Rozwiązania	1.Opracowanie scenariusza testów między obszarowych 2.Wykonanie testów między obszarowych przez użytkowników kluczowych	(+)	nd	(+)	nd	(+)

5.5. Faza IV - Przygotowanie rozwiązania

Głównym celem Fazy IV Przygotowanie Uruchomienia jest realizacja zadań związanych z przebrojeniem organizacji w zakresie wymaganym do rozpoczęcia eksploatacji produkcyjnej systemu ERP. Po aktualizacji środowiska testowego o zidentyfikowane podczas testów odchylenia możliwe jest przejście do działania F4.Z1 - Przygotowanie instrukcji stanowiskowych. Zadania te powinny być realizowane przez użytkowników kluczowych, którzy na tym etapie wdrożenia dobrze znają przyjęte rozwiązanie i potrafią je zmapować z zakresem zadań użytkowników końcowych. Dokumentacja użytkownika końcowego, w pełni skonfigurowane środowisko UAT oraz docelowo ustawione konta dostępne są podstawą do realizacji działania F4.Z2 - Szkolenie użytkowników końcowych.

Zarówno działanie F4.Z1, jak i F4.Z2, powinny być realizowane we wszystkich typach projektów. Różnice, jakie zostały zidentyfikowane podczas pracy użytkowników kluczowych to przede wszystkim zakres i czasochłonność realizowanych prac. Modelowym przykładem realizacji kompletnego zakresu prac w tym działaniu jest wyłącznie projekt standardowej implementacji systemu ERP. W przypadku pozostałych typów projektu dokumentacja musi zostać zaktualizowana, a skala zmian jest zależna od:

- długo technologicznego organizacji (reimplementacja),
- skali zmian w obecnie obsługiwanych procesach (upgrade funkcjonalny i rollout funkcjonalny),
- wymagań odnośnie zmian językowych czy dostosowania lokalizacji dla rolloutu (rollouty),

- zmiany wynikające z ewolucji rozwiązania, wynikające z różnic pomiędzy wdrażaną i obecną wersją systemu (upgrade techniczny).

W przypadku działań związanych ze szkoleniami dla użytkowników końcowych zespół wdrożeniowy identyfikuje dwa rodzaje szkoleń:

- od podstaw w pełnym zakresie procesów dla projektów implementacyjnych oraz rolloutów lub nowo wdrażanych procesów,
- szkolenia procesowe, ukierunkowane na różnicowe wynikające ze zmiany wersji systemu oraz zmian w procesach.

W ramach kolejnych kroków fazy odbywa się realizacja działania F4.Z3 Przygotowanie środowiska produkcyjnego. Niezależnie od typu projektu należy obowiązkowo potwierdzić wykonanie środowiska produkcyjnego po zakończeniu prac związanych z Fazą III. System produkcyjny na etapie zawiera wszystkie wymagane na go-live funkcjonalności. W drugim kroku przygotowań środowiska realizowane jest działanie F4.Z4 - *Przeniesienia parametryzacji* do środowiska produkcyjnego, gdzie wykonywane są działania zarządcze związane z budową scenariusza rozruchu produkcyjnego F4.Z5. Obie aktywności są realizowane niezależnie od typu projektu, ale według różnych scenariuszy. Przeniesienie parametryzacji dla projektów implementacji, reimplementacji, rollout funkcjonalny jest realizowana w pełnym zakresie dla wszystkich nowo wdrażanych procesów. Natomiast w ramach projektów o charakterze wdrożenia technicznego użytkownicy kluczowi uzupełniają parametryzację zgodnie ze zidentyfikowanymi odstępstwami pomiędzy wersjami systemu, które zidentyfikowali w ramach działania F3.Z7 (przeprowadzenie testów akceptacyjnych rozwiązania). Bardzo ważnym działaniem w ramach przeniesienia parametryzacji jest uzupełnienie środowiska produkcyjnego o aktualne ustawienia w zakresie ról, uprawnień i kont docelowych użytkowników systemu. Ustawienia związane z docelową konfiguracją urządzeń peryferyjnych takich jak: drukarki, terminale/kolektory WADACO, skanery produkcyjne, połączenia z urządzeniami i systemami zewnętrznymi. Po zakończeniu prac związanych z pełną parametryzacją systemu zaleca się wykonanie kopii środowiska PROD na UAT i weryfikację poprawności definicji instrukcji stanowiskowych oraz dostępów i kont dla użytkowników końcowych. W tym przypadku ewentualnego szkolenia powinny mieć wyłącznie charakter uzupełniający i systematyzujący nabytą w trakcie wdrożenia wiedzę. Bardzo często podczas szkoleń uzupełniających realizowana są prace związane z ponowną weryfikacją i ew. aktualizacją instrukcji stanowiskowych oraz materiałów szkoleniowych dla użytkowników końcowych. Działania z tym związane połączone są często z pre-produkcyjną weryfikacją poprawności działania procesów przez klienta.

Na podstawie przygotowanych przez integratora szablonów Scenariusza Rozruchu, który uzależniony jest od typu projektu, należy przejść do działania F4.Z3 Przygotowanie i

potwierdzenie Scenariusza Rozruchu. Zadania z tym związane powinny zostać zrealizowane w pełnym planowanym do wykonania zakresie niezależnie od typu projektu. Zdefiniowany powyżej zakres prac nie uwzględnia podziału odpowiedzialności w zakresie ich realizacji. W ramach działań optymalizacyjnych nie są brane pod uwagę aspekty formalne w zakresie odpowiedzialności stron. Definicja odpowiedzialności pomiędzy stronami jest podstawowym elementem umowy i z założenia nie ma wpływu na omawiane zmiany w optymalizacji metodyki wdrożeniowej.

Tabela 5.5 zawiera zestawienie działań występujących w Fazie IV dla różnych typów projektów.

Tabela 5.5. Faza 4

Działanie	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
F4.Z1 Przygotowanie instrukcji stanowiskowych z klucowymi użytkownikami	1.Przygotowanie instrukcji stanowiskowych z klucowymi użytkownikami 2.Potwierdzenie ról i uprawnień	(+)	(+) w ramach szkolenia liderów na środowisku UAT	(+)	(+) w ramach szkolenia liderów na środowisku UAT	(+)
F4.Z2 Szkolenia użytkowników końcowych	1.Wykonywanie szkolenia użytkowników końcowych 2.Rozwiązywanie zagadnień projektowych(błędów)	(+)	(+)	(+)	(+)	(+)
F4.Z3 Przygotowanie środowiska produkcyjnego	1.Udostępnienie środowiska produkcyjnego	(+)	II iteracja techniczna UAT	(+)	II iteracja techniczna UAT	(+)
F4.Z4 Przeniesienie parametryzacji do środowiska produkcyjnego	1.Utworzenie firmy i umiejscowienia na PROD 2.Wykonanie parametryzacji na podstawie bazy CFG lub UAT 3.Wykonanie migracji według Strategii migracji	(+)	nd	(+)	nd	(+)
F4.Z5 Przygotowanie Scenariusza Rozruchu	1.Utworzenie scenariusza rozruchu	(+)	(+)	(+)	(+)	(+)
F4.Z6 Podjęcie decyzji o Uruchomieniu Produkcyjnym	1.Wniosek o uruchomienie produkcyjne	(+)	(+)	(+)	(+)	(+)

Po zakończeniu zmian do prototypu szablon projektu jest kopiowany na środowisko testowe. System zostaje zasilony danymi i podlega pre-produkcyjnej weryfikacji. W ramach przygotowań do próbnego przełączenia systemu przygotowywana jest obsada projektu – szkolenia, potwierdzenie ról i uprawnień, ewentualne odchylenia od scenariuszy testowych, instrukcje stanowiskowe oraz materiały szkoleniowe. Zakres tych prac jest ściśle powiązany z rodzajem realizowanego projektu i powinien być dostosowany do potrzeb zespołu.

Aby projekt wdrożeniowy mógł być uruchomiony w sposób bezpieczny, planowy i zgodny z pierwotnymi założeniami należy przygotować szczegółowy scenariusz rozruchu produkcyjnego systemu. Materiał ten musi być przygotowany niezależnie od rodzaju projektu wdrożeniowego i jest kluczowy dla powodzenia całego projektu wdrożeniowego. Kierownik Projektu wraz z Architektem Rozwiązania przygotowują Scenariusz Rozruchu systemu. Dokument definiuje kolejne kroki niezbędne do wykonania w ramach procesu przygotowania uruchomienia, startu i stabilizacji systemu w środowisku produkcyjnym. Szczegółowa lista zadań, określone role oraz dokładne terminy realizacji mają za zadanie zwiększyć przewidywalność całego procesu rozruchowego. Dla dużych projektów proces ten jest rozłożony nawet na okres kilku miesięcy, a dla kluczowych dni zadania przypisane są z dokładnością do konkretnych godzin ich wykonania.

Zadanie musi podlegać weryfikacji niezależnie od rodzaju projektu (dry run). Jeśli Zespół Projektowy zidentyfikował zmiany lub konieczność zmian w infrastrukturze lokalnej, działania te muszą zostać uwzględnione w Scenariuszu Rozruchu. Częstym przykładem takich działań jest wykonanie instalacji lokalnej serwera wydruków (projekty rolloutu lub wdrożenia z wieloma rozproszonymi lokalizacjami), wymiana terminali magazynowych, infrastruktury technicznej czy serwerowej, przejście na chmurowe zasoby serwerowe. Scenariusz uwzględnia wyłącznie te działania, które muszą być zrealizowane w okresie przełączenia systemu. Wszystkie możliwe do wykonania działania wdrożeniowe i testowe w ramach infrastruktury sprzętowej zaleca się wykonać odpowiednio wcześniej przed startem produkcyjnym systemu ERP.

Próba Lokalnego Przełączenia (dry run) ma na celu finalną weryfikację stanu przygotowań systemu do startu produkcyjnego. Scenariusz dla testowego uruchomienia powinien zakładać jak najlepsze odwzorowanie działań produkcyjnych. Głównym zadaniem jest próba rzeczywistego odwzorowania zdarzeń jakie mogą i wystąpią w trakcie finalnego uruchomienia systemu ERP.

W ramach przygotowania uruchomienia Zespół Projektowy powinien przygotować tzw. Super Użytkowników, dla których zakres szkoleń należy ściśle powiązać z rodzajem projektu oraz wiedzą i doświadczeniem. Szkolenia powinny być realizowane przy wykorzystaniu instrukcji stanowiskowych i materiałów szkoleniowych dla użytkowników końcowych. Testowa weryfikacja przełączenia systemu ma na celu również zweryfikować czy dane

produkcyjne zostały właściwie zmigrowane. Na tym etapie interfejsy do systemów zewnętrznych działają w sposób prawidłowy, a na podstawie planu przełączenia jest przeprowadzony test integracji i finalna zgoda na przejście do Testów Gotowości Operacyjnej.

Po przeniesieniu wszystkich ustawień na bazę produkcyjną lub dla projektów technicznych wykonaniu upgrade przeprowadzane są Warsztaty mające na celu finalną weryfikację poprawności działania procesów biznesowych *end2end*. Działania są realizowane z wykorzystaniem danych rzeczywistych, a Wszystkie wykonywane operacje powinny być odzwierciedleniem faktycznej pracy użytkowników końcowych systemu ERP. Prace powinny być wykonywane z wykorzystaniem scenariuszy testowych, a każde odstępstwo powinno być skategoryzowane i właściwie zaopiniowane przed startem produkcyjnym.

Równolegle do wszystkich realizowanych prac realizowane są działania związane z zarządzaniem zmianą. Działanie to jest jedną z podstawowych aktywności w każdym projekcie i dla wszystkich projektów tak samo istotne. Ma to kluczowe znaczenie w momencie, kiedy należy podjąć decyzję o zamrożeniu procesu zmian, przesunięcia prac na okres po starcie produkcyjnym lub post-implementacyjnym,

Po przeprowadzonych testach, analizie zgłoszonych uwag, zastrzeżeń i odchyłeń Komitet Sterujący podejmuje decyzję o Uruchomieniu Produkcyjnym. W przypadku, kiedy zespół nie rozwiązał istotnych zagadnień Komitet Sterujący może zgodzić się na start warunkowy, pod warunkiem, że zastrzeżenia te zostaną wyjaśnione do dnia ostatecznej decyzji dotyczącej startu. W pozostałych przypadkach zakłada się, że otwarte kwestie są pod kontrolą lub zostały rozwiązane,

5.6. Faza V - Uruchomienie produkcyjne

Ostatnia Faza V projektowa powinna być uruchomiona jeżeli poprzednie fazy zostały zakończone pomyślnie. Podstawowe założenie metodyki kaskadowej polega na tym, aby przejść do kolejnej fazy muszą zostać wykonane i potwierdzone przez zespół działania w zakresie poprzedniej fazy projektu. W zakresie zadania *Decyzja o starcie produkcyjnym* wypowiedzieć się musi Komitet Sterujący i to na podstawie wyników prac IV Fazy oraz decyzji KS może zostać podjęta decyzja o starcie systemu.

Zlekceważenie tego działania może doprowadzić do niepowodzenia całego projektu wdrożeniowego. Niewłaściwie przygotowane środowisko produkcyjne, niekompletne testy czy braki w przeszkoleniu użytkowników końcowych mogą w konsekwencji prowadzić do nieplanowanych przestoi w organizacji. Konsekwencją takich działań mogą być wymierne straty finansowe oraz utrata wizerunku, nie tylko w ramach zespołu wdrożeniowego, ale również w otoczeniu całej organizacji. Dlatego też, aby przejść do Fazy V zakłada się, że

próbne przełączenie i testowe uruchomienie jest przeprowadzone, a wyniki tego działania są szczegółowo przeanalizowane przez każdą z ról projektowych w swoim zakresie.

W procesie wdrożeniowym Fazy V identyfikujemy uniwersalne działania, które realizujemy zgodnie z zatwierdzonym scenariuszem rozruchu produkcyjnego. Dokument ten ma za zadanie wesprzeć sterowanie procesem uruchomienia produkcyjnego systemu. Każde z zadań ma określony zakres, termin wykonania i podział odpowiedzialności. W ramach przygotowania do uruchomienia kluczowym zadaniem jest poprawne i terminowe wykonanie działania F5.Z1 - wykonanie migracji produkcyjnej i BO (bilansu otwarcia), które niezależnie od typu projektu powinno być wykonane. Zakres migracji prac jest ściśle związany z wielkością organizacji i zakresem danych do przeniesienia, natomiast złożoność prac w zakresie migracji produkcyjnej i BO jest ściśle powiązany z typem projektu. W zależności od typu projektu w scenariuszu migracji należy uwzględnić sposób wyłączenia obecnie wykorzystywanych systemów i sposób transferu danych operacyjnych lub ich bilans otwarcia w nowym systemie. W ramach działania można wyróżnić pewne kluczowe różnice wynikające z typu realizowanego projektu:

- implementacja, reimplementacja, upgrade funkcjonalny, rollouty
 1. import danych transakcyjnych po zakończeniu prac w obecnym systemie
 2. zakończenie rozliczeń dla transakcji w obecnym systemie i przygotowanie bilansu otwarcia dla tak migrowanych zakresów danych
- upgrade techniczny
 1. zakończenie rozliczeń dla transakcji w obecnym systemie według zakresu uzgodnionego w scenariuszu rozruchu
 2. automatyczne przeniesienie danych w procesie upgrade'u technicznego

Moment potwierdzenie poprawności wartościowo-ilościowej bilansu otwarcia oraz zakresu przeniesienia danych transakcyjnych jest podstawą do rozpoczęcia prac w ramach działania F5.Z2 - Uruchomienie produkcyjne. W dalszym ciągu wszystkie realizowane zadania powinny być wykonywane zgodnie ze scenariuszem rozruchu ze szczególnym uwzględnieniem nowych aktorów w procesie: Użytkowników końcowych. Aby procesy były ze sobą spójne, a rejestrowane dane w systemie właściwe kluczowe jest, aby każda z grup użytkowników realizowała swoje obowiązki systemowe zgodnie z instrukcją stanowiskową oraz za zgodą użytkowników kluczowych.

Projekty wdrożeniowe systemu ERP mają jeden wspólny mianownik, skalę przedsięwzięcia, która często obejmuje wszystkie jednostki organizacyjne firmy oraz setki jednocześnie uruchamianych procesów biznesowych. W przypadku tego typu wdrożeń

kluczowe jest właściwe rozplanowanie zasobów i zakresu prac, tak aby start produkcyjny polegał na wykonywaniu działań zgodnie ze scenariuszem startu produkcyjnego. Największe różnice pomiędzy projektami w ramach tego działania wynikają ze złożoności projektów. Z doświadczeń projektowych jednoznacznie wynika, że każdy projekt, którego zespół wdrożeniowy ma już za sobą doświadczenia w uruchamianiu tak złożonych systemów jest mniej narażony na ryzyko niepowodzenia na tym etapie.

Niezależnie od tego jakie są doświadczenia zespołu wdrożeniowego oraz jaki jest to typ projektu, każdy scenariusz rozruchu produkcyjnego powinien zakładać tzw. *exit plan*, czyli zbiór działań zespołu wdrożeniowego, które muszą zostać wykonane w projekcie, aby przywrócić do działania poprzedni system organizacji. W procesie potwierdzania scenariusza rozruchu kluczowe jest określenie zdarzeń biznesowo-systemowych oraz punktu w czasie, który jeśli nie zostanie dotrzymany powoduje uruchomienie procesu przywrócenia dotychczasowych procesów systemowych. Jeśli ten proces zostanie uruchomiony, projekt zostaje zakończony lub za zgodą Komitetu Sterującego może nastąpić rewizja i ewentualne przejście do działań w zakresie Fazy III w przypadku poprawnego uruchomienia produkcyjnego realizujemy działania F4.Z3 - Świadczenie wsparcia po starcie. Plan asysty rozruchowej jest przygotowany zgodnie z wymaganiami umowy, a jego szczegółowy zakres powinien być gotowy przed przystąpieniem do uruchomienia systemu. Po zakończeniu asysty rozruchowej przechodzimy do działania F5.Z4 - Zamykanie Projektu. Zgodnie z metodyką, w ramach każdego z projektów powinny zostać wykonane analogiczne działania związane z zebraniem nauczek projektowych, rozliczeniem planu projektu i produktów projektu.

Po zakończeniu procesu stabilizacji systemu rozwiązanie jest przekazywane do serwisu (zespołu wsparcia powdrożeniowego), a zespół projektowy zostaje rozwiązany.

Faza V dla projektów upgrade'u technicznego oraz rolloutu technicznego ma ograniczony charakter, szczególnie pod względem skali i zakresu asysty personalnej po uruchomieniu produkcyjnym. Działania w zakresie przełączenia produkcyjnego w przypadku takich projektów również mają bardzo ograniczony charakter i nie niosą za sobą tak dużego ryzyka biznesowego jak to ma miejsce w dużych projektach klasycznego wdrożenia systemu ERP. Niezależnie od rodzaju projektu wszystkie pozostałe działania muszą zostać wykonane zgodnie z planowanym zakresem. Finalnie każdy projekt musi zostać zakończony, oceniony i ewentualnie przekazany do serwisu.

Tabela 5.6 zawiera zestawienie działań występujących w Fazie V dla różnych typów projektów.

Tabela 5.6. Faza 5

Działanie	Implementacja	Upgrade funkcjonalny	Upgrade techniczny	Rollout funkcjonalny	Rollout techniczny	Reimplementacja
F5.Z1 Wykonanie migracji produkcyjnej Bilansu Otwarcia	1.Przygotowanie źródeł danych do migracji Produkcyjnego bilansu otwarcia według ustalonej metody 2.Wykonanie Produkcyjnego Bilansu Otwarcia	(+)i kopia na UAT	podniesienie bazy PROD i kopia na UAT	podniesienie bazy PROD i kopia na UAT	(+)i kopia na UAT	(+)
F5.Z2 Uruchomienie Produkcyjne	1. Rejestracja pierwszych transakcji produkcyjnych	(+)	(+)	(+)	(+)	(+)
F5.Z3 Świadczenie wsparcia po starcie	1. Rozwiązywanie zagadnień zgłoszonych przez użytkowników kluczowych w ramach budżetu wsparcia 2. Wsparcie on-site, jeżeli jest przewidziane umową	(+)	(+)	(+)	(+)	(+)
F5.Z4 Zamykanie etapu	1.Utworzenie protokołu zakończenia etapu 2. Lessons Learnt	(+)	(+)	(+)	(+)	(+)

6. Zaproponowane i wdrożone autorskie rozwiązania i narzędzia wspierające projektowanie i wytwarzanie oprogramowania

W rozdziale opisano zaproponowane i wdrożone przez autora nowe narzędzia oraz zmiany w już istniejących rozwiązaniach organizacyjnych, które opracowane zostały w trakcie przygotowań i prac nad rozprawą. Wszystkie z omawianych rozwiązań zostały podzielone na takie, których użycie jest ściśle powiązane z realizowaną fazą projektu oraz narzędzia, które są uniwersalne, a ich wykorzystanie jest niezależne od realizowanej fazy projektowej. W trakcie prac związanych z doktoratem wdrożeniowym złożoność i ilość realizowanych przez analizowaną organizację projektów uległa wzrostowi o około 150 procent. Z organizacji realizującej równoległe około 5 do 7 projektów wdrożeniowych równoległe przekształciła się w spółkę, która z powodzeniem realizuje od 15 do 20 projektów równoległe. Organiczny wzrost równoległe prowadzonych projektów realizowany był przy jednoczesnych zmianach w ramach narzędzi wspierających zarządzanie i realizację prac, ze szczególnym uwzględnieniem usprawnień wynikających z ewentualnej ich synergii i połączenia w jeden strumień biznesowy. Kluczowe cele, jakie zostały postawione przed rozpoczęciem dalszych prac, obejmowały:

1. Zwiększenie wydajności prac developerskich: Udoskonalenie procesów planowania prac programistycznych powinno prowadzić do zwiększenia ogólnej wydajności operacyjnej całej organizacji i efektywniejszego planowania prac w dłuższym horyzoncie czasu.
2. Poprawa dokładności planowania prac zespołu konsultingowego: Implementacja nowego rozwiązania w zakresie planowania operacyjnego zespołów wdrożeniowych powinna zmniejszyć liczbę konfliktów o zasoby pomiędzy projektami wdrożeniowymi.
3. Redukcja kosztów operacyjnych: Usprawnienie procesów wytwarzania oprogramowania poprzez opracowanie standardowych/universalnych dla wszystkich typów projektów rozwiązań powinno przyczynić się do obniżenia kosztów operacyjnych każdego z projektów.

4. Skrócenie czasu poświęcanego na prace projektowe: Opracowanie narzędzi wspierających przetwarzanie wykonanych w dotychczasowych projektach zmian powinno prowadzić do szybszej realizacji modyfikacji.
5. Podniesienie standardów jakości: Zastosowanie nowej procedury i wsparcia technologicznego procesu powinno poprawić jakość dostarczanego podczas wdrożenia oprogramowania.

Te cele i założenia stanowią podstawę do dalszej analizy i oceny efektywności wprowadzanych zmian. W pierwszej części rozdziału autor opisuje opracowane i zoptymalizowane rozwiązania wspierające cały proces wdrożeniowy oraz narzędzia związane z zarządzaniem portfelem projektów. W drugiej części przedstawione zostały narzędzia z perspektywy poszczególnych faz projektowych.

6.1. Narzędzia nadrzędne względem faz projektowych

W trakcie realizacji wdrożeń systemów zintegrowanego zarządzania, niezależnie od wykorzystywanego oprogramowania dostrzeżone zostały pewne luki systemowe w zakresie:

- dostosowania systemu w obszarze lokalizacji i dostosowań prawnych (otoczenie makroekonomiczne),
- dostosowania do rozwiązań branżowych (otoczenie konkurencyjne),
- automatyzacji procesów,
- nowych rozwiązań.

IFS Cloud razem z platformą Kubernetes służącą do zarządzania, automatyzacji i skalowania nowych aplikacji okazał się doskonałym narzędziem do dalszej ewaluacji. Przyjęta technologia umożliwia:

- Rolling updates i rollback, czyli stopniowe wdrażanie nowych wersji aplikacji bez konieczności przerywania działania poprzedniej wersji aplikacji. Razem z funkcją rollback zapewnia bezpieczeństwo i stabilność w procesie wdrażania nowych rozwiązań i ich aktualizacji w IFS Cloud.
- Izolacja kontenerów, pozwalająca na uruchamianie aplikacji w niezależnych odizolowanych kontenerach. To jest o tyle istotne, ponieważ każdy komponent może być aktualizowany lub testowany niezależnie.

- Obsługa mikrousług, kubernetes opiera się na architekturze mikroserwisów, zapewniając łatwe uruchamianie, monitorowanie i skalowanie aplikacji, które mogą działać jako odrębne usługi.

Między innymi na tej podstawie podjęta została decyzja o rozpoczęciu prac nad uniwersalnymi produktami dostarczonymi dla IFS Cloud.

6.1.1. VAP - Value Added Product - optymalizacja kosztowa

Producenci oprogramowania ERP w tym IFS, aby realizować globalne potrzeby rynku, muszą budować rozwiązania uniwersalne, niezależne od lokalnych wymagań. Przykładem są wymagania prawno-rachunkowe dla każdego z krajów. Doskonałym przykładem jest Polska, gdzie od systemów finansowo-księgowych rynek wymaga wielu wbudowanych funkcji, a globalni dostawcy ERP w większości ich nie obsługują w ramach standardowych komponentów. Wymaganiami takimi są m. in.:

- integracja i wsparcie automatów do wymiany JPK (Jednolity Plik Kontrolny),
- kontrola kontrahenta w zakresie białej listy podatników VAT, na różnych etapach rozliczeń,
- mechanizmy podzielonej płatności,
- integracja z NBP w zakresie kursów walut,
- obsługi i automaty w zakresie naliczania faktur bonusowych i bonusów retrospektywnych,
- integracja i wymiana danych pomiędzy ERP a systemem bankowym,
- obsługa lokalnych wymagań z zakresu naliczeń płac,
- inne.

Na przestrzeni kilkunastu lat, podczas wdrażania projektów ERP z większym lub mniejszym sukcesem realizowane były założenia w zakresie budowania i wykorzystania uniwersalnych produktów dodatkowych. Z założenia takie rozwiązania powinny mieć charakter uniwersalny i niezależny od charakterystyki klienta. W praktyce, niejednokrotnie okazywało się to trudne lub wręcz niemożliwe do zastosowania z z takich względów jak:

- opracowanie uniwersalnego modelu licencyjnego,
- opracowanie standardu w zakresie podejścia do wdrażania rozwiązań mających charakter uniwersalny,

- przyjętego modelu rozwoju i opieki serwisowej tego typu produktów.

Na podstawie realizowanych wdrożeń w przedsiębiorstwach zidentyfikowano, że podstawowym wyzwaniem jest przyjęcie odpowiedniego modelu wdrożeniowego dla uniwersalnych dostosowań systemu ERP. W praktyce okazało się, że aby skutecznie wdrażać tego typu rozwiązania, proces ten należy rozpocząć już na etapie przygotowania projektu do wdrożenia. Istotne jest to, że zespół wdrożeniowy musi znać zakres i zasady licencjonowania wdrażanych dostosowań o charakterze uniwersalnym. Na potrzeby naszych projektów zostało to nazwane w skrócie VAP. Nazwa pochodzi od rozwinięcia „**Value Added Product**” i określa produkty, które przynoszą wartość dodaną poprzez przetwarzanie, ulepszanie lub dostosowywanie do specyficznych potrzeb klientów.

Aby z powodzeniem wykorzystywać i utrzymywać dodatkowe rozwiązania uniwersalne (VAP’y) w ramach projektów wdrożeniowych ERP zostały zidentyfikowane kluczowe punkty kontrolne:

1. przygotowanie umowy

- umowa licencyjna na VAP’y
- zakres wdrażanych VAP’ów

2. przygotowanie projektu

- uświadomienie zespołu wdrożeniowego o zakresie użycia VAP’ów
- wyjaśnienie o braku możliwości modyfikacji VAP, ale rozwój poprzez roadmapy dla produktu

3. projektowanie rozwiązania

- przekazanie do wglądu zespołowi klienta specyfikacji funkcjonalnej VAP.

4. przygotowanie uruchomienia

- instalacja aktualizacji dla VAP oraz monitoring ewentualnego wpływu modyfikacji projektowych na standardowe związania ERP i VAPy

Zdobyte doświadczenie i wnioski wyciągnięte z wcześniejszych działań projektowych wykazały, że rozwiązania VAP powinny być instalowane razem ze standardowym rozwiązaniem ERP. Dzięki temu zespół wdrożeniowy traktuje takie rozwiązania jako komplementarne względem standardowego systemu. Kolejnym bardzo ważnym zadaniem jest nadzór, aby ewentualne potrzeby zmian w zakresie VAP realizowane były poprzez rozwój oprogramowania i były dostępne dla wszystkich pozostałych klientów w ramach aktualizacji

(stawały się standardem w systemie ERP). Takie podejście ogranicza ryzyko, że VAP zostanie przekształcony w modyfikację, która będzie traktowana jak każda inna zmiana mająca unikalny charakter dla konkretnego klienta/projektu wdrożeniowego. W przypadku rozwiązania standardowego, które zostało zmodyfikowane w ramach realizacji konkretnego projektu wdrożeniowego jest ono przekształcane w modyfikację projektową. W takiej sytuacji przestają obowiązywać zasady dotyczące rozwiązań standardowych:

- rozwój produktu zgodnie z roadmapą,
- utrzymanie serwisowe i ewentualne wsparcie w usuwaniu błędów w oparciu o umowę subskrypcji.

Jest to o tyle ważne, że VAP ma na celu jednocześnie ograniczać koszty wdrożeniowe i usprawniać docelowe procesy biznesowe klienta. Natomiast w przypadku ewentualnych zmian w VAP i automatycznej przemianie na modyfikację, cel optymalizacyjny klienta oczywiście zostanie utrzymany, ale przy znacznie większych nakładach na utrzymanie i przyszły serwis funkcjonalności.

W kolejnych punktach przedstawione zostały cele, jakie osiągnie organizacja nie tylko podczas wdrożenia, ale przede wszystkim w trakcie dalszego utrzymania i rozwoju systemu ERP.

1. Ograniczanie Kosztów Poprzez Jednolite Dostosowanie

W systemach ERP zastosowanie VAP polega na dostosowaniu rozwiązań do potrzeb organizacji bez konieczności projektowania i definiowania kosztownych i podobnych do wcześniej opracowanych już rozwiązań. W ten sposób firma korzysta z jednolitego rozwiązania, które na przestrzeni wielu lat i zebranych doświadczeń zostało już dostosowane do standardów branżowych i potrzeb organizacji. Klient unika w ten sposób tworzenia nowych, złożonych i drogich rozwiązań, które za każdym razem niosą za sobą konieczność planowania większych zasobów na przeprowadzenie całej procedury wytworzenia modyfikacji. Jednolite dostosowanie pozwala na wykorzystanie wspólnych zasobów (gotowy dokument specyfikacji, repozytorium kodu), procedur i narzędzi (praktycznie jednolite środowisko klienta IFS Cloud), co znacznie redukuje koszty związane z projektowaniem, wdrażaniem i utrzymaniem VAP'ów.

Dzięki VAP, firmy mogą efektywniej zarządzać kosztami poprzez dalszą standaryzację procesów oraz zmniejszenie nakładów na testy kolejnych aktualizacji.

2. Redukcja Błędów i Ryzyka Poprzez Standaryzację Rozwiązań

Jeżeli przedsiębiorstwo decyduje się na tworzenie nowych zmian do rozwiązań ERP, naraża się na ryzyko popełnienia błędów zarówno na etapie projektowania jak i przygotowania rozwiązania, może to doprowadzić do poważnych problemów operacyjnych i dodatkowych kosztów. Wdrożenie rozwiązań VAP, które opiera się na sprawdzonych, jednolitych rozwiązaniach, znacząco redukuje ryzyko błędów wynikających z indywidualnych modyfikacji systemu. Standaryzacja rozwiązań pozwala na uniknięcie poprawek, testów oraz dodatkowych szkoleń dla użytkowników. Dzięki temu organizacje mogą skupić się na optymalizacji działania systemu i zwiększaniu jego efektywności, zamiast na ciągłym rozwiązywaniu problemów wynikających z błędów lub niedopatrzeń w zakresie poszczególnych modyfikacji.

3. Serwisowanie, utrzymanie i rozwój systemu

Współczesne rozwiązania ERP bazujące na chmurowych scentralizowanych repozytoriach powinny być regularnie aktualizowane, aby spełniać wymagania prawne i politykę serwisową producenta. VAP odgrywa tutaj kluczową rolę, umożliwiając łatwiejsze serwisowanie i utrzymanie systemów poprzez wprowadzenie rozwiązań, które są zgodne z obowiązującymi standardami systemu oraz lokalnymi wymaganiami prawnymi. Standaryzowane produkty są łatwiejsze do aktualizacji i rozwoju w odpowiedzi na zmieniające się przepisy, co ogranicza koszty związane z koniecznością wprowadzania indywidualizowanych zmian. Dzięki VAP, firmy mogą unikać kosztów związanych z ręcznym dostosowywaniem systemów oraz zmniejszyć ryzyko niezgodności z przepisami, co w dłuższej perspektywie przekłada się na większą stabilność operacyjną i finansową.

Podsumowując Value Added Products w kontekście projektów wdrożeniowych ERP to kluczowy element, który pozwala na znaczące ograniczenie kosztów wdrożenia i serwisu poprzez standaryzację kodu, redukcję błędów w ramach podnoszenia wersji oraz uproszczenie procesów serwisowania i utrzymania. Dzięki VAP, firmy mogą efektywniej zarządzać zasobami, minimalizować ryzyko oraz lepiej reagować na zmieniające się wymagania rynku i przepisy prawne, co w ostatecznym rozrachunku prowadzi do zwiększenia efektywności i oszczędności procesów utrzymania ERP. Przykładowe opracowane przez naszą organizację rozwiązania VAP to m. in.:

- Automatykacja procesów rozliczeniowych z instytucjami Finansowymi w Czechach, Polsce, Rumunii czy Słowacji;
- Automatykacja procesów integracyjnych ze spedytorami czy drukiem fiskalnym;
- Moduł Płace, obsługi odzieży roboczej;

- Moduł wspierający obsługę naliczeń i rozliczeń bonusów sprzedażowych;
- Moduły integracyjne z narzędziami do budowy Etykiet, komunikacji EDI, systemów RCP.

6.1.2. Uniwersalna struktura projektu i działań projektowych

Zgodnie z analizą danych historycznych stale zwiększająca się ilość równolegle realizowanych projektów wymusza standaryzację i ujednoczenie struktur, szablonów, modeli tak, aby planowanie, zarządzanie, raportowanie czy nadzór projektów pochłaniał jak najmniej zasobów. Ze względu na 150 procentowy wzrost liczby równolegle realizowanych projektów i pojawiającą się potrzebę Zarządu w zakresie zarządzania portfelem projektów konieczne stało się przeprowadzenie audytu, a w kolejnych krokach ujednoczenie kilku z obecnie wykorzystywanych narzędzi.

Szkielet uniwersalnej struktury projektowej dla wszystkich projektów wdrożeniowych systemu ERP w naszej organizacji został przedstawiony na rysunku Rys.6.1.

Na tej podstawie definiowane są obecnie struktury kolejnych projektów wdrożeniowych dla wszystkich typów projektu. W związku z tym podejściem nie ma możliwości, aby projekt realizowany zgodnie z przyjętą przez naszą organizację metodyką zakładał inne niż powyżej przedstawione fazy projektowe. Wynika to z konieczności właściwego powiązania:

- planów budżetowych i zasobowych z etapu sprzedaży – punkt odniesienia dla utylizowanych kosztów wdrożeniowych,
- struktury zasobów ludzkich – analiza zapotrzebowania i utylizacji zgodnie z kompetencjami projektowymi,
- operacyjnych zadań projektowych – zadania wynikają z podziału obowiązków pomiędzy stronami i zakres realizowanego wdrożenia,
- planu przychodów względem ponoszonych kosztów – dla projektów Fix price (ustalony stały budżet na cały projekt przed jego rozpoczęciem) harmonogram płatności powinien być skorelowany z działaniami projektowymi.
- Power BI na potrzeby raportowania.

Activity	Resources	Role Assignments	Connections	Tasks		
Activity ID: P2.1 Activity Description: Solution modelling						
Responsible ID: ROMEPL Responsible Name:						
Calculated Cost Progress: 0,00% Calculated Hours Progress: 0,00%		Planned Cost Driver: Progress Method:				
Start/Finish Dates						
Early: 28.08.2024 08:00 Actual:		27.09.2024 17:00				
Late:		Total/Remaining Work Days: Days: Elapsed Work Days:				
Baseline:						
Cost	Revenue	Hours	Estimate	Activity Classes	Details	
Cost Element	Cost Element Description	Estim... Hours		Planned Hours	Baseline Hours	Earned
CALC...	CALCORE	24,00		16,00		
▶ FINA...	FINASSETT	24,00		6,00		
FINCO...	FINCORE	32,00		13,00		
FINCTL	FINCTL	16,00		7,00		
INVCO...	INVCORE	32,00		17,00		
Σ		128,00		59,00	0,00	

Rys. 6.2. Struktura kosztowa projektu

Company Name:	Base Currency:	Status:	Financial Project Exists						
	PLN	Initialized	<input type="checkbox"/>	<input type="checkbox"/>					
Valid Report Codes		Multi-Company Reporting		Activity Hours	Documents	Document Transmittals			
Activity	Resources	Role Assignments	Connections	Tasks	Demand	Supply	Work Orders	Planning	Analysis
Module	Responsible	Cost Element	Task ID	Task Name	Task Info	Task Start Date	Plann... Hours	Completed Date	
MAGCORE	IGDMP	MAG	100000364	Utworzenie magazynów	podziel logiczny	01.07.2024	2		
MAGCORE	IGDMP	MAG	100000385	MAG-08 Wadaco - wymagane proces...	Przyjęcie dostawy...	01.07.2024	2		
MAGCORE	IGDMP	MAG	100000386	MAG-11 Obrót opakowaniami zwrotny...	jak ma wyglądać u...	01.07.2024	2		
MAGCORE	IGDMP	MAG	100000387	MAG-01a Obsługa magazynu Wyrób...	struktura organiza...	01.07.2024	4		
MAGCORE	IGDMP	MAG	100000388	MAG-01b Obsługa magazynu Materiał...		01.07.2024	3		
MAGCORE	IGDMP	MAG	100000389	MAG-01c Obsługa magazynu Konasyg...	Dostęp do magazy...	01.07.2024	2		
MAGCORE	IGDMP	MAG	100000390	MAG-01d Obsługa magazynu u dosta...	Utworzenie lokaliz...	01.07.2024	2		
PRDCORE	ROMEPL	PRD	100000436	PRD-01 - Dane podstawowe	Parametryzacja u...	01.07.2024	5		
PRDCORE	ROMEPL	PRD	100000437	PRD-03 - Struktury produkcyjne	Struktury produk...	01.07.2024	3		
PRDCOST	ROMEPL	PRD	100000456	PRD-05 - Kalkulacja kosztów	Układy kosztów	01.07.2024	5		
PRDCOST	ROMEPL	PRD	100000457	PRD-05 - Kalkulacja kosztów	Modele kosztowe	01.07.2024	2		
PRDCOST	ROMEPL	PRD	100000458	PRD-05 - Kalkulacja kosztów	Kalkulacja kosztów	01.07.2024	7		

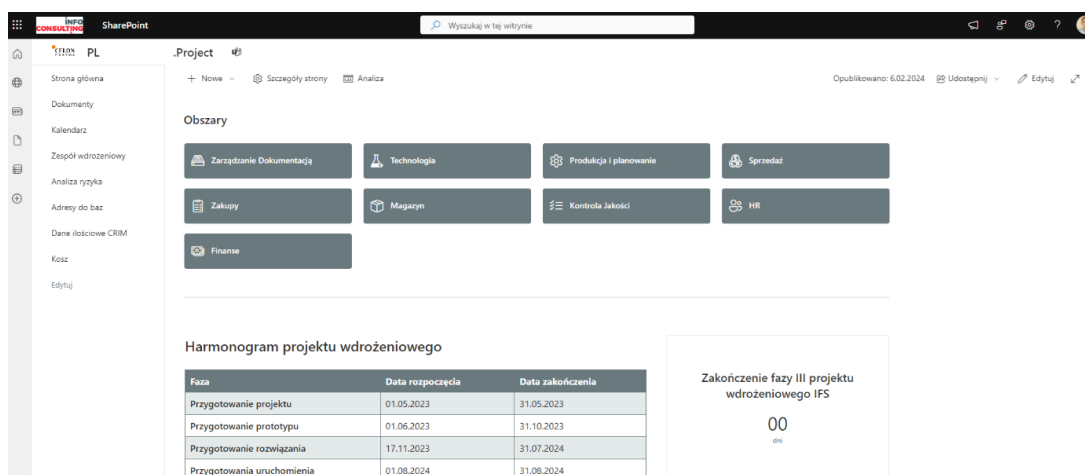
Rys. 6.3. Struktura zadań w projekcie

Podsumowując, jednolita budowa struktury projektowej daje ogromne możliwości w zakresie śledzenia projektu od etapu pozyskiwania (sprzedaży) do przekazania pod opiekę serwisową/utrzymanie. Dla mniej doświadczonych Kierowników Projektu oraz członków zespołu wdrożeniowego szablonowa struktura projektu oraz modelowy zbiór działań i zadań do realizacji w projekcie są punktem odniesienia dla aktywności, które muszą być wykonane

podczas realizacji wdrożenia. W procesie *End 2 End* bardzo ważne jest, aby mieć możliwość ustandaryzowanego schematu monitorowania ewentualnych odchyłeń na wszystkich projektach wdrożeniowych. Uniwersalne struktury w zakresie planowania i rozliczania czasu pracy, kontroli przychodów oraz kosztów dotyczą przede wszystkim projektów Time and Materials (TM). Stosowanie się do przedstawionych powyżej struktur pozwala na osiągnięcie:

- Analizy odchyłeń od planu w zakresie czasu pracy na poziomie:
 - Projektu
 - Modułu – komponentu
 - Grupy zasobów
 - Działania projektowego
- Analizy odchyłeń planu płatności oraz ewentualnej produkcji w toku
- Analizy odchyłeń planowanego i rzeczywistego harmonogramu projektowego

Repozytorium projektowe Wspólnym narzędziem do współpracy całego zespołu projektowego jest w przede wszystkim SharePoint oraz MS Teams. Synchronizacja obu tych narzędzi umożliwia sprawną komunikację i dostęp do aktualnego repozytorium projektowego. W naszej organizacji narzędzia te służą jako podstawa do systematycznej kontroli jakości i monitoringu zgodności realizowanych prac i wykonywanych produktów projektowych względem metodyki wdrożeniowej.



Rys. 6.4. Repozytorium projektu

Aby sprawnie realizować procesy zapewnienia i kontroli jakości, struktura repozytorium powinna być analogiczna w każdym z realizowanych przez organizację projektów wdrożeniowych. Podobnie jest w przypadku zespołów projektowych, dla których działania

związane ze sprawnym planowaniem, raportowaniem spotkań i ewidencjonowania materiałów oraz dokumentacji projektowej nie powinny być utrudnieniem. Dlatego tak ważne jest, aby struktura była w takim samym układzie, niezależnie od realizowanego projektu. Na rysunku Rys.6.4 przedstawiona została przykładowa witryna projektowa usprawniająca nawigację w projekcie m.in. w zakresie: harmonogramu i zbliżających się spotkań, dokumentacji, rejestrach ryzyk i zagadnień.

6.2. Faza I Przygotowanie projektu

W ramach realizowanej pracy i działań związanych ze wsparciem prac dotyczących sterowania etapem przygotowania projektu skoncentrowano się na ujednoczeniu już wykorzystywanych narzędzi, które są niezbędne do efektywnego zarządzania projektami. Aby osiągnąć spójność i zwiększyć efektywność zarządzania, stworzono zestaw szablonów projektowych, które wspierają różne typy projektu. Przedstawiono opis kluczowych narzędzi i produktów ujednoczonych podczas prac nad zmianami w tej fazie projektowej w zakresie:

- Dokument Definicji Projektu
- Przejęcie projektu wdrożeniowego – Kick-off wewnętrzny
- Karta projektu
- Szablony dokumentacji projektowej

Dokument Definicji Projektu Jednym z fundamentalnych narzędzi przygotowanych i potwierdzanych w ramach tej fazy jest Dokument Definicji Projektu. Ten dokument jest kluczowy, ponieważ zawiera szczegółowy opis wymagań dla wszystkich typów projektów, które organizacja może realizować. Główne elementy Dokumentu Definicji Projektu:

- **Opis projektu:** Szczegółowy opis celu i znaczenia projektu dla organizacji, w której projekt będzie realizowany. Powinien on odpowiadać na pytanie, dlaczego projekt jest realizowany oraz jakie korzyści przyniesie organizacji.
- **Zakres projektu:** Dokładne określenie granic projektu, w tym co jest, a co nie jest elementem składowym projektu. Wyznaczone zostają między innymi wdrażane obszary, integracje z systemami zewnętrznymi, instalowane komponenty aplikacji oraz zamówione licencje. Wskazanie wyraźnych granic definiuje oczekiwania interesariuszy oraz pozwala na sprawny proces identyfikacji rozszerzeń zakresu projektu.
- **Cele projektowe i mierniki sukcesu:** Wyraźnie zdefiniowane cele oraz kryteria sukcesu. Mierzalne cele pomagają w ocenie postępów i końcowej weryfikacji

skuteczności realizacji projektu. Przed rozpoczęciem projektu wyznaczane są punkty kontrolne i kamienie milowe, które mają na celu weryfikację statusu zaawansowania projektu po zakończeniu poszczególnych etapów projektów.

- **Zasoby i budżet:** Wymagane zasoby ludzkie, technologiczne oraz finansowe. Szczegółowy opis zasobów niezbędnych do realizacji projektu, w tym zaangażowanie zespołu, sprzętu oraz narzędzi.
- **Ryzyka i założenia:** Identyfikacja potencjalnych ryzyk i kluczowych założeń w zakresie komunikacji, struktury projektowej, harmonogramu i kamieni milowych. Analiza ryzyk oraz przygotowanie planów zarządzania nimi są kluczowe dla przewidywania i minimalizowania problemów, które mogą pojawić się podczas realizacji projektu.

Dokument ten zapewnia, że wszyscy interesariusze mają jasność co do celu, zakresu i oczekiwanych rezultatów projektu, co jest kluczowe dla jego sukcesu. Jego standaryzacja ma za zadanie ułatwić porównywanie i ocenę różnych typów projektów w organizacji.

Kick-off Wewnętrzny Drugim kluczowym narzędziem w zakresie przygotowania projektu od strony zarządczej jest schemat opracowany dla Kick-off Wewnętrznego. Zadania związane z tym działaniem wymuszają na Zespole wdrożeniowym obowiązek przejścia projektu. Formalizacja działania nakreśla zakresu przejmowanych prac od zespołów Pre-sales/sales do zespołu wdrożeniowego. Główne elementy Kick-off Wewnętrznego:

- **Cel spotkania:** Formalne przejście projektu przez zespół wdrożeniowy po etapie sprzedaży, przedstawienie celów projektowych. Wskazanie, jakie wewnętrzne cele mają zostać osiągnięte w ramach realizacji projektu wdrożeniowego.
- **Zakres przejmowanych prac:** Opis prac, które są przekazywane od zespołu sprzedażowego do zespołu wdrożeniowego. Jasne określenie zakresu przejmowanych prac minimalizuje ryzyko nieporozumień i luk informacyjnych.
- **Role i odpowiedzialności:** Wyznaczenie odpowiedzialności za poszczególne zadania w projekcie. Każdy członek zespołu musi znać swoje obowiązki i odpowiedzialności.
- **Harmonogram:** Wstępny harmonogram z kluczowymi terminami i kamieniami milowymi. Harmonogram powinien obejmować wszystkie etapy projektu, od planowania po realizację i zamknięcie.
- **Zarządzanie zmianą:** Przedstawienie sposobu ewidencji i zarządzania zagadnieniami i zmianą w projekcie.

Wprowadzenie obowiązku projektowego w tym zakresie zapewnia płynne przejście prac między zespołami i minimalizuje ryzyko nieporozumień. Formalizuje proces przekazania odpowiedzialności, co jest kluczowe dla zapewnienia ciągłości i efektywności projektu.

Karta Projektowa Karta Projektowa jest kolejnym istotnym narzędziem, które definiuje cele stawiane przed zespołem i kierownikiem projektu. Jest to dokument, który formalizuje rozpoczęcie projektu i nadaje mu oficjalny status.

Główne elementy Karty Projektowej:

- **Cele projektu:** Konkretnie, mierzalne cele, które projekt ma osiągnąć.
- **Zakres projektu:** Definicja projektu oraz szczegółowe wyjaśnienie, co zostanie dostarczone w ramach projektu jako zestaw funkcjonalności, które są następnie potwierdzane na etapie analizy wdrożeniowej
- **Interesariusze:** Identyfikacja kluczowych interesariuszy i ich ról w projekcie.
- **Budżet:** Przybliżony budżet projektu, wraz z alokacją zasobów finansowych.
- **Harmonogram:** Kluczowe daty i kamienie milowe projektu.
- **Ryzyka:** Identyfikacja potencjalnych ryzyk i planów ich zarządzania.

Karta Projektowa jest niezbędnym dokumentem, który zapewnia jasność i kierunek dla całego zespołu projektowego.

Szablony Dokumentacji Projektowej Ostatnim kluczowym elementem w ramach przygotowania narzędzi kierownika projektu są Szablony Dokumentacji Projektowej. Te szablony są zaprojektowane w taki sposób, aby ułatwiać standaryzację i ujednoczenie dokumentacji w całej organizacji.

Rodzaje Szablonów Dokumentacji Projektowej:

- **Plan projektu:** Szczegółowy plan działania, który obejmuje harmonogram, alokację zasobów, budżet oraz zarządzanie ryzykiem.
- **Raporty postępu:** Regularne raporty, które dokumentują postępy w realizacji projektu, identyfikują problemy i proponują działania korygujące.
- **Protokoły spotkań:** Standardowe formaty protokołów, które ułatwiają dokumentowanie decyzji i ustaleń z poszczególnych spotkań projektowych.
- **Raport końcowy:** Podsumowanie całego projektu, w tym osiągniętych celów, wydajności, wyników oraz wniosków na przyszłość.

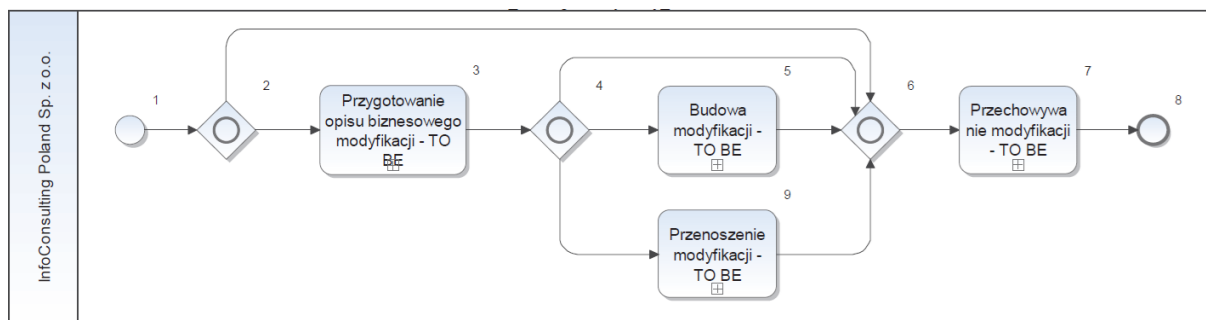
Te szablony pomagają zapewnić spójność, przejrzystość i efektywność w zarządzaniu dokumentacją projektową.

6.3. Faza II Projektowanie i prototypowanie rozwiązania/ Przygotowanie Koncepcji biznesowej

Faza II rozpoczyna proces mapowania wymagań biznesowych na wymagania funkcjonalne systemu. W trakcie tych działań konsultanci wdrożeniowi identyfikują potrzeby związane ze zmianami do standardowego rozwiązania systemowego. Rozszerzenia do standardu definiujemy jako Customizacje, Raporty, Integracje i Modyfikacje (CRIM). Zgodnie z naszą wewnętrzną procedurą wytwarzania modyfikacji, za modyfikację uważa się zmiany programistyczne podlegające dalszej opiece po wdrożeniowej i utrzymaniu. Każda z takich zmian powinna posiadać specyfikację funkcjonalną, która jest podstawą do dalszych prac związanych z programowaniem, testowaniem, odbiorem czy monitoringiem ewentualnych zmian w przypadku rozwoju standardu systemu ERP. Wszystkie opracowane przez nasz zespół wdrożeniowy specyfikacje przechowywane są w repozytorium zgłoszeń (Service Desk). W archiwum znajdują się specyfikacje wykonane dla większości wdrażanych przez naszą organizację projektów wdrożeniowych odnoszących się do różnych wersji systemu ERP. Ze względu na fakt, że zespół wdrożeniowy stale się powiększa, a przez to bardzo ciężko jest w sposób niesystemowy wesprzeć proces identyfikacji modyfikacji, mających znamiona podobnej do wykonanej już w przeszłości pojawiła się potrzeba wykonania wyszukiwarki.

Ze względu na złożoność przedsięwzięcia zdecydowaliśmy o realizacji narzędzia przy wsparciu dodatkowych środków i zasobów, które pozyskaliśmy w ramach realizacji projektu badawczo-rozwojowego o numerze POIR.01.01.01-00-0960/19. Prace nad narzędziem były realizowane równolegle do prac nad rozprawą, a doktorant w ramach tego projektu był odpowiedzialny za prowadzenie prac projektowych, wsparcie projektowania, testy oraz adaptację w środowisku produkcyjnym. Zakres wsparcia dotyczy zagadnień związanych z obsługą modyfikacji i rozpoczyna się od zidentyfikowania potrzeby zmiany do standardu systemu. Po zakończeniu weryfikacji alternatywnych rozwiązań systemowych rozpoczyna się ewentualny proces opisu biznesowego modyfikacji. Jak przedstawiono na poniższym rysunku (Rys. 6.5), przygotowany uproszczony opis modyfikacji służy do przeszukania repozytorium zgłoszeń i plików w zrealizowanych projektach. Na podstawie otrzymanych wyników zespół wdrożeniowy może podjąć decyzję w zakresie:

- Przeniesienie modyfikacji – zgodność potrzeby klienta z wcześniej wykonaną modyfikacją jest na poziomie akceptowalnym przez klienta,



Rys. 6.5. Proces budowy modyfikacji

- budowa nowej specyfikacji i modyfikacji w przypadku zgodności dostępnych modyfikacji na poziomie poniżej $< 50\%$,
- budowa specyfikacji z przeniesieniem wcześniej wykonanego rozwiązania. Adaptacja modyfikacji w przypadku zgodności dostępnych modyfikacji na poziomie $50\% - 80\%$ zgodności z dostępną modyfikacją.

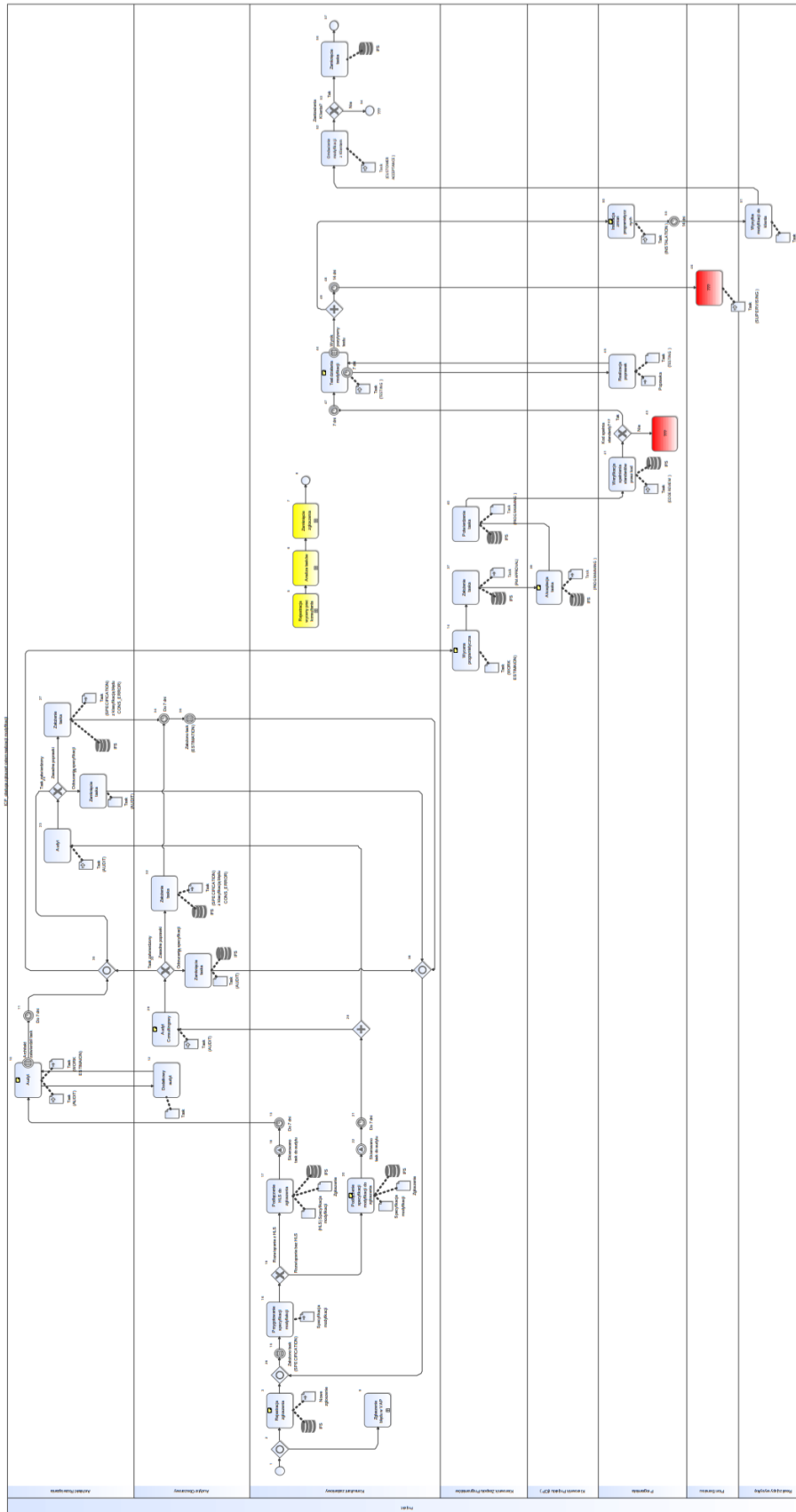
Wytyczne te mają zastosowanie tylko wtedy, kiedy Architekt Projektu wdrożeniowego nie zdecydował inaczej.

6.3.1. Aplikacja InfoSearch

Automatyczna klasyfikacja i grupowanie dokumentów zyskuje na popularności, efekty doskonale ilustruje znaczna liczba publikacji naukowych podejmujących omawianą tematykę. Automatyczna kategoryzacja tekstu polega na powiązaniu dokumentów do wcześniej zbudowanych kategorii tekstowych z wykorzystaniem algorytmów uczenia maszynowego. Ograniczeniem dla tego podejścia jest potrzeba posiadania znacznej ilości danych treningowych z manualnie oznakowanymi kategoriami. Budowanie oznaczonego zestawu danych jest dość trudne i czasochłonne. Natomiast grupowanie analizuje treść plików tekstowych, analizując ich podobieństwo i grupując podobne dokumenty razem.

Opracowanie koncepcji docelowego procesu w ramach projektowania i produkcji modyfikacji ERP, stworzono opis procesu w metodologii BPMN 2.0 (Rys.6.6). Następnie, w drodze inżynierii procesowej, przygotowano model referencyjny docelowych procesów, uwzględniając wsparcie zapewniane przez narzędzie opracowane w ramach projektu.

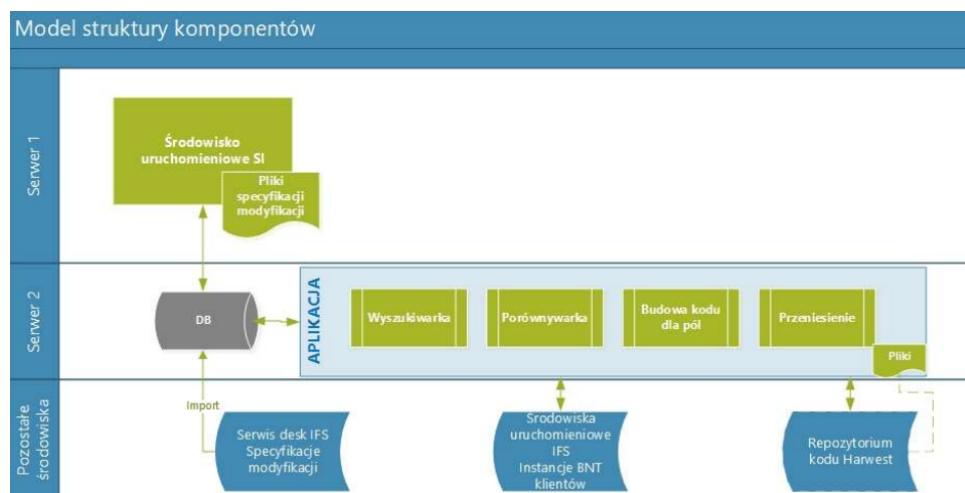
Na podstawie analizy procesów biznesowych (Rys.6.6) wyodrębniono i opisano następujące procesy związane z projektowaniem, programowaniem i wdrażaniem modyfikacji systemu IFS zgodnie z wymaganiami standardów jakości IFS Applications:



Rys. 6.6. Proces obsługi projektowania i programowania modyfikacji

- Przygotowanie opisu biznesowego modyfikacji (zarówno HLS - High level specification jak i DLS - Detailed level specification) – w zakresie wyszukiwania modyfikacji o podobnym zakresie.
- Tworzenie modyfikacji – w zakresie automatycznego generowania pól i ich kodu.
- Transfer modyfikacji między instalacjami – w zakresie oceny wpływu modyfikacji na system.
- Przechowywanie modyfikacji – w zakresie automatyzacji porównywania modyfikacji.

W ramach automatycznej klasyfikacji dokumentów jest wspierany proces związany z przygotowaniem opisu biznesowego modyfikacji. Wykonana została wyszukiwarka, w ramach której możliwe jest porównywanie specyfikacji HLS i DLS modyfikacji oparte na wynikach algorytmów klasyfikacyjnych. Przyjęte rozwiązanie pozwala na weryfikację w zakresie dostępności modyfikacji/specyfikacji wykonanych dla innego klienta, która mogłaby zostać zaadaptowana w projekcie częściowo lub w całości, tym samym ograniczając koszty projektowe potrzebne do napisania DLS i wykonania żądanej modyfikacji [88].



Rys. 6.7. Model struktury komponentów InfoSearch

Opracowane rozwiązanie zakłada współdziałanie z obecnie wykorzystywanymi przez organizację narzędziami, a ogólne zasady integracji przedstawiono na rysunku Rys.6.7. Głównym źródłem danych dla porównań i wyszukiwarki są dane pozyskane z systemu Serwis Desk, natomiast dla repozytorium kodu jest Harwest. Aplikacja składa się z komponentów realizujących porównywanie dokumentów/specyfikacji na podstawie technik przetwarzania języka naturalnego (NLP), przeszukiwania repozytorium w celu znalezienia podobnych specyfikacji modyfikacji. Wprowadzając do wyszukiwarki frazę „Zamówienie zakupu” otrzymujemy rezultaty związane z faktycznym zamówieniem zakupu, niezależnie od

The screenshot shows the 'Wyszukiwarka' (Search) interface of the InfoSearch application. The search term 'zamowienie zakupu' is entered in the search field. The results table below shows four entries with their respective IDs, titles, and counts.

LP	Id zgłoszenia	Tytuł zgłoszenia	Zgodność	Pliki
	1882305	ZAK_06_16 Zmiany w mechanizmie tworzenia Zamówienia zakupu z zapytania ofertowego	32	51
	1882300	ZAK_06_14 Mechanizm tworzenia zapytania ofertowego z linii Zapotrzebowań zakupu, wraz z funkcjonalnością dodania do istniejących zapytań.	32	79
	1882353	ZAK_12_6 Automatyczne przetwarzanie zamówień zakupu na aktywboksy	28.000000000000004	46
	1882292	ZAK_05_3.4 Mechanizm generowania listy zakupowej w oparciu o plan asortymentowy	19	162

Rys. 6.8. Interfejs aplikacji InfoSearch

popętnionych błędów czy nie właściwie zaewidencjonowanych specyfikacji modyfikacji „...zamówień zakupu...” (Rys.6.8).

Wyniki uzyskane w trakcie prac testowych na dokumentach specyfikacji modyfikacji dla systemu ERP IFS zostały pogrupowane według obszarów funkcjonalnych systemu. W ramach prac testowych zebrano 3144 dokumenty specyfikacji, a wyniki grupowania i podobieństw zostały przedstawione zostały w Tabeli 6.1.

Największa liczba dokumentów oraz klastrów jest związana z dwoma modułami funkcjonalnymi systemu ERP: Finanse oraz Dystrybucja. Największe różnice w liczbie klastrów występują w module Dystrybucji, gdzie mamy bardzo różnorodnie tematycznie modyfikacje. W module Finanse modyfikacje są w znacznym stopniu zbieżne, wiele z nich dotyczy bardzo podobnej materii i koncentruje się wokół zmian prawnych, które dla wielu klientów były przygotowywane w ramach modyfikacji systemu.

W większości przypadków lepsze wyniki uzyskano za pomocą algorytmu AP (Affinity Propagation). Znalaziono dużą liczbę klastrów z pojedynczymi dokumentami, które dotyczyły specyficznych modyfikacji dla klientów. Niezależnie od przyjętego algorytmu w każdym module znalazły się podobne modyfikacje, a w niektórych modułach bardzo zbieżne, co daje podstawę do ewentualnego przeniesienia modyfikacji do innych klientów z niewielkimi zmianami.

Tabela 6.1. Liczba klastrów według dwóch metod oraz maksymalny rozmiar klastra według obszarów funkcjonalnych [88]

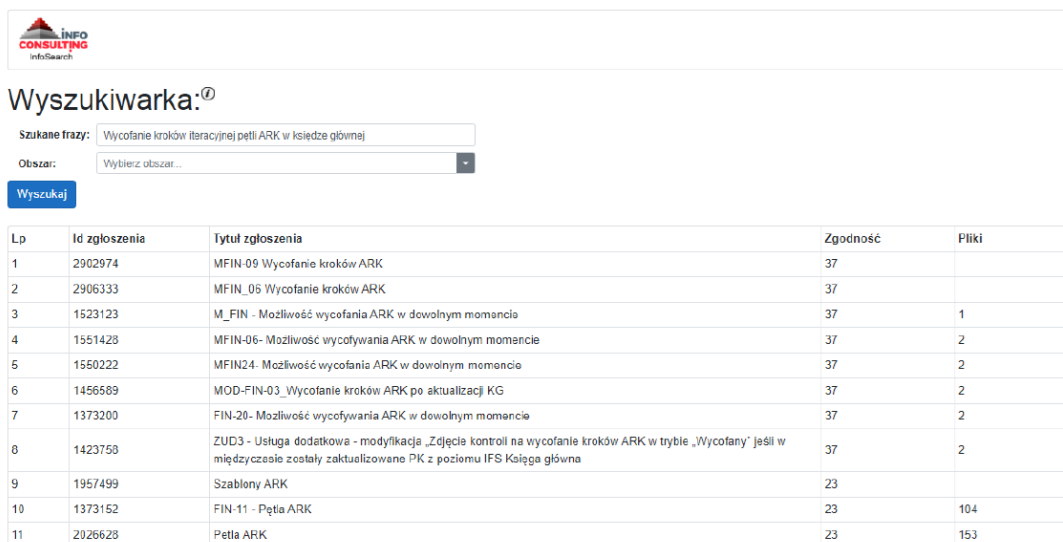
Obszar	Liczba klastrów (metoda Elbow)	Liczba klastrów (algorytm AP)	Maksymalny rozmiar klastra (metoda Elbow)	Maksymalny rozmiar klastra (algorytm AP)
Kokpity menedżerskie	2	2	2	2
CRM	2	3	8	6
Dystrybucja	100	399	100	49
Zarządzanie dokumentacją	5	4	7	5
Engineering i Zarządzanie produktem	14	19	74	74
Finanse	200	300	130	96
Warstwa administratora i aplikacji	25	39	150	100
Kadry	29	30	61	61
Utrzymanie ruchu	20	22	39	39
Produkcja	38	41	209	209
Place	10	15	67	54

Przedstawione badania pokazują, że istnieje potencjał w wykorzystaniu technik przetwarzania języka naturalnego (NLP) do porównywania dokumentacji specyfikacji modyfikacji w systemach ERP. Dzięki temu można znaleźć podobne modyfikacje, które zostały wcześniej wykonane dla innych klientów. W rezultacie możliwe jest automatyczne wsparcie w procesie produkcji modyfikacji poprzez pełne lub częściowe przeniesienie podobnych modyfikacji między środowiskami różnych klientów. To znacząco obniży koszty i czasochłonność realizowanych modyfikacji.

W związku z tym podjęto decyzję o konieczności unifikacji wybranych rozwiązań realizowanych przez zespół wdrożeniowy. W pierwszej kolejności uzyskane wnioski posłużyły do opracowania listy zagadnień, które powinny zostać przebudowane i zaprojektowane w IFS Cloud jako rozwiązania standardowe (VAPy). Analiza wykazała, że największa liczba modyfikacji, które można przenieść pomiędzy klientami występuje w module finansowym. W

drugiej kolejności opracowane wyniki posłużyły do przygotowania dokumentacji technicznej i biznesowej rozwiązań standardowych. Najbardziej zbieżne dokumenty specyfikacji zostały porównane, różnice scalone i opracowane jako jeden uniwersalny dokument opisujący dane zagadnienie biznesowe. Weryfikacja merytoryczna wyjaśniła, że wiele modyfikacji w tym module jest związanych z częstymi zmianami prawnymi w Polsce oraz kolejną adaptacją modyfikacji przenoszonej od innego klienta z drobnymi zmianami w kolejnym środowisku. Dlatego też, w pierwszym etapie prac podjęto się unifikacji specyfikacji rozwiązania oraz ujednolicenia kodu dla rozwiązań z obszaru finanse.

Doskonałym przykładem jest aktualny już VAP - „Pętla ARK”, do której tylko w zakresie adaptacji „Wycofania kroków iteracyjnej pętli ARK” było w badanej próbie co najmniej 8 u różnych klientów (Rys.6.9).



Lp	Id zgłoszenia	Tytuł zgłoszenia	Zgodność	Pliki
1	2902974	MFIN-09 Wycofanie kroków ARK	37	
2	2906333	MFIN_05 Wycofanie kroków ARK	37	
3	1523123	M_FIN - Możliwość wycofania ARK w dowolnym momencie	37	1
4	1551428	MFIN-06- Możliwość wycofywania ARK w dowolnym momencie	37	2
5	1550222	MFIN24- Możliwość wycofania ARK w dowolnym momencie	37	2
6	1456589	MOD-FIN-03_ Wycofanie kroków ARK po aktualizacji KG	37	2
7	1373200	FIN-20- Możliwość wycofywania ARK w dowolnym momencie	37	2
8	1423758	ZUD3 - Usługa dodatkowa - modyfikacja „Zdjęcie kontroli na wycofanie kroków ARK w trybie „Wycofany” jeśli w międzyczasie zostały zaktualizowane PK z poziomu IFS Księga główna	37	2
9	1957499	Szablony ARK	23	
10	1373152	FIN-11 - Pętla ARK	23	104
11	2026628	Pętla ARK	23	153

Rys. 6.9. Adaptacje zmian w kodzie - przykład pętli ARK

[88]

Po unifikacji dokumentacji i kodu źródłowego w ramach projektów wdrożeniowych IFS Cloud każdy klient otrzymuje informację biznesową dotyczącą standardowych rozwiązań oraz zasad wdrażania i przyszłego utrzymania dla tego typu rozwiązań. Zasady działania rozwiązań standardowych, które zostały wyspecyfikowane między innymi ze wsparcia opracowanej wyszukiwarki opisane zostały w podrozdziale 6.1.1.

Scenariusz testowy działania wyszukiwarki

Środowiskiem realizacji testów wydajnościowych jest obecne środowisko z dostępem do aplikacji i bazy danych. Do testów wykorzystano narzędzia:

- Baza danych Oracle (generator zapytań)
- Analiza czasów odpowiedzi

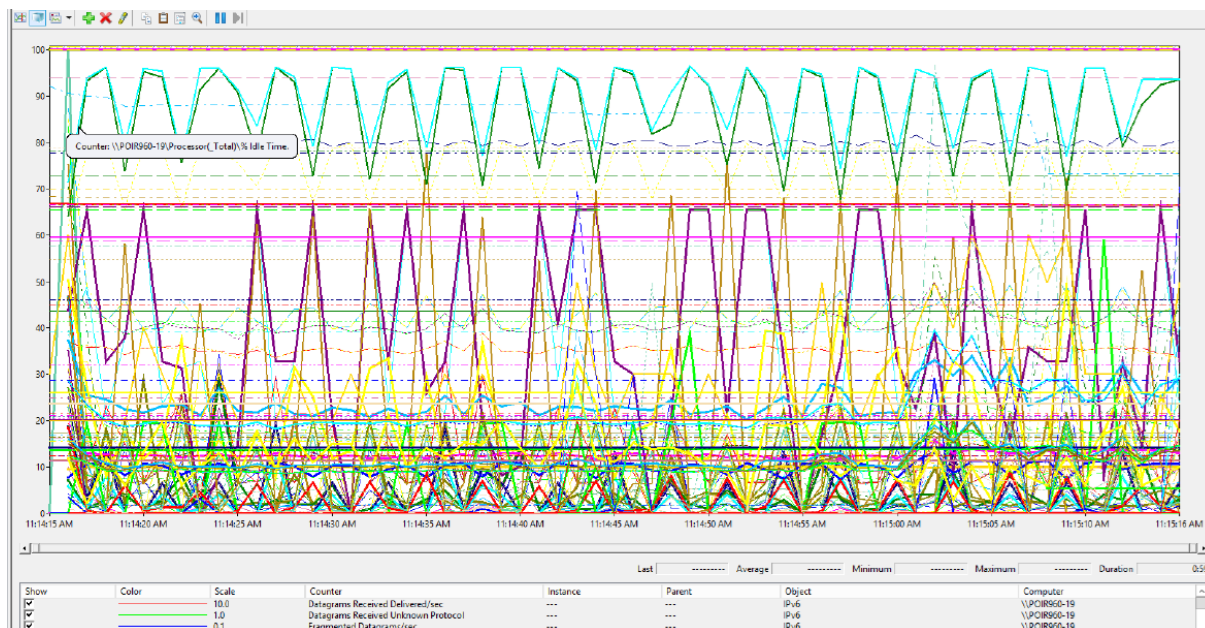
Zapytania realizowane są poprzez przygotowany dostęp za pomocą przeglądarki do aplikacji oraz dostęp do bazy danych poprzez Oracle SQL Developer, natomiast dostęp do serwera z aplikacją realizowany jest poprzez pulpit zdalny. Testy zostały zrealizowane z wykorzystaniem pakietu bazadanowego C_TEST_UTIL.API ze zbiorem funkcji i procedur. Scenariusz testowy zakłada realizację kroków:

1. logowanie do aplikacji ICSearchApp (Użytkownik domenowy)
2. logowanie do serwera aplikacji
3. uruchomienie monitora wydajności systemu Windows Serwer 2012R
4. LOAD TESTING - uruchomienie symulatora generowania 50 zapytań dla opcji Wyszukiwarka
5. uruchomienie i zapisanie pomiarów Monitora Wydajności po stronie Serwera Aplikacji SI
6. zebranie pomiarów z bazy danych i zapisanie do plików
7. wykonanie analizę zebranych pomiarów
8. zapisanie wyniku testu jako wnioski. W następnej kolejności kroki od 1 do 8 powtórzyć przy zwiększaniu zapytań do 2000 (stress testing).

Realizacja Testów

Celem testu jest symulacja 50 zapytań dla opcji Wyszukiwarka. Na środowisku DB pod adresem aplikacji: ICSearchApp(abs.local) jest wygenerowane 50 zapytań jednocześnie do wyszukiwania losowych fraz jak na rysunku Rys.6.8.

ID	LOG_TXT	PARAM_ID	TIMES
1	PARAM_ID[4013]. UPDATED IN +000000000 00:02:37.727000 sec	4013	157,73
2	PARAM_ID[4012]. UPDATED IN +000000000 00:02:34.558000 sec	4012	154,56
3	PARAM_ID[4011]. UPDATED IN +000000000 00:02:31.344000 sec	4011	151,34
4	PARAM_ID[4010]. UPDATED IN +000000000 00:02:28.161000 sec	4010	148,16
5	PARAM_ID[4009]. UPDATED IN +000000000 00:02:25.019000 sec	4009	145,02
6	PARAM_ID[4008]. UPDATED IN +000000000 00:02:21.874000 sec	4008	141,87
7	PARAM_ID[4007]. UPDATED IN +000000000 00:02:18.653000 sec	4007	138,65
8	PARAM_ID[4006]. UPDATED IN +000000000 00:02:15.424000 sec	4006	135,42
9	PARAM_ID[4005]. UPDATED IN +000000000 00:02:12.084000 sec	4005	132,08
10	PARAM_ID[4004]. UPDATED IN +000000000 00:02:08.833000 sec	4004	128,83
11	PARAM_ID[4003]. UPDATED IN +000000000 00:02:05.705000 sec	4003	125,71
12	PARAM_ID[4002]. UPDATED IN +000000000 00:02:02.596000 sec	4002	122,60
13	PARAM_ID[4001]. UPDATED IN +000000000 00:01:59.452000 sec	4001	119,45
14	PARAM_ID[4000]. UPDATED IN +000000000 00:01:56.303000 sec	4000	116,30
15	PARAM_ID[3999]. UPDATED IN +000000000 00:01:53.159000 sec	3999	113,16
16	PARAM_ID[3998]. UPDATED IN +000000000 00:01:50.024000 sec	3998	110,02
17	PARAM_ID[3997]. UPDATED IN +000000000 00:01:46.900000 sec	3997	106,90
18	PARAM_ID[3996]. UPDATED IN +000000000 00:01:43.782000 sec	3996	103,78
19	PARAM_ID[3995]. UPDATED IN +000000000 00:01:40.661000 sec	3995	100,66
20	PARAM_ID[3994]. UPDATED IN +000000000 00:01:37.539000 sec	3994	97,54
21	PARAM_ID[3993]. UPDATED IN +000000000 00:01:34.422000 sec	3993	94,42
22	PARAM_ID[3992]. UPDATED IN +000000000 00:01:31.309000 sec	3992	91,31
23	PARAM_ID[3991]. UPDATED IN +000000000 00:01:28.176000 sec	3991	88,18
24	PARAM_ID[3990]. UPDATED IN +000000000 00:01:25.046000 sec	3990	85,05
25	PARAM_ID[3989]. UPDATED IN +000000000 00:01:21.911000 sec	3989	81,91
26	PARAM_ID[3988]. UPDATED IN +000000000 00:01:18.811000 sec	3988	78,81
27	PARAM_ID[3987]. UPDATED IN +000000000 00:01:15.699000 sec	3987	75,70
28	PARAM_ID[3986]. UPDATED IN +000000000 00:01:12.601000 sec	3986	72,60
29	PARAM_ID[3985]. UPDATED IN +000000000 00:01:09.467000 sec	3985	69,47
30	PARAM_ID[3984]. UPDATED IN +000000000 00:01:06.351000 sec	3984	66,35
31	PARAM_ID[3983]. UPDATED IN +000000000 00:01:03.242000 sec	3983	63,24
32	PARAM_ID[3982]. UPDATED IN +000000000 00:01:00.099000 sec	3982	60,10
33	PARAM_ID[3981]. UPDATED IN +000000000 00:00:56.978000 sec	3981	56,98
34	PARAM_ID[3980]. UPDATED IN +000000000 00:00:53.837000 sec	3980	53,84
35	PARAM_ID[3979]. UPDATED IN +000000000 00:00:50.713000 sec	3979	50,71
36



Rys. 6.10. Obciążenie - testy wyszukiwarki

Maksymalny czas oczekiwania na odp.: 157,72 sek. Minimalny czas odp.: 3,83 sek. Średni czas oczekiwania na odp.: 80,49 sek. Średnia ilość wyników: 125 Równolegle był uruchomiony monitoring obciążenia serwera aplikacji SI, a jego wyniki prezentuje Rys.6.10

Z analizy obciążenia serwera stwierdzono iż maksymalny skok wykorzystania procesora wynosi około 25%, który pojawia się skokowo. Analizując czasy wykonywania zapytań do SI i porównując reakcję obciążenia procesora na serwerze aplikacji, jednoznacznie można stwierdzić że wszystkie zapytania do aplikacji są przetwarzane sekwencyjnie. Dany test identyfikuje problem projektowy danego rozwiązania. W takim trybie aplikacja wymaga natychmiastowej reorganizacji i przeprojektowania rozwiązania uwzględniając możliwość asynchronicznej obsługi zapytań aplikacji. Pozwoli to na szybsze odpowiedzi na konkretne zapytania, a co najważniejsze skróci czasy odpowiedzi. Dla potwierdzenia tej tezy wykonano kolejne analogiczne testy jednocześnie powiększając próbę. Jeśli teza się potwierdzi Load Testy zostaną zakończone.

6.4. Faza III

6.4.1. Reorganizacja procesu wytwarzania oprogramowania i narzędzia wspierające

W ramach realizacji Fazy III kluczowym wyzwaniem z perspektywy zarządczej jest właściwe skoordynowanie ogromnej liczby realizowanych równolegle prac. Dla tej fazy w

ramach aktualnie realizowanych projektów o średnim poziomie skali i złożoności (projekt o budżecie 700/1000 roboczodni) Kierownik Projektu nadzoruje średnio ponad 150 równoległych czynności. Zadania projektowe dotyczą wielu płaszczyzn, ale większość z nich objętych jest współpracą z zespołem technicznym.

- technologia i infrastruktura (od 20 do 200 zagadnień równoległych),
- dane - pozyskanie i przetwarzanie w zakresie migracji (ponad 100 zadań migracyjnych),
- aplikacja - konfiguracja i rozwiązania administratora (ponad 120 obszarów parametryzacyjnych wymagających kontroli),
- zagadnienia procesowe (w zależności od projektu: około 10 obszarów, dla każdego 10/20 procesów),
- zarządzanie projektem (zgodnie z metodyką wdrożeniową).

Ze względu na ogromny wpływ prac technicznych na finalne efekty projektu wdrożeniowego w trakcie prac nad Fazą III bardzo aktywnie włączany jest zespół programistów, którego pracownicy realizują zadania w ramach projektów wdrożeniowych, rozwojowych jak i serwisowych. Zadania realizowane przez zespół programistów przydzielane są do programowania na podstawie narzędzia IFS 10. Obecnie zespół wdrożeniowy realizuje średnio co najmniej 15 równoległych projektów wdrożeniowych. Każdy projekt charakteryzuje się innymi parametrami, które mogą wpływać na priorytet realizacji poszczególnych zadań w projekcie i pomiędzy nimi:

- wymagania formalne projektu - umowa
 - kary umowne
 - umowa serwisowa SLA (Service Level Agreement)
 - zakres biznesowy (MVP - minimum viable product, budżet)
- zakres modyfikacji
- rodzaj instalacji (chmura Azure, chmura sprzętowa, on-premise)
- wymagania w zakresie bezpieczeństwa
- etap projektu i wpływ innych projektów

W związku z tym, współpraca zespołu technicznego z zespołem wdrożeń uległa zmianie, a komunikacja została scentralizowana na poziomie portfela projektowego PMO (Biura Projektowego). W uproszczeniu dotychczasowy model przyjmowania zadań do

programowania polegał na tym, że każdy z programistów na bieżąco przeglądał kolejkę programistyczną i pobierał zadania do realizacji. Takie podejście miało swoje wady i zalety, ale największym wyzwaniem było:

- Dobranie zadania do kompetencji programisty,
- Nadzór nad liczbą zadań przejętych/otwartych przez programistę do realizacji,
- Oszacowanie czasu wymaganego do zakończenia prac nad modyfikacją,
- Oszacowanie skali produkcji w toku oraz oszacowanie racjonalnej daty zakończenia prac, programistycznych w oparciu o zarobienie dostępnych zasobów,
- Zarządzanie priorytetami programistycznymi w organizacji:
 - perspektywa portfela projektów wdrożeniowych i ich hierarchii,
 - zadania rozwojowe – usługi dodatkowe
 - zadania serwisowe – w tym zadania podlegające SLA
- Nadzór nad zespołem w zakresie ewentualnych nieplanowanych wyłączeń i ich wpływu na planowany czas dostawy modyfikacji „lead time” – absencje na żądanie, choroby, inne pilne a nieplanowane z wyprzedzeniem zadania.

Planowanie operacyjne zadań wymagających wsparcia programistycznego - perspektywa organizacyjna

Proces zmian w zakresie planowania prac programistycznych został przeprowadzony zarówno na płaszczyźnie technicznej jak i organizacyjnej. Wykonanie zmian w systemie byłoby zupełnie bezużyteczne gdyby nie zmiany organizacyjne i definicja nowych struktur w zakresie:

- Powołania kilkusobowych zespołów programistycznych koordynowanych przez Lidera odpowiedzialnego za odpowiednią alokację zadań w ramach przydzielonego mu zespołu i zadań,
- Powołania grupy mającej za zadanie koordynowanie i bieżący nadzór nad priorytetyzacją prac programistycznych dla całej organizacji. W skład zespołu zgodnie z minimalnymi wymaganiami organizacji weszli:
 - Koordynator zadań programistycznych dla projektów wdrożeniowych – przedstawiciel PMO (Biura projektowego),
 - Koordynator zadań serwisowych,

- Koordynator zadań rozwojowych – usług dodatkowych,
- Liderzy zespołów programistycznych.

Zgodnie z planem prace realizowane w ramach nowych struktur są uzupełnieniem bieżącej aktywności i nie powinny zajmować więcej niż 1 godzinę tygodniowo. W ramach struktur zespołu deweloperskiego w pierwszym etapie rozwoju planowania operacyjnego zostały zdefiniowane 3 nowe zespoły. Zespoły zostały tak dobrane, aby poziom kompetencyjny i zdolności programistyczne były na bardzo podobnym poziomie. Za planowanie, wsparcie i rozliczanie prac każdego zespołu odpowiedzialny jest Lider zespołu. Bez zgody władz organizacji nie byłoby możliwe utworzenie nowych struktur koordynujących prace programistyczne, a tym samym zmiana modelu planistycznego.

Planowanie operacyjne zadań wymagających wsparcia deweloperskiego - perspektywa systemowa

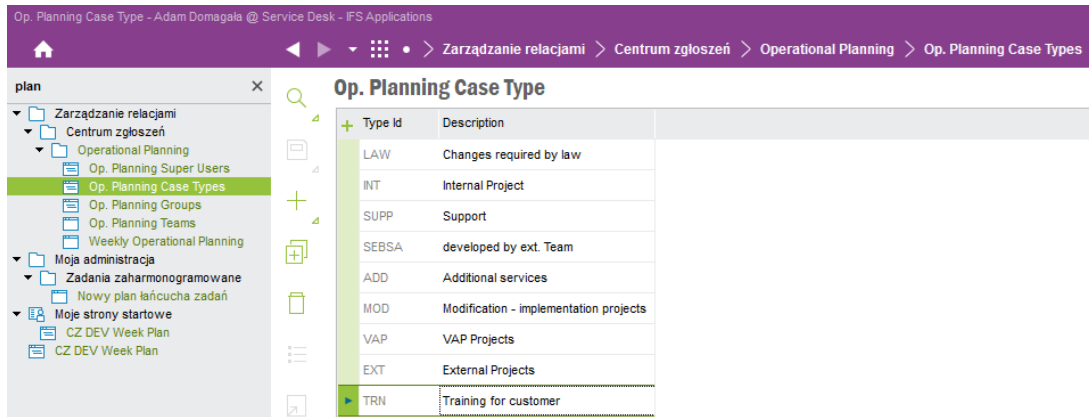
Projekt rozwiązania dotyczący planowania operacyjnego został zbudowany w taki sposób, aby systemowo możliwe było obsłużenie różnego typu zadań realizowanych w ramach działalności organizacji. Projekt zakładał, że planowanie operacyjne po pilotażowym uruchomieniu w zakresie prac programistycznych będzie mogło zostać rozpropagowane w pozostałych jednostkach organizacyjnych firmy. W związku z tym słownik typów zadań został zbudowany w formie listy otwartej z możliwością ciągłej rozbudowy.

Rysunek Rys.6.11 przedstawia listę typów zadań, które według planu mogą być wykorzystywane do planowania prac wdrożeniowych, serwisowych, usług dodatkowych czy też R&D. W kontekście planowania prac programistycznych w pierwszej kolejności zdecydowano o podzieleniu prac programistycznych według źródeł ich pochodzenia:

- Działania związane z serwisem po wdrożeniowym,
- Usługi dodatkowe realizowane przez dział serwisu,
- Działania wdrożeniowe,
- Działania R&D – nowe produkty.

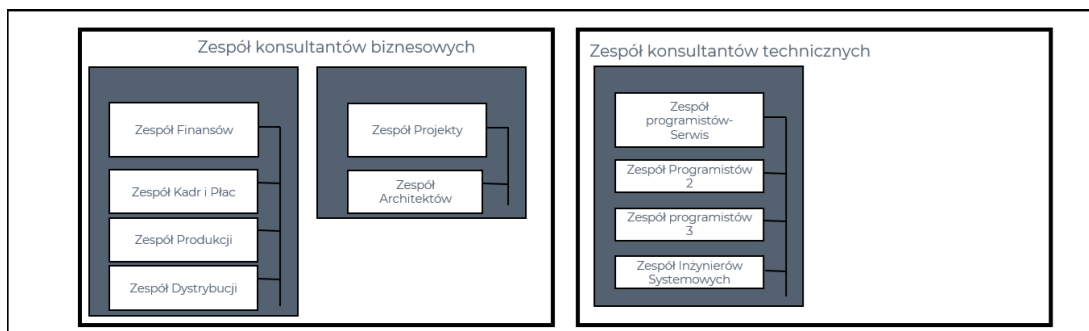
Według założeń podział ten został wykonany wyłącznie na potrzeby grupowania źródła pochodzenia pożądanej zmiany i nie powinien wpływać na proces kolejowania czy priorytetyzację zadań. Jednak w trakcie wdrożenia rozwiązania i zmian struktury zespołu deweloperów zdecydowano o zmianie podejścia w odniesieniu do programistów realizujących prace dla klientów serwisowych. Odstępstwo było spowodowane:

- Złożonymi wymaganiami klientów serwisowych w zakresie SLA, co wpływało na brak możliwości planowania tych prac w trybie tygodniowym, a w wielu przypadkach nawet dziennym,



Type Id	Description
LAW	Changes required by law
INT	Internal Project
SUPP	Support
SEBSA	developed by ext. Team
ADD	Additional services
MOD	Modification - implementation projects
VAP	VAP Projects
EXT	External Projects
TRN	Training for customer

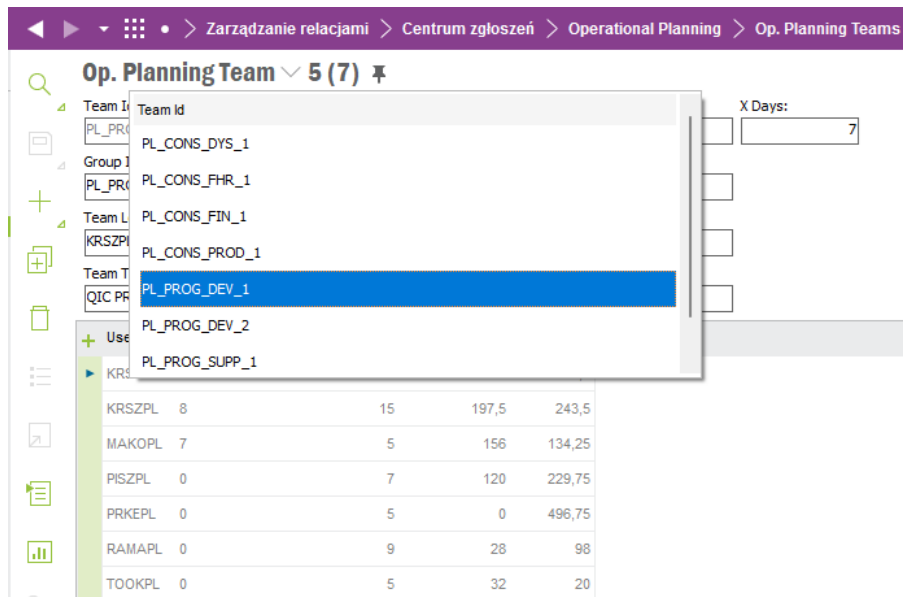
Rys. 6.11. Planowanie operacyjne według typu pracy



Rys. 6.12. Podział zespołów deweloperskich i konsultingowych

- Innym charakterem zadań serwisowych, które w wielu przypadkach mają znamiona pracy ciągłej – nieskalowalnej w zakresie czasu realizacji czy terminu wykonania.

W związku z tym zespół programistów został podzielony na grupy operacyjne o zbieżnych kompetencjach, w skład których wchodził lider oraz do 7-8 programistów. Jedna z tych grup została dedykowana do realizacji prac serwisowych, a pozostałe miały za zadanie realizować wszystkie kolejowane zadania. Zespół dedykowany do realizacji prac serwisowych powinien również podlegać cotygodniowej kontroli w zakresie potencjalnej dostępności czasu do realizacji innych prac niż serwisowe. W ramach wolnych zasobów powinien być również wykorzystywany do działań w pozostałych obszarach programistycznych, ale tylko i wyłącznie wtedy, kiedy nie realizuje innych bieżących aktywności serwisowych. Rysunek Rys.6.12 przedstawia podział zespołów deweloperskich i konsultingowych na moment uruchomienia zaprojektowanego rozwiązania. Zlokalizowane w Polsce zespoły programistów zostały utworzone w taki sposób, aby mogły realizować zadania o każdym poziomie złożoności, a Lider zespołu był jednocześnie mentorem dla mniej doświadczonych osób. W ten sposób planowaliśmy osiągnąć optymalne wyniki w zakresie terminowości i doboru zadań. Z perspektywy technicznej Lider zespołu odpowiada za cotygodniowy nadzór nad terminowością zadań i optymalnym obciążeniem swojego zespołu. Opracowany w systemie IFS



Rys. 6.13. Planowanie operacyjne - grupy zadaniowe

mechanizm umożliwi bieżącą administrację zespołami w zakresie przedstawionym na rysunku Rys.6.13 i dotyczy m. in:

- dodawania/usuwania członków zespołu w tym zmianę Lidera zespołu
- nadzoru nad rodzajem zadań kierowanych do zespołu (kolejki zadań)
- długość okresu planistycznego (domyślnie w organizacji, ustawiona na cykl 7-dniowy)
- bieżący podgląd w zakresie ilości przypisanych zadań oraz ich kalkulacji kosztowej

Bieżące planowanie prac programistycznych bazuje na nowym opracowanym dedykowanym ekranie, który zbiera informacje z poszczególnych zespołów należących do grupy programistów. Ekran przedstawia dostępność zespołów w zakresie czasu pracy z perspektywy konkretnego tygodnia. Narzędzie daje możliwość podglądu dostępności zespołów programistycznych w dłuższym horyzoncie czasowym poprzez wybór kolejnych okresów planistycznych (tygodnie). Daje to możliwość rozplanowania wszystkich prac programistycznych dostępnych na moment realizacji planu i określenie w oparciu o planowany budżet terminu realizacji poszczególnych zadań.

Rysunek Rys.6.15 przedstawia narzędzie, które jest podstawą spotkania zespołu planistycznego. Na podstawie prezentowanych informacji, osoby odpowiedzialne za poszczególne kanały biznesowe (wdrożenia, serwis, R&D czy usługi dodatkowe) mają wpływ na ewentualną zmianę priorytetów i terminy przypisania ich zadań do realizacji. Liderzy programistów przed spotkaniem weryfikują stan bieżących prac programistycznych w swoich zespołach i aktualizują dostępność z perspektywy najbliższych 3-4 tygodni planistycznych.

The screenshot shows the 'Op. Planning Teams' interface. The left sidebar contains a navigation tree with 'Op. Planning Teams' selected. The main area displays details for 'Op. Planning Team 5 (7)'. Below the details is a table of assigned tasks.

User Id	Assigned Cases	Assigned Tasks	Assigned Hours	Used Hours
KRSOPL	0	10	80	124,5
KRSZPL	8	15	197,5	243,5
MAKOPL	7	5	156	134,25
PISZPL	0	7	120	229,75
PRKEPL	0	5	0	496,75
RAMAPL	0	9	28	98
TOOKPL	0	5	32	20

Rys. 6.14. Planowanie operacyjne - grupa zadaniowa

Warto zwrócić uwagę, że w przypadku naszej organizacji dostępność pracownika w zespole programistycznym liczona jest jako 32 godzinny wymiar dla normalnego tygodnia roboczego. Przyjęty, zaniżony względem formalnego 40 godzinnego tygodnia pracy wymiar jest wynikiem korelacji naszych doświadczeń zderzonych z formalnymi normami czasu pracy. Na podstawie doświadczenia z poprzednich projektów zdecydowano, że każdy z programistów ma dostępne około 2 godziny dziennie (do 8 godzin tygodniowo) na prace związane z ewentualnymi nieplanowanymi poprawkami, konsultacjami czy wsparciem innych członków zespołu, a co nie jest związane z zadaniami w jego kolejce programistycznej. Ekran planistyczny pozwala na ewentualną zmianę planów lub przydzielenie prac w zakresie dostępnego do rozplanowania czasu pracy „szacowana dostępność” dla danego tygodnia. „Szacowana dostępność” całej grupy programistycznej jest uzależniona od:

- Niedostępności pracowników – urlopy, chorobowe, szkolenia, itp.
- Poziomu realizacji prac zaplanowanych na poprzedni okres planistyczny.

Podczas sesji planistycznej podejmowane są decyzje w zakresie:

- Ewentualnej zmiany zakolejkowanych już do realizacji zadań – np. na kolejne okresy planistyczne, a wymagające przyspieszenia w realizacji.
- Przydzielenia zadania do konkretnego programisty – ponieważ robił już podobną modyfikację, lub z innej przyczyny, która może wpłynąć na poprawienie efektywności w zakresie realizacji i odbioru.

Weekly Operational Planning 44 (82) ✕

Group Id: Group Description: Year: Week No:

Estimated Capacity: Assigned Capacity to Employees: Assigned Capacity to Queues:

Team Description	Team Id	Team Leader	Estimated Capacity	Assigned Capacity To Employees	Team Queues	Capacity Assigned To Queues
External team from Czech and Slovakia	CZ_EXT_TEAM	SLKUSK	0	0		0
Poland Programming Team 1	PL_PROG_DEV_1	KRSZPL	92	0	QIC PROG;QIC PROG ADV;QIC PROG LEAD;QL...	0
Poland Programming Team 2	PL_PROG_DEV_2	RAWLPL	24	0	QIC PROG;QIC PROG ADV;QIC PROG LEAD;QL...	0
Poland Programming Support Team 1	PL_PROG_SUPP_1	PISKPL	0	0	QIC PROG;QIC PROG ADV;QIC PROG LEAD	0

User Id	Description	Estimated Capacity	Assigned Capacity	Modified By	Modified Date
KRSOPL	torpot, górstal	16	0	KRSZPL	17.11.2023 12:25:24
KRSZPL	Zadządanie + zaległe zadania	8			
MAKOPL	zadania zaległe: certyfikacja; ksef dla "deski"	12			
PISZPL	dostępne wolne moce na tydzień 47 + Urlop	24			
PRKEPL	zadania zaległe: projekt "Deska" - brak wolnych (tydzień 47)	0			
RAMAPL	zadania zaległe: Wik (PKP Cargo); IFS CLOUD BER... - brak wolnych (tydzień 47)	0			
TOOKPL	dokończyć Alu; + KSEF; + wolne moce na tydzień 47	32			

Rys. 6.15. Planowanie operacyjne w podziale na tygodnie

Podczas spotkania omawiane są modyfikacje, których termin wykonania przypadał na bieżący tydzień, a nie zostały zrealizowane w terminie. W takich przypadkach omawiany jest plan naprawczy, aktualizowana jest data wykonania i jeśli jest to konieczne, identyfikowana jest przyczyna opóźnień i podejmowane są działania mające na celu ich minimalizację. Bezpośrednio z ekranu planistycznego, każdy członek zespołu decyzyjnego ma możliwość analizy w czasie rzeczywistym status przydzielonych do konkretnego programisty prac w kontekście:

- zadań przypisanych do programisty w ramach bieżącego okresu planistycznego,
- wszystkich zadań przydzielonych do programistów,
- wszystkich zadań przydzielonych do wybranego programisty,
- zadań przydzielonych w innych okresach planistycznych.

Zespół planistyczny ma również możliwość weryfikacji wykorzystania budżetu, czyli porównania czasu planowanego do czasu już poświęconego na wykonanie oraz dokonać analizy w zakresie ilości nawrotek na każdym etapie wytwarzania zmiany. Większość z powyższych operacji możliwe jest do wykonania z poziomu dedykowanych raportów ekranowych, jak jest to przedstawione na rysunku Rys.6.16.

ID zgl.	Tytuł	ID zadania	Działanie	Activity Desc	Typ zad.	Utworzone	Czas u.	Godz. pozostałe	Budget	Najpóźniejsza data rozpoczęcia pr...	Termin real.	Status zad.
IC3051769	Problem z białą listą na kliencie	500	37	PL PROGRAMMING	RnD Department	09.09.2024 09:23:47	0	0,0	0	11.09.2024	13.09.2024 00:00:00	Wycena
IC3051125	SCM (errors)	504	37	PL PROGRAMMING	RnD Department	26.08.2024 12:39:29	1	0,0	0	27.08.2024	29.08.2024 00:00:00	Wycena
IC3038951	SPR.003 Mechanizmy kontroli w obarzarze SP...	521	37	PL PROGRAMMING	RnD Department	27.08.2024 15:02:43	19	0,0	0	04.09.2024	06.09.2024 00:00:00	Wycena
IC3036585	PRD.001 Integracja z SAPIAPO	600	37	PL PROGRAMMING	RnD Department	27.09.2023 12:34:50	235,75	0,0	0	19.06.2024	21.06.2024 00:00:00	Wycena
IC3040522	Integracja z COGNOS	100	37	PL PROGRAMMING	RnD Department	31.07.2023 08:57:55	241	0,0	0			Wycena

Rys. 6.16. Planowanie operacyjne przypisanych zadań

Bardzo ważnym zadaniem jest zarządzanie zadaniami przekazanymi do kolejki. Z perspektywy planistycznej wszystkie zadania trafiają do jednej grupy zadań, ale w ramach działania dotyczącego programowania lub wyceny prac. Niezależnie czy aktywności dotyczą programowania nowej modyfikacji, realizacji poprawek, zmian czy rozbudowy każda następną iteracja wraca z kolejną wartością sekwencji, ale w ramach zadania na programowanie. Analogicznie sytuacja wygląda w przypadku działań dotyczących wyceny/szacowania kosztów prac, niezależnie czy kolejne prace dotyczą poprawek, czy zmian krok powinien wystąpić z określeniem wartości czasu koniecznego na wykonanie pracy przez programistę.

Podczas projektowania i realizacji rozwiązania dotyczącego planowania prac programistycznych wykonane zostały dedykowane ekrany do analizy kolejki programistycznej. Na bazie zapytań SQL oraz konfiguracji systemowych funkcji warunkowych zespół planistycznych ma możliwość m.in.:

- identyfikacji zadań, które są w kolejce do wyceny,
- identyfikacji zadań, które są w kolejce do programowania,
- bieżącej analizy „lead time” – najpóźniejszej daty rozpoczęcia programowania w momencie zakończenia prac związanych z wyceną zadania,
- bieżącej analizy „lead time” – dla wszystkich zadań przekazanych zarówno do wyceny jak i programowania,
- planowania testów i prac odbiorowych modyfikacji w związku z oszacowanym terminem realizacji prac programistycznych.

Głównym celem zmian w procesie planistycznym naszej organizacji było określenie na podstawie systemowego stanu wiedzy jak największej ilości szczegółów związanych z realizowanymi zadaniami programistycznymi, co przedstawia przykładowe poniższe zestawienie (Rys.6.17). Z dotychczasowych doświadczeń projektowych wynikało, że najpoważniejszym ryzykiem projektowym były potencjalne opóźnienia związane z przygotowaniem rozwiązania jako finalnego produktu Fazy III. Główną przyczyną opóźnień w zakresie Fazy III były najczęściej nieterminowo przygotowane i zaakceptowane prace związane ze zmianami do standardu systemu (CRIM). Mając do dyspozycji dodatkowe

ID zad.	Tytuł	ID zadania	Działanie	Activity Desc	Typ zad.	Utworzone	Czas użyty	Godz. pozostałe	Budget	Najpóźniejsza data rozpoczęcia pr...	Termin real.	Status zad.
IC3040336	LOG-09 Generowanie zapotrzebowania na u...	500	37	PL PROGRAMMING	RnD Department	28.08.2024 09:58:19	0	216,0	64	10.10.2024	25.10.2024 00:00:00	Przebiega w planie
IC3051067	Integracja APS – Przetwarzania otrzymanych ...	500	37	PL PROGRAMMING	RnD Department	03.09.2024 19:57:18	0	0,0	40	04.09.2024	13.09.2024 00:00:00	Przebiega w planie
IC3050362	HRM- nowa opcja zniszczenia narzędzi	500	37	PL PROGRAMMING	RnD Department	22.08.2024 12:07:41	0	0,0	24	30.08.2024	05.09.2024 00:00:00	Przebiega w planie
IC3044257	Instalacja rozwiązania KSEF	100	37	PL PROGRAMMING	RnD Department	15.01.2024 12:13:48	0	0,0	24			Przebiega w planie
IC3045787	Integracja farmaprom - rozszerzenie	300	32	PL WORK ESTMA.	RnD Department	24.06.2024 14:44:36	0	0,0	3	20.05.2024	30.08.2024 00:00:00	Przebiega w planie
IC3051708	Wydruk faktury podreczej-brak na wydruku f...	200	3	APPLY LOCAL FX	Support	13.09.2024 09:01:22	0	0,0	3	28.10.2024	16.09.2024 00:00:00	Przebiega w planie
IC3051313	CR - New ENG Search Fields	30	32	PL WORK ESTMA.	RnD Department	23.08.2024 14:00:11	0	0,0	2,5	30.08.2024	28.08.2024 00:00:00	Przebiega w planie
IC3051303	CR-New ENG BOM Columns	30	32	PL WORK ESTMA.	RnD Department	23.08.2024 13:50:46	0	0,0	2,5	30.08.2024	30.08.2024 00:00:00	Przebiega w planie
IC3036038	FN-M-011-Uzupelnienie typu dostawy i typu P...	300	32	PL WORK ESTMA.	RnD Department	27.08.2024 10:24:02	0	0,0	2,5	02.09.2024	03.09.2024 00:00:00	Przebiega w planie
IC3036738	FN-I-012- STOCK	300	32	PL WORK ESTMA.	RnD Department	27.08.2024 10:46:54	0	0,0	2,5	02.09.2024	03.09.2024 00:00:00	Przebiega w planie
IC3037106	LOG-034 INTRASTAT - integracja z systemem ...	300	32	PL WORK ESTMA.	RnD Department	06.09.2024 09:07:57	0	0,0	2,5	05.09.2024	06.09.2024 00:00:00	Przebiega w planie
IC3036798	FN-M-031 Pyta ARK	300	32	PL WORK ESTMA.	RnD Department	29.08.2024 08:25:37	0	0,0	2,5	05.09.2024	06.09.2024 00:00:00	Przebiega w planie
IC3051532	WMS-06.2 Kwarantanna produkcyjna rozsz...	30	32	PL WORK ESTMA.	RnD Department	02.09.2024 16:21:46	0	0,0	2,5	06.09.2024	09.09.2024 00:00:00	Przebiega w planie
IC3051557	FN-M-004 Dostosowanie formatów plików ba...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 14:54:24	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051554	FN-M-002 Data kursu waluty na podstawie da...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 14:56:09	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051912	GL107 Notifications on Critical and Moderate N...	300	32	PL WORK ESTMA.	RnD Department	12.09.2024 22:37:51	0	8,0	2,5	18.09.2024	19.09.2024 09:00:00	Przebiega w planie
IC3051564	FN-C-004 Checkboxy w transakcjach podatko...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 12:21:57	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051561	FN-C-001 Wyciągnięcie pól: Konto bankowe, ...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 17:47:29	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051555	FN-M-003 Wydruk kompensaty dostawcy ora...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 15:04:56	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051562	FN-C-002 Wyszukiwanie po numerze NIP	300	32	PL WORK ESTMA.	RnD Department	05.09.2024 12:20:11	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051565	FN-R-001 Raport porównujący rejestr VAT UE...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 17:50:29	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051558	FN-M-005 Kompensaty zaliczek na oknie rejes...	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 17:49:01	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie
IC3051560	FN-I-001 Wysyłka VAT-UE	30	32	PL WORK ESTMA.	RnD Department	05.09.2024 17:45:56	0	0,0	2,5	11.09.2024	12.09.2024 00:00:00	Przebiega w planie

Rys. 6.17. Zestawienie zadań do przypisania

narzędzie planistyczne oraz związane z nim dodatkowe raporty zarówno Kierownik Projektu jak i Lider Zespołu Programistycznego mogą w szybki sposób otrzymać informacje, kiedy musi nastąpić rozpoczęcie prac programistycznych, aby osiągnąć planowany termin realizacji zadania. Współpraca Zespołu Planistycznego, Biura Projektowego i Zespołu Programistów jest szczególnie ważna, w okresie, kiedy liczba zadań śpiętrza się a termin najpóźniejszego startu oddala się względem bieżącej daty. W tej sytuacji bardzo ważne jest właściwe zarządzanie zadaniami, które są na ścieżce krytycznej każdego z projektów wdrożeniowych.

6.4.2. Cykl Deminga - programowanie CRIM w ramach portfela projektowego

Proces obsługi zmian w naszej organizacji funkcjonuje na podstawie procedury zatwierdzonej na poziomie Prezesa Zarządu. Autoryzacja dokumentu na najwyższym szczeblu organizacji obrazuje, jak ważnym elementem procesu wdrożeniowego są zagadnienia związane z produkcją modyfikacji. Powodem tego stanu rzeczy są:

- wspólne zasoby programistyczne między działem serwisu i działem wdrożeń
- wysokie koszty wytwarzania i potrzeba dokładnego śledzenia kosztu/czasu
- wpływ na powodzenie projektu
 - rozszerzanie zakresu rozwiązania (pływający zakres)
 - jakość dostarczonych produktów
 - terminowość realizowanych zadań

Doskonałą okazją do aktualizacji procedury wytwarzania zmian w naszej organizacji okazał się wdrożony i produkcyjnie działający nowy model planistyczny prac programistycznych. Głównym celem jaki został postawiony wobec nowej odsłony procedury wytwarzania modyfikacji, była koncentracja na następujących elementach:

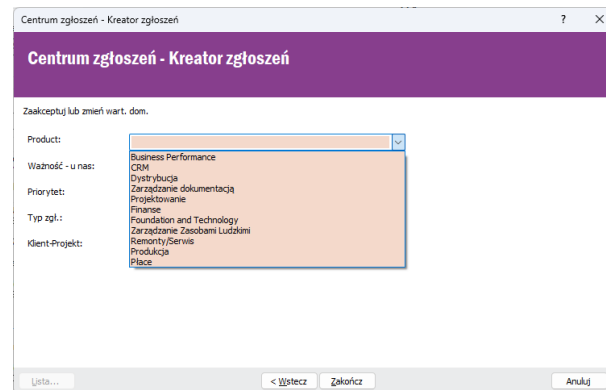
- nadzór nad efektywnością i budżetem zadań w ramach obsługi działań programistycznych,
- usprawnienie procesu związanego z zarządzaniem jakością,
- usprawnienie procesu – komunikacja, monitoring terminów realizacji i ewentualnych zaległości.

Podczas prac nad rozprawą i w ramach realizacji bieżących projektów wdrożeniowych wprowadzono kilka testowych zmian systemowych i procesowych w ramach obsługi zgłoszeń. Wprowadzony został m. in.:

- systemowy obowiązek rejestracji zgłoszenia już na etapie HLS (High Level Specification),
- obowiązek nagrywania w formie video z wyniku z testów modyfikacji,
- systemowa wycena prac konsultanta i programisty,
- audyt i wprowadzenie zespołu serwisowego do procesu wytwarzania modyfikacji na etapie wdrożenia.

Część zmian okazała się trudna w realizacji i w perspektywie czasu naturalnie wygasła, czego przykładem było przykładowo nagrywanie wyników z testów. Przechowywanie danych powiązanych ze specyfikacjami było pracochłonne, kosztowne, a finalnie bardzo sporadycznie wykorzystywane. Inne procesy jak rejestr wycen konsultantów, czy udział serwisu w procesie wytwarzania zmian jest optymalizowany w dalszym ciągu. Docelowo procedura została uzbrojona w narzędzia mające na celu automatyzację komunikacji i monitoringu prac do wykonania w procesie. Pierwszym ważnym zagadnieniem, które wpływa na poprawność komunikacji jest właściwe przypisanie zmiany do produktu jak na rysunku Rys.6.18. Lista modułów dla zgłaszanej modyfikacji jest bazą i jedną z podstawowych informacji, która jest przekazywana do aplikacji "InfoSearch", której cel został opisany w niniejszym rozdziale.

Proces obsługi zgłoszeń rozpoczyna się na etapie standardowego kreatora „Centrum Zgłoszeń” systemu IFS. Potwierdzeniem prawidłowej rejestracji zgłoszenia jest, tak jak to przedstawione na rysunku Rys.6.19, wygenerowane ID zgłoszenia wraz z listą możliwych do wykonania czynności. Jediną zmianą jaka została wprowadzona do procesu na tym etapie, była rezygnacja z automatycznego generowania przez system wszystkich zadań możliwych do



Rys. 6.18. Przypisanie zmiany do produktu

realizacji w ramach zgłoszenia. Dotychczasowo użytkownik po wprowadzeniu zgłoszenia mógł zobaczyć wszystkie możliwe do realizacji kroki w procesie. Powodowało to bardzo duże zamieszanie szczególnie w przypadku mniej doświadczonych użytkowników, którzy nie byli w stanie prawidłowo określić etapu, na którym jest dane działanie lub kto opiekuje się bieżącym zadaniem w ramach zgłoszenia. Dlatego też, zdecydowano o odejściu od generowania pełnego szablonu zadań i zobowiązano każdego kolejnego uczestnika do świadomego założenia zadania dla kolejnej osoby/użytkownika w procesie. Ewentualne systemowe zatory w procesie zostały wyeliminowane poprzez opracowane zdarzenie systemowe, które na bieżąco śledzi czas przetrzymywania danego zadania na każdym z kroków i powoduje generację komunikatów email do osoby odpowiedzialnej oraz jej przełożonego w przypadku przekroczeń czasu bezczynności o zadany okres. Korelacja zmian w zakresie procesu obsługi zgłoszeń i planowania prac programistycznych wymagało budowy nowych ekranów, które umożliwiły efektywniejszą pracę w zakresie zgłoszeń. Kluczowe zmiany dotyczyły:

- ekranu umożliwiającego obsługę budżetów zadań,
- przeglądu zgłoszeń,
- automatyzację numeracji zadań powiązaną z rodzajem działania,
- ograniczeniem możliwości wyboru przez użytkownika działania niezgodnego z procesem,
- dodanie nowych pól wspierających i zdarzeń systemowych umożliwiających proces monitoringu zadań,
- dodatkowe opcje dostępne pod prawym przyciskiem myszy dla użytkownika, na przykładzie rysunku Rys.6.20.

Rys. 6.19. Kreator zgłoszenia w centrum zgłoszeń

Rys. 6.20. Dodatkowe opcje w ekranie obsługi zgłoszenia

Proces obsługi zgłoszenia, którego celem jest wykonanie zmiany do standardowego rozwiązania lub wsparcia programistycznego, zostało obudowane szczegółowym scenariuszem, który aktualny przebieg przedstawiono w tabeli 6.2:

Tabela 6.2. Kroki w procesie wytwarzania zmian programistycznych projektu ERP

ID działania	Nazwa działania	Opis	Czas realizacji
1XX	Specyfikacja wysokiego poziomu (HLS)	Opracowanie dokumentacji specyfikacji wysokiego poziomu modyfikacji zgodnie z szablonem HLS.	Ustalony z PM, realizowane w fazie II
2XX	Audyt	Przekazanie dokumentacji HLS do audytu przez Architekta Projektu.	7 dni
3XX	Audyt i wycena programowania (HLS)	Po audycie dokumentacji HLS, zadanie kierowane na wycenę programistyczną przez Architektów Technicznych.	7 dni
1XX	Specyfikacja funkcjonalna modyfikacji	Opracowanie szczegółowej specyfikacji modyfikacji (SM) zgodnie z wymaganiami formalnymi i uzgodnieniami z klientem.	Potwierdzony z PM, czas uzupełniony w "Termin realizacji"
25X	Audyt specyfikacji modyfikacji	Audyt specyfikacji funkcjonalnej modyfikacji pod kątem spójności i poprawności wyceny prac.	7 dni
3XX	Audyt i wycena programistyczna	Wycena programistyczna z uwzględnieniem szacowanego czasu i daty realizacji.	7 dni
4XX	Zgoda na programowanie	Zatwierdzenie dokumentacji przez Kierownika Projektu (PM) oraz klienta.	7 dni
5XX i 5X1	Programowanie	Zadanie kierowane na programowanie po weryfikacji budżetu i akceptacji klienta.	Zgodnie z budżetem, poprawki w 3 dni
55X	Audyt kodu	Przegląd kodu pod kątem zgodności z wewnętrznymi standardami jakości.	7 dni
6XX i 6X1	Testy modyfikacji	Weryfikacja działania modyfikacji przez konsultanta, ewentualne zgłoszenie poprawek.	7 dni
7XX	Instalacja modyfikacji	Instalacja modyfikacji w środowisku klienta.	Do 7 dni
8XX	Odbiór modyfikacji	Odbiór modyfikacji przez klienta po przedstawieniu mu wdrożonej zmiany.	14 dni

Journal	Description	Tasks	Business Objects	Case Overview	Contacts	Support Info	Customer Info	SLA	Budget	Customization Project	Terms of Service (Read Only)					
Task Id	Task Owner	Task Activity	Total Bu.	Sales Estimation	Authorized By	Authorization Date	Revoked By	Revoked Date	Copied To Case Budget	Estimation R...	Budget Cons	Budget Dev	Used Budget...	Budget PM	Budget Other	Total Used B...
310	RAWLPL	PL WORK ESTIMATION	26,5	0						1	0	26,5	0,5	0	0	0,5
500	JAPAPL	PL PROGRAMMING	24	0							0	24	25	0	0	25
501	JAPAPL	PL PROGRAMMING	16	0							0	16	16	0	0	16
502	JAPAPL	PL PROGRAMMING	6	0							0	6	7	0	0	7
700	MASPL	PL INSTALLATION	1	0							0	1	1	0	0	1
701	MASPL	PL INSTALLATION	1	0							0	1	0	0	0	0
702	TOGEPL	PL INSTALLATION	1	0							0	1	0	0	0	0
550	RAWLPL	PL CODE REVIEW	0	0							0	0	0,5	0	0	0,5

Budget Type	Description	Budget	Notes	Changed By	Changed Date	Sales Estimation
ADM	Administration	0,5	TOGEPL	14.06.2024	10:50:16	
CODEREV	Code Review	0,5	TOGEPL	14.06.2024	10:50:16	
DEL	Delivery to customer	1	TOGEPL	14.06.2024	10:50:16	
EST	Work Estimation	0,5	TOGEPL	14.06.2024	10:50:16	
PRG	Programming	24	RAWLPL	17.06.2024	08:10:54	

Rys. 6.21. Wycena związana z obsługą modyfikacji

1. Specyfikacja wysokiego poziomu (HLS) - PL PRE-SPECIFICATION (ID zadania: 1XX)

Czas realizacji: Ustalony z Kierownikiem Projektu (PM) - działanie najczęściej realizowane jest w ramach FAZY II podczas opracowania Koncepcji Biznesowej.

Proces rozpoczyna się od założenia zgłoszenia przez konsultanta oraz przejęcia zgłoszenia pod opiekę. Oznacza to rozpoczęcie pracy nad dokumentacją specyfikacji wysokiego poziomu dla potencjalnej modyfikacji zgodnie z uzgodnionym przez zespół wdrożeniowy szablonem HLS. Dokument ten, jest dołączany do zgłoszenia jako załącznik. Następnie, w zakładce „Budget” (Rys.6.21), wykonuje się wycenę prac konsultanta zgodnie z wytycznymi określonymi w dalszej części dokumentacji. Po zakończeniu tych czynności zadanie jest zamykane, a konsultant przechodzi do zakładania zgłoszeń związanych z audytem oraz oszacowania czasu pracy jaki będzie przez niego wymagany do opracowania kompletnej specyfikacji funkcjonalnej modyfikacji. Dokument HLS wraz z podstawowymi informacjami o modyfikacji trafia do wyszukiwarki modyfikacji, która w oparciu o algorytm przetwarzania języka naturalnego (NLP) pozwala na wyszukanie podobnych prac wykonanych w poprzednich projektach wdrożeniowych. Użytkownik w momencie opracowywania specyfikacji powinien skorzystać z wyszukiwarki w celu weryfikacji, czy podobny materiał/modyfikacja nie została już wcześniej wykonana dla innego klienta lub projektu. Na chwilę obecną synchronizacja z InfoSearch jest wyzwalana przez administratora, ale docelowo zakłada się, że każda z modyfikacji po jej zakończeniu (wszystkie zadania zamknięte), tzn. wykonano ostatni krok 8XX został zamknięty lub zakończony będzie automatycznie przesyłana do wyszukiwarki.

2. Audyt - PL AUDIT (ID zadania: 2XX)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Zadanie jest kierowane na audyt do Architekta Projektu w momencie, kiedy Konsultant przygotował dokument specyfikacji HLS dla oczekiwanej zmiany systemu. Architekt ma za zadanie zweryfikować proponowaną modyfikację poprzez:

- możliwość alternatywnego rozwiązania na bazie standardu systemu (ewentualna eliminacja modyfikacji),
- możliwość alternatywnego rozwiązania mającego na celu ewentualne uproszczenie proponowanej zmiany,
- weryfikację wpływu na spójność rozwiązania z perspektywy między obszarowej,
- identyfikację ryzyk związanych z ewentualnym przygotowaniem zmiany.

Podczas audytu Architekt weryfikuje dokument „HLS” oraz sprawdza, czy została wprowadzona wycena prac przez konsultanta Rys.6.21, w przypadku braku uzupełnia budżet lub przywraca zadanie na przygotowanie specyfikacji HLS. Po zatwierdzeniu dokumentu przez Architekta zakładane jest zadanie na wycenę i wstępny audyt od strony programistycznej.

3. Audyt i wycena programowania (HLS) - PL WORK ESTIMATION (ID zadania: 3XX)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Po pozytywnym zakończeniu audytu Architekta Projektowego, zadanie jest przez niego kierowane na kolejną „PLIC RDP/QIC PROG EXT” w celu wyceny programistycznej. Audyt Architekta musi być zakończony przed założeniem tego zadania. Uzgodniono, że działania dotyczące wyceny prac programistycznych będą podejmowane przez Architektów Technicznych odpowiedzialnych za koordynację prac programistów. Efektem pracy Architekta Technicznego powinna być szczegółowa wycena prac programistycznych, informacja zwrotna o ewentualnych ryzykach, ograniczeniach oraz sposobie podejścia do budowy rozwiązania. Jeśli opisana przez Konsultanta dokumentacja jest niewystarczająca, to zadanie wraca do Audytora celem wyjaśnienia nieścisłości.

4. Specyfikacja funkcjonalna modyfikacji - PL SPECIFICATION (ID zadania: 1XX)

Czas realizacji: Potwierdzony z Project Managerem (PM). Termin realizacji uzupełniony w polu "Termin realizacji". Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Jeżeli podjęto decyzję o realizacji modyfikacji, zadanie to rozpoczyna się od stworzenia szczegółowego dokumentu specyfikacji modyfikacji (SM), który musi być uzgodniony z klientem i spełniać wszystkie wymagania formalne. Wymagania formalne dla dokumentu Specyfikacji są określone szablonem, który przed rozpoczęciem prac jest przedstawiany klientowi do wglądu. Po zaakceptowaniu specyfikacji przez klienta Konsultant dołącza dokument do zgłoszenia i uzupełnia informację dotyczącą wyceny testów i ewentualnego czasu, który jest potrzebny na prace odbiorowe z klientem (Rys.6.21). Następnie zamyka zadanie dotyczące wykonania specyfikacji i zakłada zadanie na audyt.

5. Audyt specyfikacji modyfikacji - PL AUDIT (ID zadania: 25X)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Podobnie jak w poprzednim etapie prac nad specyfikacją wysokiego poziomu, audyt przeprowadza Architekt. Zadanie jest kierowane do odpowiedniej kolejki w zależności od obszaru wdrażanego systemu. Architekt sprawdza dokumentację pod względem spójności funkcjonalnej, zgodności formalnej i poprawności wyceny prac konsultingowych. W przypadku odrzucenia dokumentu zadanie jest zamykane, a następnie zakładane ponownie z odpowiednią klasyfikacją błędu. W przypadku akceptacji lub warunkowego dopuszczenia do dalszej obsługi Architekt zamyka zadanie na audyt i otwiera zadanie na audyt i wycenę prac programistycznych. Możliwe jest, że Architekt Projektu w przypadku braku posiadanej wiedzy z zakresu przedmiotowej specyfikacji może posiłkować się wiedzą innej osoby z danego zespołu merytorycznego. W tej sytuacji zakłada task na audyt dla innej osoby z właściwą adnotacją.

6. Audyt i wycena programistyczna - PL WORK ESTIMATION (ID zadania: 3XX)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Po przejściu audytu, z perspektywy funkcjonalnej oraz spójności rozwiązania od strony międzyobszarowej, zadanie kierowane jest do audytu i wyceny programistycznej. Architekt Techniczny weryfikuje rozwiązanie od strony możliwości realizacji i ewentualnych ryzyk związanych z realizacją rozwiązania. Natomiast wycena programistyczna obejmuje szacowany czas wykonania oraz orientacyjną datę realizacji na podstawie aktualnego „zatarowania” zespołu programistycznego i aktualną kolejkę programistyczną. Na rysunku Rys.6.21 przedstawiono nowo opracowany ekran do realizacji wycen związanych z obsługą modyfikacji. Narzędzie umożliwia śledzenie kosztów prac programisty i konsultanta na każdym etapie wytwarzania modyfikacji. W ekranie uprawnieni użytkownicy mogą zdefiniować wycenę dla każdego typu pracy:

- wycena programistyczna - PRG - Programming wraz z elementami składowymi kosztu:
 - administracja (ADM)
 - przegląd / audyt kodu (CODEREV)
 - wysyłka do klienta (DEL)
 - czas na analizę - wycena prac (EST)
 - programowanie (PRG)
- wycena konsultingowa (CONS) wraz z elementami składowymi kosztu:
 - wykonanie specyfikacji (COSPEC)
 - wykonanie testów modyfikacji (COTEST)
 - odbiór modyfikacji (CODEL)
- wycena Kierownika Projektu (FINAL)
 - wycena kosztowa (INTCAL)
 - wycena przychodowa (SALECAL)

Jeśli specyfikacja umożliwiła wycenę oraz wstępne zaplanowanie możliwości realizacji Architekt Techniczny lub Audytora, a w przypadku audytów warunkowych Konsultant zakłada zadanie na zgodę wykonania modyfikacji do Kierownika Projektu. W przypadku odrzucenia dokumentu specyfikacji, na etapie wyceny programistycznej, zadanie jest zamykane i otwierane jest nowe w celu audytu poprawionego dokumentu.

7. Zgoda na programowanie - PL PM APPROVAL (ID zadania: 4XX)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Po wykonaniu procesu audytu specyfikacji oraz wyceny konsultanta i programisty, zakładane jest zadanie dla Kierownika Projektu (PM) danego projektu. Oznacza to, że dokument specyfikacji oraz wyceny zostały przygotowane i należy zatwierdzić je z klientem w przypadku usługi dodatkowej lub zweryfikować odstępstwo względem planu budżetowego i przygotować wewnętrzny dokument zmiany (change request) w przypadku modyfikacji objętej umową dla której nastąpiło odchylenie od budżetu ponad 20 procent pierwotnie planowanego czasu lub modyfikacji nie było w ogóle w zakresie projektu. Zadanie jest zatrzymane do realizacji do momentu uzyskania akceptacji Kierownika Projektu.

8. Programowanie - PL PROGRAMMING (ID zadania: 5XX i 5X1)

Czas realizacji: Zgodnie z budżetem modyfikacji, w terminie ustalonym przez Lidera/Kierownika Zespołu Programistów (KZP), w przypadku zadań na poprawkę jest to 3 dni od daty zgłoszenia.

Po akceptacji klienta i weryfikacji budżetu, Kierownika Projektu zakłada zadanie na programowanie. Zlecenie pracy kierowane jest do odpowiedniej kolejki programistycznej, która na bieżąco jest przeglądana przez KZP. Jeśli jednak na zadaniu nie jest ustawiony „termin realizacji” oraz zadanie nie zostało przyjęte do realizacji to system generuje monit do wszystkich KZP w celu podjęcia zadania. Zadanie dodatkowe (54X) może być zakładane przez Konsultanta w przypadku nieprawidłowości wykrytych podczas testów modyfikacji.

9. Audyt kodu - PL CODE REVIEW (ID zadania: 55X)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji audytu.

Po zakończeniu prac programistycznych zadanie na przegląd kodu zakładane jest przez Programistę. Ma ono na celu weryfikację, czy kod spełnia wymagane standardy dotyczące wewnętrznej polityki jakości. W przypadku niezgodności, błędów czy nienależytej staranności, jest zakładane zadanie na poprawki dla programisty odpowiedzialnego za zgłoszenie. Audyt kodu nie jest zadaniem obowiązkowym w procesie i KZP może przekazać zadanie do testów konsultanta bez weryfikacji lub na podstawie wrywkowej weryfikacji kodu. Krok ten jest istotny w przypadku prac realizowanych przez mniej doświadczonych programistów.

10. Testy modyfikacji - PL TESTING (ID zadania: 6XX i 6X1)

Czas realizacji: 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji testów modyfikacji.

Zadanie testowe kierowane jest do Konsultanta, który odpowiada za zgłoszenie, celem przeprowadzenia weryfikacji poprawności działania modyfikacji. Jeśli testy wykażą nieprawidłowości, Konsultant tworzy kolejne zadanie na programowanie (poprawki programistyczne). Konsultant odpowiedzialny za zgłoszenie decyduje o ewentualnym przekazaniu zadania na testy dla innego konsultanta (konsultant zadaniowy), dotyczy to przede wszystkim sytuacji, w której nie może on dotrzymać 7-dniowego okresu na wykonanie testów. W sytuacji kiedy modyfikacja spełnia wymagania specyfikacji poprzez prawidłowe przejście scenariusza testowego, Konsultant zakłada jednocześnie 2 zadania:

- (a) przegląd serwisowy modyfikacji - PL SUPERVISING

(b) instalacja modyfikacji - PL INSTALLATION.

11. Serwisowy przegląd modyfikacji - PL SUPERVISING (ID zadania: 650)

Czas realizacji: 14 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji przeglądu serwisowego.

Jeśli modyfikacja i specyfikacja jest zrozumiała przez zespół serwisu oraz spełnia wszystkie formalne wymagania w zakresie polityki jakości, zadanie jest zamykane. Oznacza to, że modyfikacja jest wykonana poprawnie i serwis przejmie ją w sposób automatyczny pod opiekę w momencie wejścia w życie umowy serwisowej. Działanie to ma na celu:

- (a) uproszczenie procesu przekazania projektu wdrożeniowego pod opiekę serwisową,
- (b) równoległe do wdrożenia przygotowywanie zespołu serwisowego do przejęcia klienta. Zespół serwisowy na bieżąco poznaje złożoność konfiguracji klienta i nowe rozwiązania.
- (c) identyfikację ewentualnych ryzyk związanych z przyjętym rozwiązaniem z perspektywy Pionu Serwisu.

12. Instalacja Modyfikacji - PL INSTALLATION (ID zadania: 7XX)

Czas realizacji: Do 7 dni. Po tym okresie następuje monit o opóźnieniu w zakresie realizacji instalacji modyfikacji w środowisku klienta.

Zadanie skierowane na instalację zmian na bazach klienta, wykonywane jest przez programistę, a po zakończeniu prac jest przekazywane do osoby realizującej wysyłkę do klienta.

13. Odbiór modyfikacji - PL CUSTOMER ACCEPTANCE (ID zadania: 8XX)

Czas realizacji: 14 dni. Po tym okresie następuje monit o opóźnieniu w zakresie odbioru modyfikacji.

Po zainstalowaniu zmian Konsultant kontaktuje się z klientem, aby omówić i zatwierdzić modyfikację. Zadanie można zamknąć po uzyskaniu akceptacji od klienta. Odbiór modyfikacji powinien być wykonany co najmniej z mailowym potwierdzeniem klienta.

Powyższy proces wspierany jest systemowo poprzez oprogramowane odpowiednie automaty i funkcjonalności dostępne w systemie IFS, który zobrazowany został na rysunku Rys.6.22. Natomiast raportowanie w zakresie statusu realizacji jest obsługiwane na dedykowanych raportach opartych o dane w większości przechowywane w tabeli z zadaniami. Szczegóły dotyczące raportowania zadań zostały opisane w poprzednim rozdziale, który dotyczył zmian w modelu planowania prac programistów w naszej organizacji.

Journal	Description	Business Objects	Case Overview	Contacts	Support Info	Customer Info	SLA	Budget	Customization Project	Terms of Service (Read Only)		
Pokaż: <input type="checkbox"/> Zadania w real. <input type="checkbox"/> Moje zadania <input type="checkbox"/> Wg. term. real. <input type="checkbox"/> Należąco <input type="button" value="Odśwież"/>												
ID zad.	Status	Termin real.	Uwaga	Działanie	Activity Desc	Najpóźniej...	Typ	ID org.	Wykonanie	Utworzone przez	First Close D.	Opikun
100	Zamknięte	01.06.2024 00:...	Specification preparation	36	PL SPECIFICATION	29.05.2024	RnD Department	PLIC CONS	Customization...	TOGEPL	14.06.2024 1...	Tomasz
200	Zamknięte	14.05.2024 12:...		34	PL AUDIT	10.05.2024	Consulting	PLIC CONS	Customization...	TOGEPL	12.05.2024 1...	Krzysztof
300	Zamknięte	07.05.2024 00:...	Technical audit of specifications/solution...	32	PL WORK ESTIMATION	02.05.2024	RnD Department	PLIC RDP	System audit	TOGEPL	07.05.2024 1...	Rafał Wł
310	Zamknięte	07.05.2024 00:...	Technical audit of specifications/solution...	32	PL WORK ESTIMATION	26.04.2024	RnD Department	PLIC RDP	System audit	TOGEPL	17.06.2024 0...	Rafał Wł
400	Zamknięte	10.05.2024 00:...	Software specifications	37	PL PROGRAMMING	08.05.2024	RnD Department	PLIC CONS	Customization...	TOGEPL	14.06.2024 1...	Tomasz
401	Zamknięte	24.06.2024 00:...	PII approval to implement modifications	51	PL PII APPROVAL	20.06.2024	Consulting	PLIC CONS	Customization...	RAWLPL	17.06.2024 1...	Adam D
500	Zamknięte	25.06.2024 00:...	Software specifications	37	PL PROGRAMMING	19.06.2024	RnD Department	PLIC RDP	Customization...	ADDOPL	08.07.2024 1...	Jakub P
501	Zamknięte	13.09.2024 00:...	Software specifications	37	PL PROGRAMMING	10.09.2024	RnD Department	PLIC RDP	Customization...	KRSZPL	09.09.2024 1...	Jakub P
502	Zamknięte	20.09.2024 00:...	Software specifications	37	PL PROGRAMMING	19.09.2024	RnD Department	PLIC RDP	Customization...	TOGEPL	20.09.2024 0...	Jakub P
503	Przekazane do kolejki	26.09.2024 00:...	Software specifications	37	PL PROGRAMMING	24.09.2024	RnD Department	PLIC CONS	TOGEPL			Tomasz
550	Zamknięte	12.07.2024 00:...	Code Review	98	PL CODE REVIEW	10.07.2024	RnD Department	PLIC RDP	System audit	JAPAPL	08.07.2024 1...	Rafał Wł
551	Przekazane do kolejki	13.09.2024 00:...	Code Review	98	PL CODE REVIEW	11.09.2024	RnD Department	PLIC CONS	JAPAPL			Jakub P
600	Zamknięte	15.07.2024 00:...	Testing of modifications/solutions	39	PL TESTING	11.07.2024	RnD Department	PLIC CONS	Customization...	JAPAPL	12.07.2024 0...	Tomasz
601	Zamknięte	16.09.2024 00:...	Testing of modifications/solutions	39	PL TESTING	12.09.2024	RnD Department	PLIC CONS	Customization...	JAPAPL	17.09.2024 1...	Tomasz
700	Zamknięte		Handling shipment to the customer	45	PL INSTALATION		RnD Department	PLIC RDP	Customization...	TOGEPL	18.07.2024 0...	Maciej S
701	Zamknięte		Handling shipment to the customer	45	PL INSTALATION		RnD Department	PLIC RDP	Customization...	TOGEPL	11.09.2024 1...	Maciej S
702	Przekazane do kolejki		Handling shipment to the customer	45	PL INSTALATION		RnD Department	PLIC CONS	TOGEPL			Tomasz

Rys. 6.22. Repozytorium zadań dla zgłoszenia - w procesie obsługi zmiany [85]

Podsumowując efekty wynikające ze zmian w procesie planowania prac programistycznych dla modyfikacji, zadań serwisowych, zadań R&D oraz usług dodatkowych organizacja osiągnęła wymierne efekty poprzez przyspieszenie procesu produkcji modyfikacji. Przedstawione wyniki dotyczą sześciomiesięcznego okresu próbnego podczas którego zespoły programistyczne realizowały zadania ze wszystkich powyższych grup w ponad piętnastu równoległe prowadzonych projektów. Średnia ilość zadań spływających do zespołu wynosiła pomiędzy 5 a 10 zgłoszeń w zakresie modyfikacji dotyczących projektów wdrożeniowych. Średni czas dostarczenia gotowego zadania programistycznego przed zmianą w procesie planowania i obsługi modyfikacji wynosił około czterech tygodni od momentu jego przekazania na kolejkę programistyczną. Na chwilę obecną po 6 miesięcznym okresie próbnym czas dostarczenia skrócił się do około trzech tygodni. W połączeniu z elastycznością w zakresie podejmowania decyzji o kolejności prac programistycznych dało to znaczną wartość dodaną w zakresie decyzyjności i wpływu na tempo oraz priorytety w zakresie produkcji. W analizowanym okresie (kwartał Q2 oraz kwartał Q3 w roku) średnia wartość produkcji w toku nigdy nie przekroczyła 5 tygodni. Prace badawcze będą kontynuowane w kolejnych okresach, gdzie istnieje ryzyko, że zadań będzie znacznie więcej (kwartał Q4 oraz kwartał Q1), kiedy znaczna część klientów z naszego portfela projektowego podejmuje decyzję o starciu produkcyjnym. Okres ten może mieć istotny wpływ na odchylenia od założeń ponieważ w czasie stabilizacji generowana jest bardzo duża ilość zgłoszeń w zakresie poprawek, ewentualnych zadań rozwojowych lub wniosków dotyczących wsparcia.

6.5. Faza IV

6.5.1. Szablon scenariusza rozruchu produkcyjnego jako narzędzie wspierające zespół wdrożeniowy

Scenariusz rozruchu produkcyjnego, to szczegółowy plan działań, który musi zostać podjęty, aby skutecznie uruchomić nowy system w środowisku produkcyjnym. Scenariusz rozruchu obejmuje wszystkie etapy przygotowań, migracji danych, testowania, uruchomienia i wsparcia powdrożeniowego. Dokument ten opisuje m.in. proces przejścia z poprzedniego systemu na wdrażany system lub nową wersję oraz integrację z innymi systemami. Kluczowymi elementami scenariusza są działania związane z przygotowaniem infrastruktury IT, migracją danych, testami przedprodukcyjnymi, konfiguracją systemu oraz weryfikacją gotowości do uruchomienia. Rozruch produkcyjny obejmuje także wsparcie dla użytkowników, zarówno na poziomie technicznym, jak i operacyjnym. W dokumencie powinien być zdefiniowany harmonogram działań ze szczegółowością do konkretnych godzin i dni, przypisanie odpowiedzialności oraz plan wsparcia powdrożeniowego, co ma na celu zminimalizowanie ryzyka błędów i zakłóceń w funkcjonowaniu systemu po wdrożeniu. Ze względu na złożoność procesu i rozciągłość w czasie istnieje ogromne ryzyko, że niektóre z zadań mogłyby zostać pominięte niezależnie czy dotyczy wdrożenia, upgrade'u, rollout czy re-implementacji. W związku z tym opracowany został uniwersalny szablon scenariusza rozruchu produkcyjnego, który ma za zadanie zabezpieczyć zespół przed nieoczekiwanymi zadaniami i ograniczyć ryzyko wystąpienia dodatkowych prac do niezbędnego minimum. Zagadnienie dotyczące przygotowania scenariusza jest pod opieką:

- Kierownika Projektu
- Architekta Projektu
- Konsultantów Wiodących
- Architekta Technicznego (o ile rola została powołana w projekcie)

Przedstawiony poniżej szablon scenariusza rozruchowego został podzielony na obszary merytoryczne według odpowiedzialności. Przedstawiony zakres nie obrazuje układu dokumentu, lecz skupia się na zakresie, który powinien zostać pokryty przez Zespół Projektowy wdrożenia systemu ERP. Proces wdrożenia jak również scenariusz rozruchu produkcyjnego jest zdefiniowany w kilku perspektywach:

- perspektywa zarządcza:
 - harmonogram

- zasoby ludzkie
- zakres

- perspektywa techniczna
 - infrastruktura sprzętowa
 - oprogramowanie
 - licencje

- perspektywa danych
 - strategia migracji
 - migracja danych stałych
 - migracja danych transakcyjnych

- perspektywa procesów biznesowych
 - procesy obszarowe
 - procesy między-obszarowe
 - procesy między-spółkowe
 - integracje z systemami zewnętrznymi

Na bazie powyższych perspektyw został zdefiniowany uniwersalny zakres scenariusza rozruchu produkcyjnego dla różnych typów projektów wdrożeniowych systemu ERP w naszej organizacji i obejmuje on co najmniej poniższy zakres.

1. Przygotowanie systemu i infrastruktury

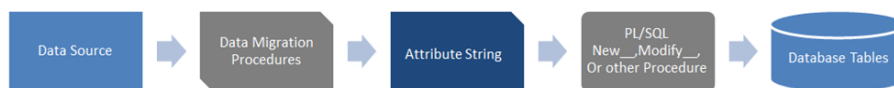
Przygotowanie infrastruktury systemowej jest realizowane w projekcie wdrożeniowym jako jedno z pierwszych zadań w ramach I Fazy wdrożenia. Na podstawie przygotowanych środowisk PROD, CFG i UAT jest definiowany raport z instalacji. Raport ten jest dokumentem, który stanowi podstawę do dalszego zarządzania środowiskiem systemowym przez klienta. Na tej podstawie ustawiane są czynności związane z instalacją aktualizacji/wysyłek, backupowaniem środowisk, kopiowaniem baz czy obsługą procedur bezpieczeństwa. Na podstawie zdobytej praktyki w zakresie powyższych aktywności zespół planuje działania produkcyjne w następujących płaszczyznach:

- **Środowisko produkcyjne:** Proces obejmuje uzgodnienia w zakresie instalacji nowej wersji systemu na wybranej infrastrukturze sprzętowej. W zależności od typu projektu działania będą dotyczyły przygotowania pełnej instalacji od podstaw (upgrade techniczny) lub aktualizacja najnowszych zmian i przygotowanie przed Go-Live. Zostają zainstalowane i uruchomione wszystkie niezbędne usługi wspierające i komponenty dotyczące połączeń z systemami zewnętrznymi w oparciu o przygotowane na środowisku testowym rozwiązanie. Kluczowe jest zapewnienie zgodności z wymaganiami technicznymi i specyfikacją systemu.
- **Infrastruktura IT:** W zakresie infrastruktury ustawienia bazowe realizowane są w Fazie I. Natomiast w Fazie II w większości projektów wdrożeniowych weryfikowane są zagadnienia:
 - weryfikacja zakresu wymaganych komponentów do realizacji procesów klienta
 - weryfikacja zakresu wymaganych licencji
 - sizing infrastruktury sprzętowej - weryfikacja
 - identyfikacja dodatkowego oprogramowania i urządzeń peryferyjnych. Na tej podstawie ustalane są czynności związane ze zmianami w infrastrukturze (zwiększenie mocy i wielkości serwerów), połączenie ze środowiskiem produkcyjnym dodatkowych narzędzi czy urządzeń.
- **Procedura przygotowania kopii zapasowej i ewentualnego odtworzenia bazy:**

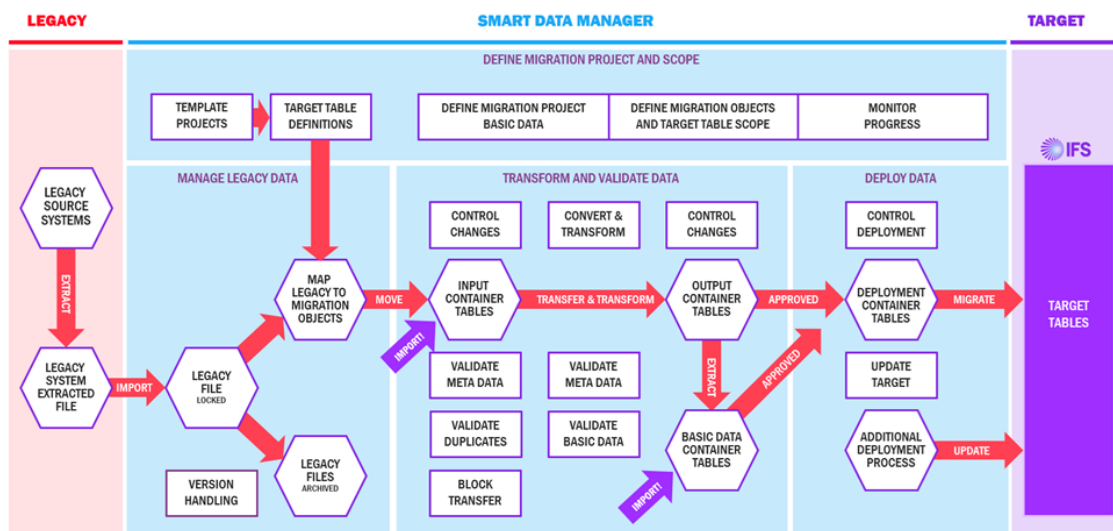
Jednym z ważniejszych zagadnień na etapie planowania rozruchu produkcyjnego jest opracowanie mechanizmów umożliwiających przywrócenie obecnie użytkowanego systemu na wypadek nieudanej próby startu. Dlatego też, zanim zostanie podjęta migracja danych transakcyjnych, należy przygotować pełną kopię zapasową zarówno obecnego jak i docelowego systemu ERP. Kopie te powinny być przechowywane w bezpiecznym miejscu, a ich integralność powinna być sprawdzona. Kopia zapasowa jest kluczowym elementem zapewniającym ochronę przed potencjalnymi problemami, które mogą pojawić się podczas realizacji scenariusza rozruchu produkcyjnego.

2. Migracja danych

- **Strategia migracji:** Dokument opracowywany przez zespół w Fazie II projektu ze szczególnym uwzględnieniem klasycznych wdrożeń oraz rolloutów systemu ERP. Strategia migracji w przypadku tych projektów uwzględnia dostępne w systemie narzędzia IFS Data Migration Tool, który zawiera zestaw narzędzi w postaci predefiniowanych procedur migracji przedstawionych na schemacie Rys.6.23.



Rys. 6.23. IFS Data Migration Tool



Rys. 6.24. Smart Data Manager

Innym rozwiązaniem jakie oferuje IFS w najnowszej wersji jest rozbudowane narzędzie Smart Data Manager (SDM), które umożliwia zarządzanie zaawansowaną migracją danych. Szczególnie istotne z perspektywy projektów, gdzie skala migracji danych uniemożliwia nadzorowanie procesu standardowymi półautomatycznymi metodami. Schemat Rys.6.24 przedstawia modelowy proces zarządzania danymi w oparciu o SDM. Niezależnie od przyjętej metody migracyjnej najważniejsze jest uzgodnienie schematów danych do migracji (szablony danych). Jeżeli ten etap mamy za sobą to w kolejnym etapie według SDM należy wykonać zabezpieczenie plików wzorcowym i przejść do kroku polegającego na mapowaniu obiektów do migracji (MAP LEGACY TO MIGRATION OBJECTS) z polami w systemie. SDM umożliwia zdefiniowanie struktury tabel dla danych źródłowych (INPUT CONTAINER TABLES), które będą w oparciu o mapowanie transportowane (TRANSFORM AND VALIDATE DATA) do wynikowych tabel testowych (OUTPUT CONTAINER TABLES). Po procesie akceptacji może nastąpić załadowanie danych (DEPOY DATA) do obiektów wynikowych (DEPLOYMENT CONTAINER TABLES) skąd dane mogą zostać przeniesione do wybranego środowiska systemu IFS. Rozwiązanie jest o tyle wygodne, że proces może być wielokrotnie powtarzany, a przełączanie

pomiędzy środowiskami UAT, CFG czy PROD nie powinien przysporzyć większych problemów.

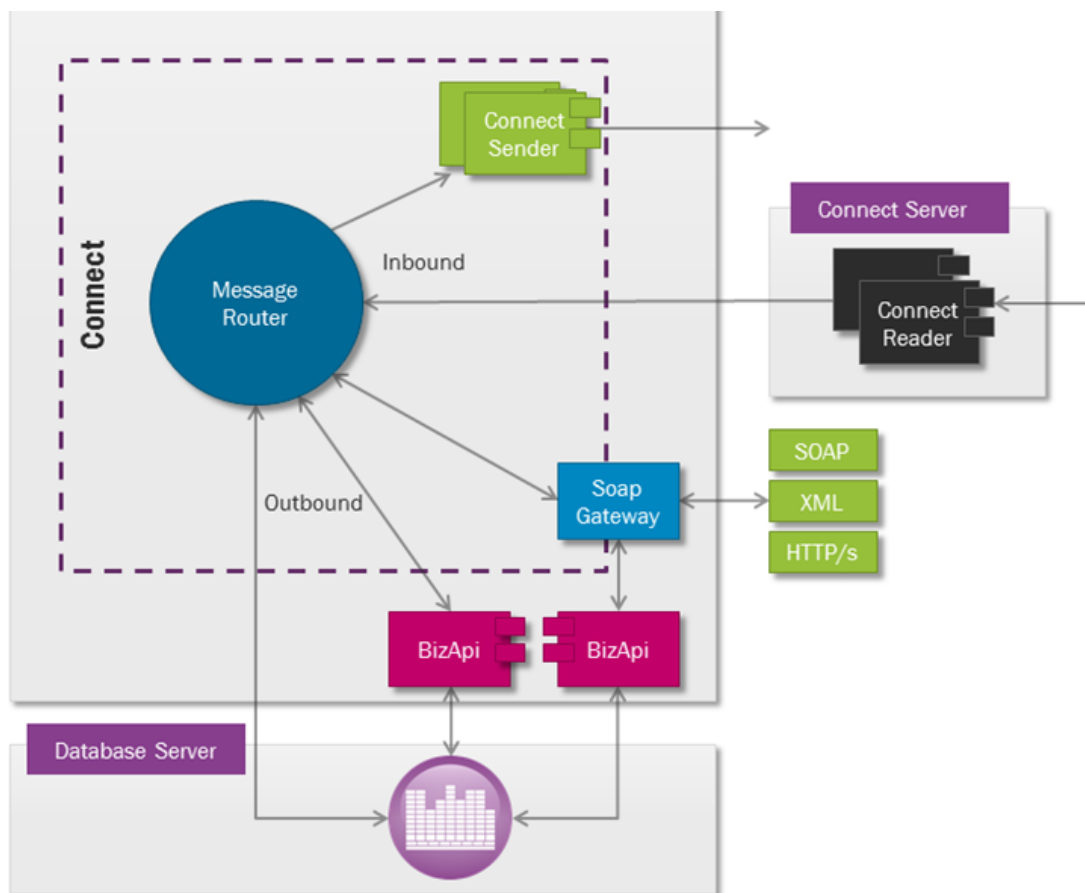
- **Migracja danych stałych i konfiguracji** W wielu projektach wdrożeniowych napotykałoby się na problemy przy starcie produkcyjnym, które były związane z nieprawidłową konfiguracją lub jej brakiem. W wielu przypadkach problemy te były do obejścia jeśli tylko zespół projektowy wykonałby właściwe przeniesienie przetestowanej konfiguracji ze środowiska testowego. Dlatego też w scenariuszu zostały uwzględnione zagadnienia dotyczące przeniesienia parametryzacji: ustawień systemowych, konfiguracji ról i użytkowników, uprawnień obszarowych, schematów workflow czy parametrów związanych z interfejsami zewnętrznymi. Proces ten obejmuje przenoszenie danych konfiguracyjnych oraz weryfikację ich zgodności z nowym systemem. Ważne jest, aby wszystkie istotne parametry były odpowiednio skonfigurowane w celu uniknięcia zakłóceń w działaniu systemu po jego uruchomieniu.
 - **Migracja danych transakcyjnych:** Na podstawie opracowanej strategii i dostępnych narzędziach w procesie realizacji scenariusza rozruchu produkcyjnego realizowana jest migracja, która obejmuje przeniesienie danych operacyjnych, takich jak dane transakcyjne, księgowe, personalne czy historyczne, z poprzedniej wersji systemu do nowej. Proces ten wymaga zaplanowania odpowiedniej sekwencji migracji oraz zastosowania narzędzi do ekstrakcji, transformacji i załadowania danych (ETL). Ważne jest, aby dane były przenoszone z zachowaniem ich integralności oraz bezbłędnie mapowane na nową strukturę. Dotyczy to wszystkich typów projektu z wyjątkiem upgrade technicznego.
3. **Testy przedprodukcyjne - dry run** W zależności od rodzaju projektu testy przedprodukcyjne realizowane w ramach scenariusza rozruchu produkcyjnego mogą wyglądać inaczej, ale wspólną ich cechą jest fakt, że muszą być przeprowadzone na kopii finalnie przygotowanego środowiska produkcyjnego. Bardzo ważne, aby nie mylić tej aktywności i nie dublować jej z testami obszarowymi/ międzyobszarowymi, które realizowane są w ramach Fazy III jako potwierdzenie przygotowania rozwiązania.
- **Testy funkcjonalne:** Testy mają na celu weryfikację, czy wszystkie **kluczowe** funkcje systemu działają prawidłowo. Każdy moduł jest testowany pod kątem zgodności z krytycznymi do działania na start wymaganiami, obejmując takie aspekty jak wprowadzanie danych, oraz ich przetwarzanie. Testy funkcjonalne często odbywają się w środowisku testowym, które odwzorowuje produkcyjne, aby upewnić się, że system działa poprawnie. Ze względu na bardzo ograniczony zakres czasu dostępny dla tej aktywności działania te realizowane są bezpośrednio

na bazie produkcyjnej i w ograniczonym, kluczowym zakresie. Bardzo ważne, aby dla tego działania powstał scenariusz testowy oraz zakładane wyniki dla poszczególnych kroków. Ponieważ czynności te realizowane są w celu ograniczenia ewentualnych przestoju w poszczególnych zespołach operacyjnych. W takim przypadku dane są wycofywane, ale w wielu przypadkach nie zostaną z bazy usunięte tylko zmienią status na „archiwalny”. Dry-run dla procesów krytycznych jest istotny ponieważ podczas tego okresu mogą zostać wyeliminowane problemy natury podstawowej:

- uprawnienia dostępowe
- dostępu do infrastruktury (terminale, drukarki, sieć)
- przepływu danych WE/WY z systemów zewnętrznych. (Zamówienia, faktury)
- **Testy interfejsów oraz spójności międzyobszarowej:** Weryfikacja integracji systemu z innymi aplikacjami oraz modułami, które muszą współpracować z nowym systemem. Celem tych testów jest upewnienie się, że przepływy danych między ERP a systemami zewnętrznymi działają poprawnie, a dane są odpowiednio synchronizowane. Testy obejmują zarówno sprawdzenie poprawności przetwarzania, jak i wydajności komunikacji między systemami. W zależności od projektu może zostać przyjęty inny schemat integracji, ale wzorcowym schematem połączenia jest proponowany przez producenta model (Rys.6.25).

Procedura testowa spójności pomiędzy obszarami lub systemami zewnętrznymi jest opisana szczegółowo w krokach:

- scenariusz testów międzyobszarowych: Scenariusz zakłada realizację wszystkich kroków, które znajdują się na ścieżce krytycznej najważniejszych procesów biznesowych firmy - testy End2End. Planowany przepływ jest proponowany przez Architekta Projektu i musi być zrozumiany, zweryfikowany i zatwierdzony przez Kluczowych użytkowników systemu.
- specyfikacja funkcjonalna dla interfejsu z systemem zewnętrznym: Zgodnie z częścią specyfikacji w zakresie scenariusza testowego, który zawiera listę czynności potwierdzających poprawność realizacji projektowanej integracji. W pierwszym etapie realizujemy weryfikację jednostkową na danych testowych. Po poprawnym przejściu testów przechodzimy do weryfikacji z większą próbą danych rzeczywistych, na podstawie której otrzymamy weryfikację ilościową i jakościową.
- **Akceptacja scenariusza i testów:** Zgodnie ze scenariuszem rozruchu produkcyjnego w określonych momentach na osi czasu podejmowana jest decyzja o przejściu do realizacji kolejnych działań rozruchowych lub decyzja o



Rys. 6.25. IFS connect [85]

przywróceniu dotychczasowego systemu. Jednym z takich obowiązkowych momentów jest zakończenie testów funkcjonalnych i spójności międzyobszarowej. Na tym etapie powinien wydarzyć się proces akceptacji i potwierdzenia możliwości uruchomienia systemu przez kluczowych interesariuszy projektu. Obejmuje to przegląd wyników testów oraz weryfikację, czy wszystkie krytyczne wymagania zostały spełnione. W przypadku wykrycia błędów lub niezgodności wymagane są odpowiednie obejścia lub jeśli to możliwe poprawki przed uruchomieniem systemu.

4. Rozruch produkcyjny

- **Wyłączenie starego systemu:** Ostatnie dni przed zmianą systemu ERP w scenariuszu rozruchu produkcyjnego powinny być rozpisane ze szczegółowością na poziomie godzinowym. Każdy z obszarów zobowiązany jest do opracowania planu rozliczenia otwartych procesów i sposobu ich transferu do nowego systemu. Przed uruchomieniem nowego systemu konieczne jest wyłączenie starej wersji, aby zapobiec równoczesnej pracy na dwóch systemach, co mogłoby prowadzić do konfliktów danych. Proces obejmuje odłączenie dostępu użytkowników, zamknięcie bieżących procesów oraz finalne archiwizowanie danych.
- **Uruchomienie nowego systemu:** Po wyłączeniu starego systemu, nowy system zostaje uruchomiony w środowisku produkcyjnym. Obejmuje to włączenie wszystkich modułów zgodnie z zakresem planowanym do uruchomienia na Go-Live, weryfikację poprawnego działania interfejsów, podłączenie do zewnętrznych systemów oraz testowanie wydajnościowe w rzeczywistych warunkach. Proces startu produkcyjnego jest monitorowany przez zespół wdrożeniowy, aby natychmiast reagować na ewentualne problemy.

5. Wsparcie powdrożeniowe

- **Asysta po starcie:** W pierwszych dniach po uruchomieniu nowego systemu zapewniona jest ścisła asysta techniczna i operacyjna. Zespół wsparcia monitoruje działanie systemu, odpowiada na pytania użytkowników oraz pomaga w rozwiązywaniu problemów operacyjnych. Celem jest zapewnienie stabilnego działania systemu i minimalizowanie zakłóceń w pracy.
- **Pierwsza linia wsparcia:** Pierwsza linia wsparcia obejmuje zespół odpowiedzialny za natychmiastowe rozwiązywanie prostych problemów użytkowników, takich jak problemy z logowaniem, dostępem do danych, obsługą funkcji systemu czy nieprawidłowościami dotyczącymi podstawowej konfiguracji. Zespół ten często obejmuje personel techniczny,

kierowników liniowych odpowiedzialnych za przygotowanie i działanie systemu zgodne z instrukcją stanowiskową, dedykowanych konsultantów wdrożeniowych.

- **Druga linia wsparcia:** Druga linia wsparcia to bardziej zaawansowani specjaliści techniczni, którzy zajmują się bardziej złożonymi problemami, takimi jak błędy systemowe, problemy z integracjami, czy optymalizacją wydajności. Zespół ten współpracuje z dostawcami oprogramowania oraz działem IT w celu szybkiego rozwiązywania problemów.
- **Monitoring i poprawki powdrożeniowe:** Po uruchomieniu systemu prowadzony jest stały monitoring jego działania, aby wykrywać potencjalne problemy z wydajnością, stabilnością lub bezpieczeństwem. W razie potrzeby są wprowadzane poprawki, które mogą obejmować zarówno korekty błędów, jak i wprowadzenie usprawnień na podstawie bieżących doświadczeń użytkowników.

Harmonogram godzinowy dla Go-live

- **Przygotowania techniczne i migracja danych:** Obejmuje finalne przygotowanie infrastruktury technicznej, ewentualne tymczasowe powiązania z obecnie wykorzystywanymi systemami oraz przeprowadzenie pełnej migracji danych z systemów źródłowych do nowego środowiska produkcyjnego. Wszystkie dane muszą być dokładnie zweryfikowane i sprawdzone, aby zapewnić ich poprawność. Jednym z kluczowych czynników umożliwiających rozpoczęcie pracy produkcyjnej jest formalne potwierdzenie wszystkich bilansów otwarcia (magazynu, finansów, otwarte zamówienia klienta i zakupu, otwarte zlecenia produkcyjne, zapotrzebowania) zarówno w zakresie wartościowym jak i ilościowym. Proces ten obejmuje także ostateczne dostosowanie systemu do specyfikacji wymagań klienta oraz zapewnienie pełnej gotowości technicznej do startu produkcyjnego.
- **Dry run:** Ostatnia runda testów weryfikujących stabilność systemu oraz jego zgodność z wymaganiami funkcjonalnymi. W tym czasie wykonywane są także ostateczne konfiguracje parametrów systemowych, aby zoptymalizować działanie i dostosować go do rzeczywistych warunków produkcyjnych. W scenariuszu powinien zostać zawarty zakres koniecznych do wykonania działań w zakresie testów, odpowiedzialność za ich realizację oraz czas, w którym te czynności muszą być zrealizowane.
- **Go-live (start produkcyjny):** Z perspektywy wdrożeniowej pierwsza produkcyjna transakcja w systemie ERP jest potwierdzeniem rozpoczęcia pracy na nowym systemie. Scenariusz rozruchu, jeśli to tylko możliwe również na etapie pracy produkcyjnej powinien zawierać możliwość powrotu do pracy w dotychczasowym systemie.

Oczywiście jest to zależne od możliwości technologicznych obecnych rozwiązań systemowych, ale doświadczenia wdrożeniowe identyfikują sytuacje, kiedy kilka dni po starcie identyfikowane są kluczowe braki w zakresie danych lub spójności rozwiązania. Te odstępstwa mogą być na tyle istotne, że poprawa danych czy rozwiązania może zająć nieakceptowaną z perspektywy biznesu ilość czasu. W takiej sytuacji jedynym dobrym rozwiązaniem może być przywrócenie produkcyjne dotychczasowego systemu. W trakcie procesu rozruchowego Zespół Projektowy monitoruje system, aby reagować na wszelkie problemy w czasie rzeczywistym, a także zapewnia wsparcie operacyjne dla użytkowników.

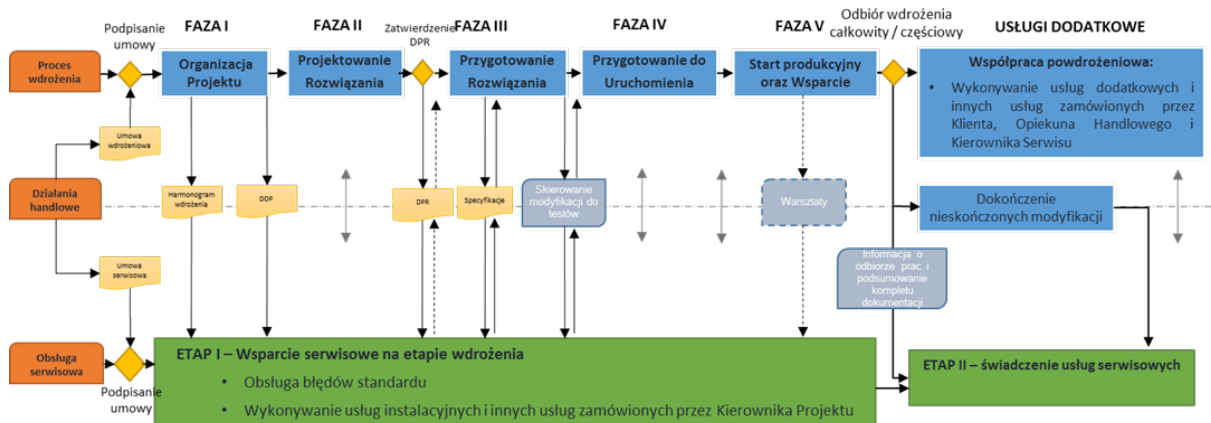
Przedstawiony zakres scenariusza produkcyjnego ukazuje, jak wiele zagadnień musi być przez zespół wdrożeniowy zarządzanych. Ze względów bezpieczeństwa zarówno klienta jak i integratora wszystkie zadania powinny zostać zaplanowane w sposób analogiczny do pozostałych projektów wdrożeniowych, a co istotniejsze nie pominięte w procesie planowania rozruchu. Dlatego też, założono, że zaproponowany szablon scenariusz rozruchu produkcyjnego jest zbiorem wytycznych, które muszą być zaadresowane podczas planowania Go-Live.

6.6. Faza V

6.6.1. Przekazanie projektu wdrożeniowego do serwisu

Obecnie większość projektów wdrożeniowych typu implementacja, upgrade techniczny lub funkcjonalny realizowanych jest do najnowszej wersji systemu IFS Cloud. Natomiast projekty dotyczące rolloutów realizowane są zgodnie z obecnie wykorzystywaną przez klienta wersją systemu. Zmiana systemu do najnowszej wersji powoduje zmianę w zakresie odpowiedzialności za środowisko instalacyjno-rozwojowe. Tym samym zakres, odpowiedzialność i mechanizmy realizacji opieki serwisowej uległy znaczącej zmianie. Szczególnie istotne są różnice w zakresie opieki nad środowiskiem instalacyjno-rozwojowym, które w aktualnej wersji architektury IFS nazywane jest Build Place. Do wersji IFS Cloud odpowiedzialność za środowisko instalacyjno-rozwojowe była po stronie partnera wdrożeniowo-serwisowego, a klient finalnie otrzymywał gotową paczkę instalacyjną przygotowaną do wgrania na środowiska UAT, CFG, a po testach również na PROD. Natomiast w IFS Cloud to klient w pełni zarządza środowiskiem Build Place i podejmuje decyzje w zakresie:

- dostępu do środowiska,
- przygotowania paczki instalacyjnej,



Rys. 6.26. Współpraca wdrożenia - serwis

- zakresu zmian, które będą pod opieką partnera i własnych zespołów wsparcia.

W związku z tym, że obecnie klient odpowiada za uprawnienia dostępowe do środowiska rozwojowego, pojawia się bardzo poważne wyzwanie dotyczące ryzyka ewentualnych zmian w kodzie, które nie zostały zautoryzowane przez dział serwisu lub wdrożenia. Dlatego też, zostały zaktualizowane umowy serwisowe, które mają na celu umożliwienie przejścia pełnej opieki nad zrealizowanymi przez zespół wdrożeniowymi zmianami CRIM. Zagadnienia podjęte w tej części rozprawy syntetyzują proces przejścia instalacji klienta przez serwis, działania realizowane od Fazy I do ostatnie zostały przedstawione na rysunku 6.26. Przepływy przedstawiają zakres prac, jakie powinny zostać wykonane w trakcie i po zakończeniu wdrożenia w projekcie, dla którego serwis będzie realizował kompletny zakres oferowanych usług.

Wszystkie zaprezentowane na powyższym schemacie kroki zostały opisane w ramach rozdziału 5 i jeśli proces wdrożeniowy realizowany jest zgodnie z opisanymi w rozdziale założeniami to przejście projektu nie powinno przysporzyć żadnej dodatkowej pracy. Jedynym zagadnieniem, które obecnie jest rozważane to zasady wsparcia przez zespół serwisu w ramach działań związanych ze stabilizacją

Założenia podstawowe dla pełnego przejścia projektu wdrożeniowego systemu ERP do serwisu

Komunikacja i dostęp do materiałów projektowych odbywa się na podstawie opracowanego przez wdrożenia szablonu struktury projektowej w aplikacji SharePoint. Na podstawie otrzymanych dostępowych uprawnień pracownicy serwisu mają możliwość wglądu w bieżącą dokumentację projektową. W przypadku projektu realizowanego na bazie repozytorium klienta zespół wdrożeniowy i wskazane osoby z serwisu również mają dostęp do repozytorium projektowego. Jednak po zakończeniu wdrożenia cała dokumentacja musi zostać skopiowana do zasobów naszej organizacji. W przypadku dokumentów dotyczących zmian

standardu systemu proces odbioru realizowany jest zgodnie z przyjętą i opisaną w niniejszym rozdziale procedurą, a narzędziem, które służy za repozytorium danych jest system Service Desk (IFS).

Serwis ma możliwość wglądu w dokumentację wytwarzaną przez zespół wdrożeniowy na przestrzeni całego projektu, ale szczegółowej kontroli podlegają:

- Podpisanie Protokołu Zakończenia Projektu oraz przekazanie wszystkich planowanych do przejścia przez serwis modyfikacji.
- Zamknięcie wszystkich zgłoszeń przypisanych do zespołu wdrożeniowego. W przypadku otwartych zgłoszeń, lista tych zgłoszeń musi zostać przekazana z określeniem terminów ich rozwiązania, które są wyłączone z obsługi serwisowej do momentu ich zamknięcia.
- Dokumentacja techniczna, schematy instalacji.

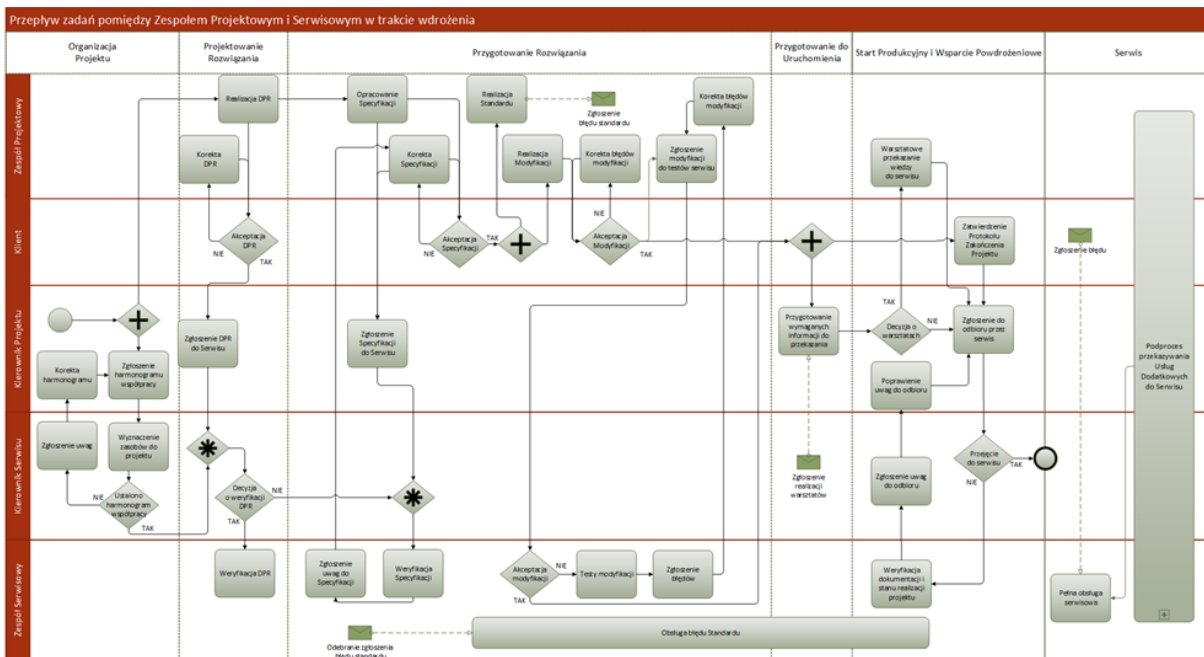
Pozostała dokumentacja jest przekazywana zespołowi serwisu do wglądu na potrzeby usprawnienia zadań związanych z bieżącą obsługą klienta. W skład takiej dokumentacji powinny wchodzić formalne produkty dla wszystkich Faz projektowych i są to m. in. protokoły odbioru, Dokument Definicji Projektu (DDP), dokumenty projektu rozwiązania (DPR), specyfikacje modyfikacji, wykaz zainstalowanych wersji oprogramowania oraz informacje o środowiskach testowych i produkcyjnych.

Czynności realizowane w procesie przekazywania systemu

W trakcie realizacji projektu zespół serwisowy sukcesywnie zapoznaje się z wdrożonym rozwiązaniem, weryfikuje przekazywane dokumenty, bierze udział w testach modyfikacji, a także opiniuje dokumenty projektowe. Konsultanci serwisowi opiniują modyfikacje i testują je przed uruchomieniem produkcyjnym zgodnie z zapisami procedury serwisowej. Szczegółowy schemat przebiegu procesu przejścia procesu pod opiekę serwisową został przedstawiony na rysunku Rys.6.27.

Podsumowując najważniejsze z perspektywy proceduralnej jest przestrzeganie procedury wytwarzania CRIM. Modyfikacje (CRIM) są przekazywane do serwisu w ramach dokładnie określonych czynności, w tym opracowania i weryfikacji specyfikacji, testowania oraz uzyskania akceptacji od klienta. Wszystkie istotne informacje dotyczące modyfikacji, takie jak błędy i poprawki, są przekazywane zespołowi serwisowemu, który jest odpowiedzialny za ich stabilizację po uruchomieniu. W przypadku modyfikacji zrealizowanych niezgodnie z procesem wytwarzania zmian, modyfikacja nie zostanie przejęta przez Zespół Serwisu. Pozostałe czynności realizowane są w ramach procesu wdrożeniowego i nie powinny zaburzyć procedury przeniesienia projektu na serwis.

Z perspektywy formalnej bezpośrednio po starcie produkcyjnym systemu, serwis rozpoczyna wsparcie klienta. W związku z trwającą asystą i procesem stabilizacji systemu



Rys. 6.27. Schemat procesu przekazania projektu wdrożeniowego do serwisu

realizowanymi przez Zespół Projektowy (wdrożenia) serwis koncentruje się wyłącznie na zadaniach związanych z koordynacją błędów standardu systemu oraz gotowych rozwiązań (produktów) opracowanych przez naszą organizację. Po zakończeniu wdrożenia i akceptacji wszystkich prac przez klienta, zespół serwisowy przejmuje pełną odpowiedzialność za obsługę klienta, systemu, dalszy jego rozwój, a wszelkie otwarte modyfikacje lub błędy są zgłaszane do serwisu zgodnie z ustaleniami umowy serwisowej.

6.6.2. Narzędzia wspierające zarządzanie projektem - naučky projektowe (z eng. lessons learnt)

Ostatnia Faza projektu z perspektywy klienta koncentruje się na stabilizacji przyjętego na start rozwiązania systemowego. W naszej organizacji służą do tego: Service Desk (IFS) oraz narzędzia umożliwiające poprawę i rozwój rozwiązania zebrane pod nazwą Build Place. Z perspektywy zarządczej jest to czas na przebrojenie zarówno organizacji wdrożeniowej jak i klienta na współpracę na podstawie umowy serwisowej. Kolejnym zadaniem, jakie stoi przed zespołem wdrożeniowym jest przeprowadzenie analizy projektu i wyciągnięcie nauki z doświadczeń zdobytych w trakcie realizacji wdrożenia. W naszej organizacji zgodnie z przedstawionymi w rozprawie informacjami jednym z kluczowych rozwiązań do prowadzenia projektu jest system ERP IFS. W systemie zebrane są informacje w następujących obszarach:

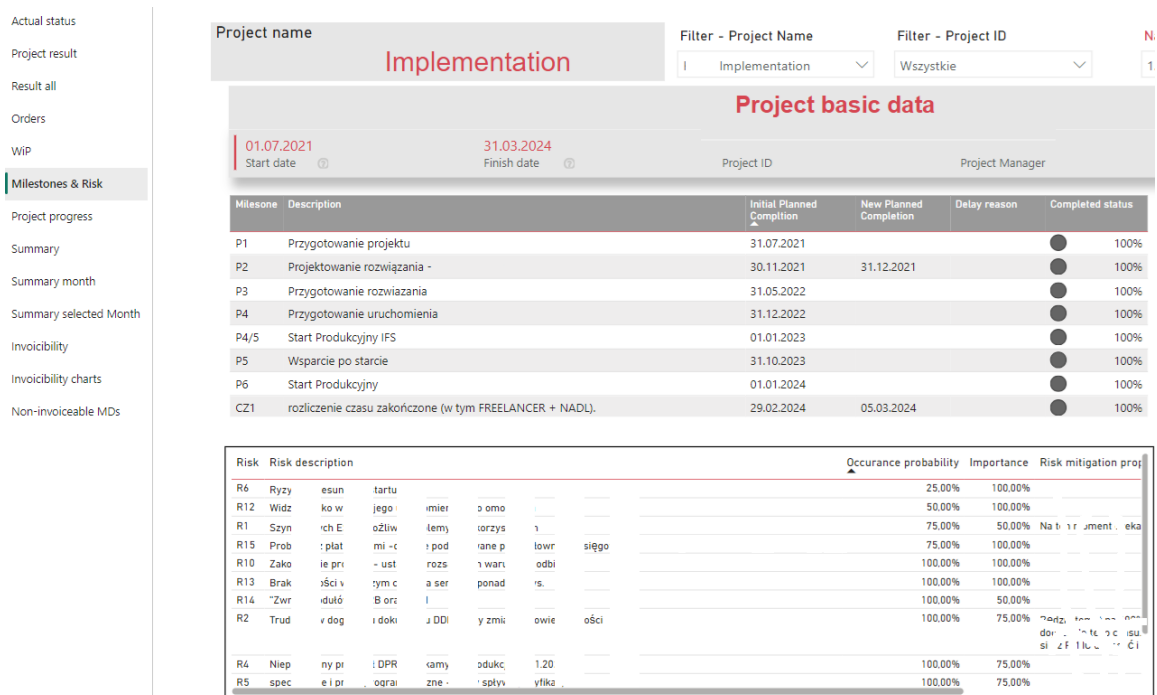
- Zasoby projektowe

- Harmonogram ramowy i szczegółowy
- Zadania projektowe powiązane z harmonogramem projektu
- Dokumenty przychodowe i kosztowe
- Utylizacja czasu pracy
- Rejestr zmian do systemu CR w zakresie CRIM
- Rejestr ryzyk, kamieni milowych oraz informacji potrzebnych do raportowania postępów projektu przed Komitetem Sterującym i Zarządem organizacji

Wiele z powyższych informacji jest wykorzystywana na potrzeby operacyjnego zarządzania i w tym przypadku większość danych jest eksportowana do plików płaskich lub raportowania za pomocą zapytań SQL w postaci tabelarycznej. Lista raportów do wyników płaskich została przedstawiona na rysunku Rys.6.30. Jednak w przypadku raportowania zarządczego, analiz porównawczych większej ilości danych wymaganym narzędziem są hurtownie danych. W naszym przypadku organizacja zdecydowała się, aby system ERP zintegrować z rozwiązaniem oferowanym przez Microsoft. Na bazie zdefiniowanej roli o raz zestawów danych (widoków) dane transportowane są do Power BI, gdzie następuje obróbka danych i ich wizualizacja. Power BI to zaawansowane narzędzie analityczne, które umożliwia przekształcanie danych w cenne informacje za pomocą interaktywnych raportów i pulpitów nawigacyjnych. Integracja ta pozwala na bieżące monitorowanie postępów, kontrolę finansową oraz analizowanie wyników realizacji projektów z perspektywy 'lessons learnt', co jest kluczowe dla optymalizacji przyszłych przedsięwzięć.

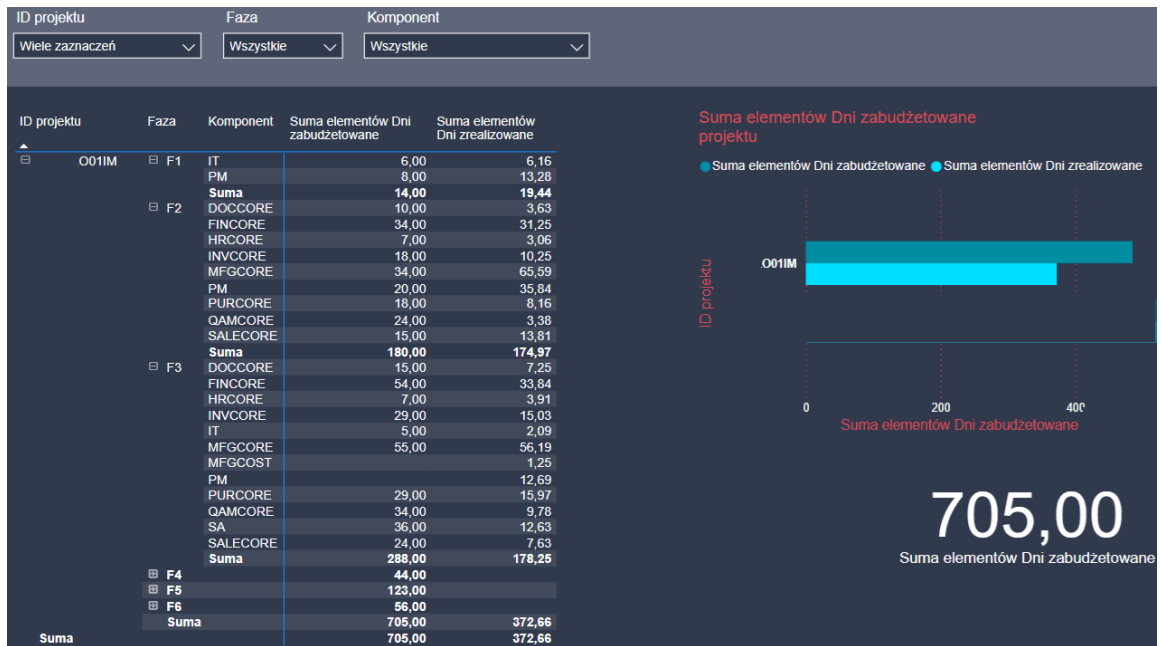
Integracja Power BI z systemem ERP pozwala na automatyczne pobieranie i wizualizację tych danych w formie dedykowanych raportów, które zgodnie z rolą w organizacji dostępne są Zarządu, Kierownikom Zespołów operacyjnych czy Kierownikom Projektów. Dodatkowym atutem jest to, że Power BI jest dostępny z dowolnego urządzenia mobilnego posiadającego możliwość uruchamiania aplikacji mobilnych lub z poziomu przeglądarki.

Jednym z kluczowych mierników efektywności na projekcie jest wydajność zespołu, która w przypadku naszej organizacji mierzona jest m. in. poprzez wskaźnik dni sprzedażowych vs dni ewidencjonowanych na projekt (Rys.6.28).



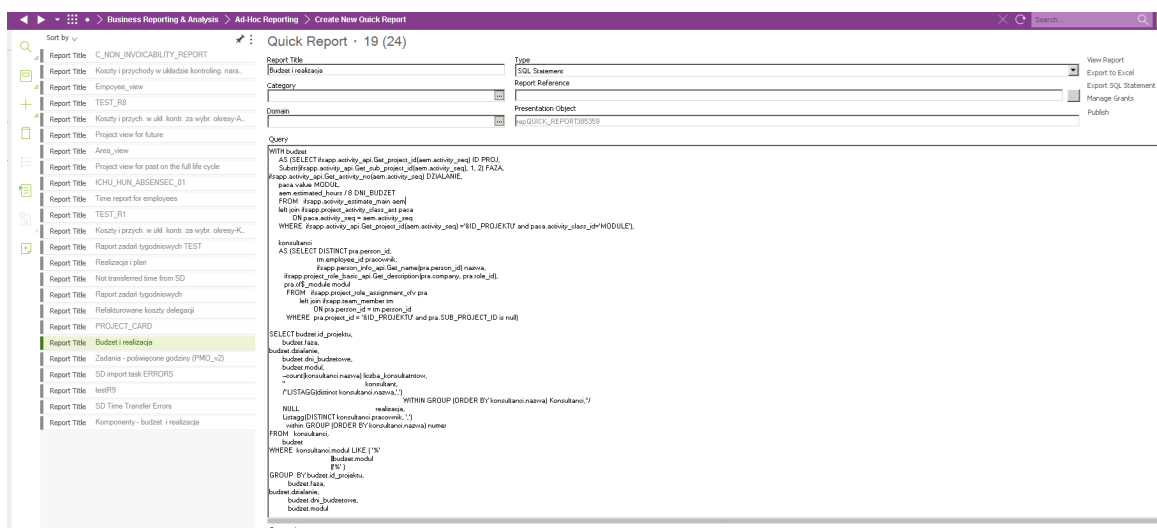
Rys. 6.28. Budżet projektu - analiza nieefektywności

Każdy z uczestników projektu zobowiązany jest do bieżącej ewidencji czasu pracy, natomiast Kierownicy Projektów przy wsparciu Kierowników Zespołów mają obowiązek rozliczyć czas pracy z klientem (projekty Time&Materials) lub porównać czasy rzeczywiste z planowanymi budżetami dla poszczególnych zadań (projekty Fix Price i zadania wyceniane). Na tej podstawie otrzymujemy informacje analityczne dotyczące efektywności każdego z członków zespołu, które w powiązaniu z doświadczeniem projektowym, obszarem wdrożeniowym pozwalają na planowanie działań mogących usprawnić procesy w przyszłości. Wskaźnik tzw. odrzuconego przez klienta lub Kierownika czasu pracy jest kosztem nie tylko zespołu, ale pomniejsza wynik finansowy organizacji. W związku z tym zdecydowano się wprowadzić kolejną miarę związaną z czasem pracy. Czas planowany na zadania, który został oszacowany na etapie ofertowych projektu jest porównywany z rzeczywistymi zapisami w systemie. Działanie to jest realizowane dla czasu zarejestrowanego w systemie (komplety danych zespołu projektowego) oraz czasu potwierdzonego przez Kierowników i korelowane jest z planem sprzedażowym (6.29).



Rys. 6.29. Budżet projektu - wycena vs realizacja

Jest to miara, która pozwala nam szacować skuteczność planowania projektu na etapie sprzedażowym z dokładnością komponentu w ramach modułu dla każdej z faz projektowych. Miara ta pozwala również na mierzenie efektywności poszczególnych członków zespołów i wyciąganie wniosków na przyszłość. Raport był pierwotnie realizowany w oparciu o zapytanie SQL do raportu podręcznego, którego źródło zostało zapisane bezpośrednio w aplikacji poprzez konstruktor zapytań jak na rysunku Rys.6.30.



Rys. 6.30. Budżet vs realizacja

Natomiast Power BI, po połączeniu z ERP, pobiera dane dotyczące kosztów projektowych oraz przychodów związanych z realizacją poszczególnych produktów. Dzięki temu organizacje

mogą w sposób zautomatyzowany śledzić bieżące wykonanie projektu, analizować koszty oraz monitorować zgodność budżetu z założonymi planami. Integracja ta umożliwi również analizę kluczowych wskaźników KPI (Key Performance Indicators), takich jak rentowność projektów, produktywność zasobów czy stopień wykorzystania budżetu. Podejście od strony hurtowni zdecydowanie przyspiesza procesy przygotowania wartościowych wyników z perspektywy analizy danych. Dobrze opracowane i przesłane do Power BI może zastąpić zestaw kilkunastu raportów podręcznych (SQL) Rys.6.30.

```

2 SELECT
3     company_id AS Company_id,
4     emp_no AS Emp_no,
5     ifsapp.COMPANY_PERS_API.Get_Name(company_id, EMP_NO) AS Name,
6     org_code AS Org_code,
7     account_date AS account_date,
8     SUM(internal_quantity) AS internal_quantity,
9     SUM(Sales_quantity) AS Sales_quantity,
10    project_id AS Project,
11    sub_project_id AS Subproject,
12    activity_seq AS Activity_id,
13    short_name AS Activity_name,
14    activity_no AS activity_no,
15    report_cost_code AS Report_cost_code,
16    INTERNAL_COMMENTS AS Internal_comments,
17    INVOICABILITY AS Invoicability,
18    C_CASE_ID AS Case_id,
19    C_SD_TASK_ID AS Task_id,
20    C_OVERHOURS AS C_OVERHOURS
21
22 FROM
23     ifsapp.PROJECT_TRANSACTION
24
25 WHERE
26     UPPER(EMP_NO) LIKE NVL (
27         ifsapp.company_person_api.get_person_emp_no(company_id, ifsapp.PERSON_INFO_API.GET_ID_FOR_USER(ifsapp.FND_SESSION_API.GET_FND_USER())),
28         ifsapp.company_person_api.get_person_employees(company_id, ifsapp.PERSON_INFO_API.GET_ID_FOR_USER(ifsapp.FND_SESSION_API.GET_FND_USER()))
29     )
30
31     AND TO_CHAR(ACCOUNT_DATE, 'yyyy-MM-DD') BETWEEN '&DATE_FROM' AND '&DATE_TO'
32
33 GROUP BY
34     ROLLUP (
35         company_id,
36         emp_no,
37         ifsapp.COMPANY_PERS_API.Get_Name(company_id, EMP_NO),
38         org_code,
39         account_date,
40         project_id,
41         sub_project_id,
42         activity_seq,
43         short_name,
44         activity_no,
45         report_cost_code,
46         C_CASE_ID,
47         C_SD_TASK_ID,
48         INTERNAL_COMMENTS,
49         INVOICABILITY,
50         C_OVERHOURS
51     )
52
53 HAVING
54     (
55         (
56             company_id IS NOT NULL
57             AND emp_no IS NOT NULL
58             AND ifsapp.COMPANY_PERS_API.Get_Name(company_id, EMP_NO) IS NOT NULL
59             AND org_code IS NULL
60             AND account_date IS NULL
61             AND project_id IS NULL
62             AND sub_project_id IS NULL
63             AND activity_seq IS NULL
64             AND short_name IS NULL
65             AND activity_no IS NULL
66             AND report_cost_code IS NULL
67             AND C_CASE_ID IS NULL
68             AND C_SD_TASK_ID IS NULL
69             AND INVOICABILITY IS NULL
70         )

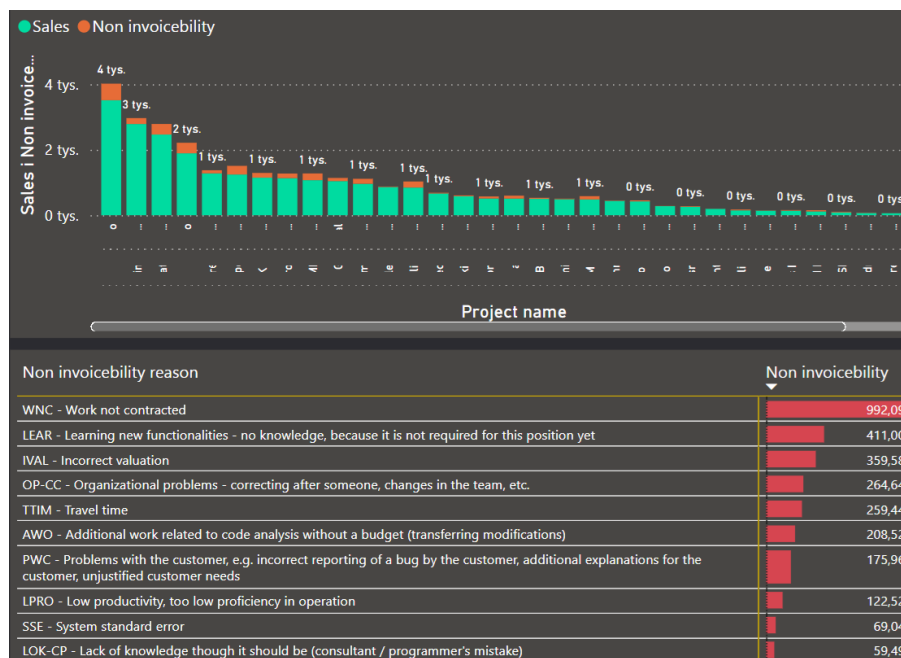
```

Rys. 6.31. Czas pracy - karta projektowa

Lessons learnt (nauki z doświadczeń) to proces systematycznego gromadzenia i analizowania wiedzy nabytej podczas realizacji projektów. Kluczową rolą Power BI w tym procesie jest umożliwienie łatwego dostępu do historycznych danych projektowych, co pozwala na dokładną analizę skuteczności działań, identyfikację problemów oraz sukcesów,

które miały miejsce podczas realizacji wcześniejszych projektów (Rys.6.32). Jedno ze źródeł danych wygląda następująco i dotyczy wyłącznie czasu pracy (Rys.6.31).

Natomiast z perspektywy Power BI dane można prezentować w bardzo przystępny dla użytkownika sposób.



Rys. 6.32. Karta Projektu - lessons learnt

Za pomocą Power BI można stworzyć interaktywne raporty zawierające dane zebrane podczas projektów dotyczące kluczowych wskaźników, takich jak:

- **Skuteczność zarządzania ryzykiem:** Analiza ryzyk i podejmowanych działań korygujących, oparta na danych ERP i Power BI, pozwala zidentyfikować skuteczność tych działań i wprowadzać ulepszenia w strategiach zarządzania ryzykiem.
- **Czas realizacji:** Raporty oparte na danych z harmonogramu planowanego względem rzeczywistych dat zakończenia poszczególnych kamieni milowych. Umożliwiają identyfikację faz projektu, w których występowały opóźnienia, co prowadzi do opracowania strategii ich eliminacji w przyszłości.
- **Odchylenia budżetowe:** Analiza różnic między założonymi budżetami a rzeczywistymi kosztami pozwala na lepsze planowanie finansowe w przyszłych projektach.
- **Wydajność zasobów:** Analiza wykorzystania zasobów i ich efektywności pozwala zrozumieć, które zespoły były przeciążone lub niewłaściwie alokowane, co ułatwia planowanie przyszłych projektów.

Power BI umożliwia nie tylko retrospektywną analizę danych, ale także identyfikację trendów, wzorców i powtarzających się problemów, co jest kluczowe dla lessons learnt. Dane wizualizowane w raportach mogą być analizowane w różnych przekrojach, co daje organizacji szerszy obraz realizacji projektów wdrożeniowych.

7. Podsumowanie

Cel, który został zdefiniowany przez Zarząd firmy InfoConsulting Poland Sp. z o.o. przed rozpoczęciem prac nad niniejszym doktoratem wdrożeniowym był następujący: usprawnienie procesu wdrożeniowego systemów ERP. Określone w ten sposób wyzwanie zostało następnie podzielone na dwa kolejne zadania:

- rozwinięcie obecnej metodyki wdrożeniowej systemów ERP w taki sposób, aby świadomie zarządzić procesem realizacji różnego typu projektów,
- opracowanie nowych oraz usprawnienie obecnie wykorzystywanych narzędzi, aby poprawić efektywność pracy projektowej na każdym etapie i poziomie struktury projektu.

Wykonanie zadania w zakresie rozwoju metodyki wdrożeniowej ze szczególnym uwzględnieniem projektów informatycznych typu upgrade oraz rollout zostało potwierdzone publikacją wykazującą jak ważne jest świadome zarządzanie projektami wdrożenia ERP, ale o różnym charakterze i zakresie. Opracowane i wdrożone w organizacji zmiany do metodyki pozwoliły w sposób wymierny skrócić czas realizacji projektów technicznych. Efekt został osiągnięty poprzez opracowanie narzędzi metodycznych umożliwiających reorganizację części prac i docelowe wyeliminowanie fazy II i III dla tego typu projektów co zostało udowodnione w rozdziale 5.

Wykonanie zadania w zakresie opracowania i usprawnienia narzędzi informatycznych zostało zrealizowane w ramach rozprawy poprzez zaprojektowanie, opracowanie i dalsze produkcyjne wdrożenie rozwiązań:

- reengineering procesu i narzędzi planistycznych w zakresie prac programistycznych realizowanych w ramach projektów wdrożeniowych,
- projektowanie, wsparcie realizacji, testy i wdrożenie wyszukiwarki dokumentów specyfikacji modyfikacji „InfoSearch” z wykorzystaniem algorytmów klasteryzacji i wyznaczania podobieństwa tekstów,
- usprawnienie procesu projektowania i programowania CRIM (Customizacje, Raporty, Integracje, Modyfikacje) w projektach wdrożeniowych,

- mechanizm wsparcia procesu decyzyjnego w zakresie opracowywania VAP (Value Added Product),
- opracowanie szablonów projektowych w zakresie m. in.: struktury projektowej, scenariusza startu produkcyjnego, dokumentów.

Efekty uzyskane w ramach realizacji prac dotyczących wyszukiwarki „InfoSearch” zostały podsumowane w publikacji, której doktorant jest współautorem. Rozprawa doktorska podejmuje tematykę związaną z szeroko pojętą metodyką projektową systemów informatycznych. Na wstępie, w ramach drugiego rozdziału zostało przedstawione przez autora nowatorskie podejście w zakresie identyfikacji różnych typów projektów wdrożeniowych, które były bazą i punktem odniesienia dla kolejnych rozdziałów. Definicja projektów wdrożeniowych jest wynikiem publikacji [dom], której doktorant jest pierwszym autorem. W rozdziale trzecim wykonany został przegląd literatury w zakresie dostępnych standardowych metodyk wdrożeniowych, które są podstawą opracowanych przez producentów metodyk. Standardy w zakresie metodyki oraz architektury, jakie zostały opracowane przez największych i najpopularniejszych dostawców, są przedstawione w rozdziale czwartym. W rozdziale piątym autor opracował adaptację metodyki IC oraz IFS w podziale na zidentyfikowane w rozdziale drugim typy wdrożeń. W rozdziale szóstym zostały przedstawione narzędzia, procesy czy mechanizmy, które podczas prac badawczych zostały zaprojektowane i w znacznej większości są w użyciu naszej organizacji. W ocenie autora największym osiągnięciem w ramach badań jest opracowanie podziału projektów wdrożeniowych oraz uzupełnienie metodyki wdrożeniowej według zaproponowanego modelu. Kolejnym istotnym osiągnięciem doktoratu wdrożeniowego jest doprowadzenie do uruchomienia wyszukiwarki dokumentów specyfikacji modyfikacji, która była realizowana w naszej firmie jako jeden ze strategicznych projektów w 2021 roku.

Na podstawie zrealizowanych przez autora prac badawczych Zespół InfoConsulting Poland Sp. z o.o. uzyskał:

- dostosowaną do upgrade oraz rollout metodykę wdrożeniową,
- opracowany zestaw szesnastu rozwiązań standardowych, które w dużym stopniu zostały zidentyfikowane poprzez wyszukiwarkę, na podstawie której opracowano specyfikacje docelowych rozwiązań,
- w okresie testowym (sześć miesięcy) skrócenie czasu wytwarzania modyfikacji z około czterech do trzech tygodni, w efekcie zespół projektowy efektywniej realizuje zadania planowane do wykonania w ramach FAZY III,
- produkcyjne wdrożenie i bieżące użytkowanie nowego modelu planistycznego prac programistycznych wraz ze zmianami w zakresie budżetowania, co doprowadziło do

zmniejszenia odchyłeń w zakresie kosztów rzeczywistych programowania, a wartością sprzedaną do klienta,

- ograniczenie kosztów wdrożeniowych dla projektów typu upgrade oraz rollout techniczny, co zostało udokumentowane w rozdziale 5 podrozdział 5.3/5.3.1 oraz 5.4. Obniżenie kosztów zostało osiągnięte poprzez reorganizację prac w zakresie budowy środowiska systemowego i przeniesieniu tych prac do Fazy I. W efekcie dla tych typów projektów autor wykluczył potrzebę realizacji dwóch Faz projektowych (I i II).

Przedstawione wyniki badań w ocenie autora, udowadniają postawioną w pracy tezę, której nadrzędnym celem jest obniżenie kosztów realizacji projektów wdrożeniowych określonego typu.

Kierunki dalszego rozwoju badań są głównie związane z rozwojem narzędzi sztucznej inteligencji. Organizacja, dla której kluczową wartością i podstawowym kapitałem jest kapitał ludzki musi stale podążać za trendami zarówno w zakresie wykorzystywanych narzędzi jak i standardów zarządzania. W związku z tym wskazane jest, aby kontynuować prace w zakresie usprawnienia modelu planistycznego, ale z większym naciskiem na zespół konsultingowy. Natomiast z perspektywy naukowej doskonałym kierunkiem jest adaptacja systemów ERP w zakresie sztucznej inteligencji. W chwili obecnej Systemy Zintegrowanego Zarządzania są na poziomie AI Specjalistycznym (Narrow AI), czyli mogą wspierać realizację pojedynczych zadań. Docelowo przejście do czwartego poziomu rozwoju, czyli Autonomicznych Systemów AI pozwoliłoby na całkowicie autonomicznie w tym fizyczne działanie, bez konieczności ludzkiego nadzoru. W kontekście systemów ERP organizacje mogłyby działać jak autonomiczne współpracujące ze sobą roboty, połączone w jeden w różnego rodzaju ekosystemy sprzężone z IoT na poziomie całej gospodarki.

Bibliografia

- [1] ML Markus i C Tanis. „*Chapter 10: The Enterprise System Experience-From Adoption to Success. Framing the Domains of IT Management: Projecting the Future Through the Past, Pinnaflex Educational Resources*”. 2000.
- [2] Meg Murray i George Coffin. „*A case study analysis of factors for success in ERP system implementations*”. W: *AMCIS 2001 Proceedings* (2001), s. 196.
- [3] Helmut Klaus, Michael Rosemann i Guy G Gable. „*What is ERP?*” W: *Information systems frontiers 2.2* (2000), s. 141–162.
- [4] THOMAS F Gattiker, DANIEL Chen i DALE L Goodhue. „*Agility through standardization: A CRM/ERP application*”. W: *Bendoly, E. and FR Jacobs (2005). Strategic ERP Extension and Use, Stanford University Press, Stanford* (2005), s. 87–96.
- [5] Andrejs Tambovcevs. „*ERP system implementation: A case study of the construction enterprise*”. W: *Economics & Management 15* (2010), s. 1092–1098.
- [6] Mahadevan Supramaniam i Mudiwaran Kuppusamy. „*ERP System Adoption in Malaysia: A Comparative Analysis between SMEs and MNCs*”. W: *The European Conference on Information Systems Management. Academic Conferences International Limited. 2010*, s. 372.
- [7] Edyta Kucharska i in. „*ERP Systems in Energy Industry-opportunities and challenges.*” W: *Przegląd Elektrotechniczny 98.11* (2022).
- [8] Shruti Nagpal, Sunil Kumar Khatri i Ashok Kumar. „*Comparative study of ERP implementation strategies*”. W: *2015 Long Island Systems, Applications and Technology. IEEE. 2015*, s. 1–9.
- [9] N Bancroft, H Seip i A Sprengel. „*Implementing SAP R/3. How to Introduce a Large System into a Large Organization, 2nd Edition. Greenwich, Conn*”. 1998.
- [10] Palitha R Kuruppuarachchi. „*Organisational factors and IT projects-a critical review*”. W: *Proceedings of the 2000 IEEE International Conference on Management of Innovation and Technology. ICMIT 2000.'Management in the 21st Century'*(Cat. No. 00EX457). T. 2. IEEE. 2000, s. 496–501.

- [11] M Mäkipää. „*Implementation of Enterprise Resource Planning System—theoretical research approach and empirical evaluation in two cases*”. W: *Proceedings of the 26th information systems research seminar in Scandinavia*. 2003.
- [12] Anne N Parr i Graeme Shanks. „*A taxonomy of ERP implementation approaches*”. W: *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. IEEE. 2000, 10–pp.
- [13] Jeanne W Ross i Michael R Vitale. „*The ERP revolution: surviving vs. thriving*”. W: *Information systems frontiers 2.2* (2000), s. 233–241.
- [14] Sunil Kaushik i Akashdeep Bhardwaj. „*Compressing Effort and Schedule in ERP Implementations with Iterative Methodologies: Few Industrial Cases*”. W: ().
- [15] Mary M Dunaway. „*ERP implementation methodologies and strategies*”. W: *V Readings on Enterprise Resource Planning* (2012), s. 46–58.
- [16] Victoria Stanciu i Andrei Tinca. „*ERP solutions between success and failure*”. W: *Accounting and Management Information Systems 12.4* (2013), s. 626–649.
- [17] C Shaw. „*ERP system upgrade: An analysis of the costs, benefits and requirements*”. Cranfield University (United Kingdom), 2002.
- [18] Tanja Scheckenbach i in. „*Issues of ERP upgrade in public sectors: A case study*”. W: *Human-Computer Interaction. Applications and Services: 16th International Conference, HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part III 16*. Springer. 2014, s. 754–763.
- [19] Ike C Ehie i Mogens Madsen. „*Identifying critical issues in enterprise resource planning (ERP) implementation*”. W: *Computers in industry 56.6* (2005), s. 545–557.
- [20] David L Olson i Fan Zhao. „*Critical success factors in ERP upgrade projects*”. W: *Research and Practical Issues of Enterprise Information Systems: IFIP TC 8 International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2006) April 24–26, 2006, Vienna, Austria*. Springer. 2006, s. 569–578.
- [21] Jorge Gomes i Mário Romão. „*Improving project success: A case study using benefits and project management*”. W: *Procedia Computer Science 100* (2016), s. 489–497.
- [22] Moutaz Haddara i Ahmed Elragal. „*The Readiness of ERP Systems for the Factory of the Future*”. W: *Procedia Computer Science 64* (2015). Conference on ENTERprise Information Systems/International Conference on Project MANagement/Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2015 October 7-9, 2015, s. 721–728. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2015.08.598>.

- [23] Fergal Carton i Frederic Adam. „*Analysing the impact of enterprise resource planning systems roll-outs in multi-national companies*”. W: Academic Conferences i Publishing International (ACPI). 2003.
- [24] Christian Barth i Stefan Koch. „*Critical success factors in ERP upgrade projects*”. W: *Industrial Management & Data Systems* 119.3 (2019), s. 656–675.
- [25] Per Svejvig i Tina Blegind Jensen. „*Making sense of enterprise systems in institutions: a case study of the re-implementation of an accounting system*”. W: *Scandinavian Journal of Information Systems* 25.1 (2013), s. 1.
- [26] Christian Leyh i Peter Muschick. „*Critical Success Factors for ERP system upgrades–The Case of a German large-scale Enterprise*”. W: (2013).
- [27] Jaehoon Whang, Moon-Bong Lee i Kijoo Kim. „*A case study on the successful upgrade of ERP system*”. W: *PACIS 2003 Proceedings* (2003), s. 71.
- [28] Martin Riedel. „*Managing SAP ERP 6.0 upgrade projects*”. Galileo Press Bonn, 2009.
- [29] Sonny Demi i Moutaz Haddara. „*Do cloud ERP systems retire? An ERP lifecycle perspective*”. W: *Procedia computer science* 138 (2018), s. 587–594.
- [30] Bartosz Wachnik. „*IT re-implementation project. A multiple case study analysis*”. W: *Informatyka Ekonomiczna* 1 (43 (2017), s. 86–97.
- [31] Hayder K Aljamee i Saja M Naeem. „*The benefits of applying project management methodology on project delay: A study in construction projects in Iraq*”. W: *IOP Conference Series: Materials Science and Engineering*. T. 745. 1. IOP Publishing. 2020, s. 012155.
- [32] Maja Gaborov i in. „*Comparative analysis of agile and traditional methodologies in IT project management*”. W: *Journal of Applied Technical and Educational Sciences* 11.4 (2021), 1–ArtNo.
- [33] Philipp Rosenberger i József Tick. „*Multivariate optimization of PMBOK, version 6 project process relevance*”. W: *Acta Polytechnica Hungarica* 18.11 (2021), s. 9–28.
- [34] Kunal Mohan i Frederik Ahlemann. „*Committed use of project management methodologies: Understanding the role of costs, benefits, and psychological needs*”. W: (2013).
- [35] Udesh S Senarath. „*Waterfall methodology, prototyping and agile development*”. W: *Tech. Rep.* (2021), s. 1–16.
- [36] Theo Thesing, Carsten Feldmann i Martin Burchardt. „*Agile versus waterfall project management: decision model for selecting the appropriate approach to a project*”. W: *Procedia Computer Science* 181 (2021), s. 746–756.

- [37] Harkirat Kaur Aroral. „Waterfall process operations in the fast-paced world: project management exploratory analysis”. W: *International Journal of Applied Business and Management Studies* 6.1 (2021), s. 91–99.
- [38] Bogdan-Alexandru Andrei i in. „A study on using waterfall and agile methods in software project management”. W: *Journal of Information Systems & Operations Management* (2019), s. 125–135.
- [39] Joao Alexandre Lobo Marques i in. „Effectiveness analysis of waterfall and agile project management methodologies—a case study from Macau’s construction industry”. W: *Revista Gestão em Análise* 12.1 (2023), s. 23–38.
- [40] Nevenka Kirovska i Saso Koceski. „Usage of Kanban methodology at software development teams”. W: *Journal of applied economics and business* 3.3 (2015), s. 25–34.
- [41] Michal Hron i Nikolaus Obwegeser. „Scrum in practice: an overview of Scrum adaptations”. W: *Hawaii International Conference on System Sciences*. Curran Associates, Inc. 2018, s. 4496–4505.
- [42] Pedro Serrador i Jeffrey K Pinto. „Does Agile work?—A quantitative analysis of agile project success”. W: *International journal of project management* 33.5 (2015), s. 1040–1051.
- [43] Godfred Yaw Koi-Akrofi, Joyce Koi-Akrofi i Henry Akwetey Matey. „Understanding the characteristics, benefits and challenges of agile it project management: A literature based perspective”. W: *International Journal of Software Engineering & Applications (IJSEA)* 10.5 (2019), s. 25–44.
- [44] Maja Due Kadenic, Konstantinos Koumaditis i Louis Junker-Jensen. „Mastering scrum with a focus on team maturity and key components of scrum”. W: *Information and Software Technology* 153 (2023), s. 107079.
- [45] Rashina Hoda, Norsaremah Salleh i John Grundy. „The rise and evolution of agile software development”. W: *IEEE software* 35.5 (2018), s. 58–63.
- [46] Samar Al-Saqqa, Samer Sawalha i Hiba AbdelNabi. „Agile software development: Methodologies and trends.” W: *International Journal of Interactive Mobile Technologies* 14.11 (2020).
- [47] Hamzah Alaidaros, Mazni Omar i Rohaida Romli. „The state of the art of agile kanban method: challenges and opportunities”. W: *Independent Journal of Management & Production* 12.8 (2021), s. 2535–2550.
- [48] Darrell K Rigby, Jeff Sutherland i Andy Noble. „Agile at scale”. W: *Harvard business review* 96.3 (2018), s. 88–96.

- [49] Daniel Brosseau i in. „*The journey to an agile organization*”. W: *McKinsey & Company, May 10* (2019), s. 14–27.
- [50] Sakshi Sachdeva. „*Scrum methodology*”. W: *Int. J. Eng. Comput. Sci* 5.16792 (2016), s. 16792–16800.
- [51] Putu Adi Guna Permana. „*Scrum method implementation in a software development project management*”. W: *International Journal of Advanced Computer Science and Applications* 6.9 (2015), s. 198–204.
- [52] Howard Lei i in. „*A statistical analysis of the effects of Scrum and Kanban on software development projects*”. W: *Robotics and Computer-Integrated Manufacturing* 43 (2017), s. 59–67.
- [53] Philipp Diebold i in. „*What do practitioners vary in using scrum?*” W: *Agile Processes in Software Engineering and Extreme Programming: 16th International Conference, XP 2015, Helsinki, Finland, May 25-29, 2015, Proceedings 16*. Springer. 2015, s. 40–51.
- [54] Eduard Pangestu Wonohardjo, Rizky Febriyanto Sunaryo i Yusuf Sudiyono. „*A systematic review of SCRUM in software development*”. W: *JOIV: International Journal on Informatics Visualization* 3.2 (2019), s. 108–112.
- [55] Martin Tomanek i Jan Juricek. „*Project risk management model based on PRINCE2 and SCRUM frameworks*”. W: *arXiv preprint arXiv:1502.03595* (2015).
- [56] Osama Al-Baik i James Miller. „*The kanban approach, between agility and leanness: a systematic review*”. W: *Empirical Software Engineering* 20 (2015), s. 1861–1897.
- [57] Rajat B Wakode, Laukik P Raut i Pravin Talmale. „*Overview on kanban methodology and its implementation*”. W: *IJSRD-International Journal for Scientific Research & Development* 3.02 (2015), s. 2321–0613.
- [58] Mariam Houti, Laila El Abbadi i Abdellah Abouabdellah. „*E-Kanban the new generation of traditional Kanban system, and the impact of its implementation in the enterprise*”. W: *Proceedings of the International Conference on Industrial Engineering and Operations Management Rabat*. 2017, s. 11–13.
- [59] Muhammad Ovais Ahmad i in. „*Kanban in software engineering: A systematic mapping study*”. W: *Journal of Systems and Software* 137 (2018), s. 96–113.
- [60] Gholamreza Jamali i Mina Oveisi. „*A study on project management based on PMBOK and PRINCE2*”. W: *Modern Applied Science* 10.6 (2016), s. 142–146.
- [61] Klas Skogmar. „*PRINCE2, the PMBOK guide and ISO 21500: 2012*”. W: *London: Axelos* (2015).

- [62] Radka Vaničková. „Application of PRINCE2 project management methodology”. W: *Studia Commercialia Bratislavensia* 10.38 (2017), s. 227.
- [63] Saiful Islam i Nina Evans. „Key success factors of PRINCE2 project management method in software development project: KSF of PRINCE2 in SDLC”. W: *International Journal of Engineering Materials and Manufacture* 5.3 (2020), s. 76–84.
- [64] Amir Faraji i in. „Applicability-compatibility analysis of PMBOK seventh edition from the perspective of the construction industry distinctive peculiarities”. W: *Buildings* 12.2 (2022), s. 210.
- [65] João Victor Rojas Luiz, Fernando Bernardi de Souza i Octaviano Rojas Luiz. „PMBOK® and critical chain practices: Antagonisms and opportunities for complementation”. W: *Gestão & Produção* 24 (2017), s. 464–476.
- [66] Filipe Amaro i Luisa Domingues. „PMBOK 6th meets 7th: How to link both guides in order to support project tailoring?” W: *Procedia Computer Science* 219 (2023), s. 1877–1884.
- [67] Marcelle Cury Rodrigues, Luisa Domingues i João Pedro Oliveira. „Tailoring: a case study on the application of the seventh principle of PMBOK 7 in a public institution.” W: *Procedia Computer Science* 219 (2023), s. 1735–1743.
- [68] P Rosenberger i J Tick. „Suitability of PMBOK 6th edition for agile-developed IT Projects, 2018”. W: *CINTI 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics, Budapest Hungary*.
- [69] Nilton Takagi i João Varajão. „Success management and the project management body of knowledge (PMBOK): An integrated perspective”. W: (2020).
- [70] Yad Senapathy. „PMBOK 7th Edition Guide 2024: Release Date, Latest Version Changes, PMI Knowledge Areas for Project Management (PMP)”. <https://www.4pmti.com/learn/pmbok-guide-7th-ed/>. Dostęp: 2024-07-07. 2024.
- [71] Eoghan Toomey. „The project management challenges, benefits, risks and limitations of adopting agile methodologies for a multiphase ERP program”. W: (2021).
- [72] Gartner. „Gartner 2024 Report”. <https://www.gartner.com/reviews/market/cloud-erp-for-product-centric-enterprises>. Dostęp: 2024-07-07. 2024.
- [73] SAP. „SAP Dokumentacja techniczna”. <https://help.sap.com/docs/>. Dostęp: 2024-07-07. 2024.
- [74] SAP. „What is SAP HANA”. <https://www.sap.com/products/technology-platform/hana/what-is-sap-hana.html#data-virtualization> note =(Dostęp: 2024-07-07). 2024.

- [75] SAP. „SAP Web Dispatcher”. https://help.sap.com/docs/ABAP_PLATFORM/683d6a1797a34730a6e005d1e8de6f22/488fe37933114e6fe1000000a421937.html. Accessed: 2024-10-02. 2024.
- [76] Aroon Raj. „What is SAP S/4HANA?” <https://erpiseasy.com/2021/03/20/what-is-sap-s4hana/comment-page-1/>. Dostęp: 2024-07-07. 2024.
- [77] SAP. „SAP Activate Methodology”. <https://www.sap.com/documents/2021/10/9e775207-fe7d-0010-bca6-c68f7e60039b.html>. Dostęp: 2024-07-07. 2024.
- [78] Leanix. „SAP Activate Methodology”. <https://www.leanix.net>. Dostęp: 2024-07-07. 2024.
- [79] SAP. „SAP Activate RoadMaps”. <https://go.support.sap.com/roadmapviewer/>. Dostęp: 2024-07-07. 2024.
- [80] SAP. „SAP Activate Methodology - Community”. <https://community.sap.com/t5/technology-blogs-by-members/the-beginner-s-guide-to-sap-activate-best-practices-guided-configuration/ba-p/13460868>. Dostęp: 2024-07-07. 2024.
- [81] SAP. „The Secure Software Development Lifecycle at SAP”. <https://www.sap.com/documents/2016/03/a248a699-627c-0010-82c7-eda71af511fa.html>. Dostęp: 2024-07-07. 2024.
- [82] ALL4ONE. „Metodyka wdrożeniowa all for one dla projektu upgrade”. <https://www.all-for-one.pl/pl/oferta/rise-with-sap/>. Dostęp: 2024-08-30. 2024.
- [83] ALL4ONE. „Metodyka wdrożeniowa all for one dla projektu wdrożenia od podstaw”. <https://www.all-for-one.pl/pl/oferta/grow-with-sap/>. Dostęp: 2024-08-30. 2024.
- [84] Microsoft Dynamics. „Microsoft Dynamics Dokumentacja techniczna”. <https://help.sap.com/docs/>. Dostęp: 2024-07-07. 2024.
- [85] IFS. „IFS Dokumentacja techniczna”. https://docs.ifs.com/techdocs/24r1/010_overview/. Dostęp: 2024-07-07. 2024.
- [86] Adam Domagała i in. „Post-Implementation ERP Software Development: Upgrade or Reimplementation”. W: Applied Sciences 11.11 (2021). ISSN: 2076-3417.

- [87] Sandro Luís Freire de Castro Silva i Saulo Barbará de Oliveira. „*Planning and Scope Definition to Implement ERP: The Case Study of Federal Rural University of Rio de Janeiro (UFRRJ)*”. W: *Procedia Computer Science* 64 (2015). Conference on ENTERprise Information Systems/International Conference on Project MANagement/Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2015 October 7-9, 2015, s. 196–203. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2015.08.481>.
- [88] Katarzyna Grobler-Debska i in. „*Automatic classification of specification of ERP modifications documents*”. W: (2022), s. 01–06. DOI: [10.1109/MMAR55195.2022.9874269](https://doi.org/10.1109/MMAR55195.2022.9874269).