



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



Politechnika Świętokrzyska
Wydział Zarządzania i Modelowania Komputerowego

Kierunek studiów:
Inżynieria danych

Sławomir Luściński

Materiały dydaktyczne do przedmiotu

Modelowanie procesów biznesowych

opracowane w ramach realizacji Projektu
**„Dostosowanie kształcenia
w Politechnice Świętokrzyskiej do potrzeb
współczesnej gospodarki”**
FERS.01.05-IP.08-0234/23

Kielce, 2025





Spis treści

Wprowadzenie	4
1. Zarządzanie procesami biznesowymi BPM	5
1.1. Cykl zarządzania procesami biznesowymi	5
1.2. Rozwój BPM	9
1.3. Role i odpowiedzialności w BPM	11
2. Automatyzacja procesów biznesowych	16
2.2. Pojęcie automatyzacji procesów biznesowych	16
2.1. Architektura systemów automatyzacji procesów	17
2.2. Silnik workflow	18
2.3. Robotyzacja procesów a zarządzanie procesami biznesowymi	23
2.4. Korzyści z automatyzacji procesów	26
3. Inteligentne zarządzanie procesami biznesowymi iBPM	29
3.1. Definicja iBPM	29
3.2. Uczenie maszynowe w analizie procesów	33
3.3. Automatyczne podejmowanie decyzji	35
3.4. Przyszłość iBPM	39
4. Platforma Bonitasoft Community Edition	41
4.1. Charakterystyka	41
4.2. Projektowanie procesów	42
4.3. Realizacja procesów	43
5. Cykl budowy aplikacji procesowej	45
5.1. Przygotowanie projektu	45
5.2. Tworzenie diagramu procesu	48
5.3. Definiowanie ról i uprawnień	50
5.4. Testowanie aplikacji procesowej	51
5.5. Publikacja i uruchomienie procesu	54
Literatura	55
Spis rysunków	57
Spis tabel	57



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



Materiały dydaktyczne objęte licencją Creative Commons BY 4.0.

Licencja dostępna pod adresem: <https://creativecommons.org/licenses/by/4.0/>





Wprowadzenie

Celem niniejszego opracowania jest kompleksowe omówienie zarządzania procesami biznesowymi (BPM) ze szczególnym uwzględnieniem wykorzystania platformy open-source Bonitasoft Community Edition, jako platformy do budowy aplikacji procesowych. Założono, że czytelnik posiada znajomość zagadnień związanych z modelowaniem procesów z zastosowaniem notacji BPMN 2.0.

Analiza zagadnień związanych z zarządzaniem procesami biznesowymi oraz ich automatyzacją w środowisku Bonitasoft ukazuje szerokie spektrum możliwości, jakie oferuje ta platforma w kontekście projektowania, wdrażania i optymalizacji procesów zgodnych z notacją BPMN 2.0. W szczególności na uwagę zasługuje rozwój inteligentnego zarządzania procesami biznesowymi (iBPM), w tym zastosowanie sztucznej inteligencji i uczenia maszynowego do analizy procesów oraz automatycznego podejmowania decyzji.

Platforma Bonitasoft, dzięki modułowi Bonita Studio, silnikowi workflow oraz portalowi do monitorowania, pozwala na kompleksowe zarządzanie cyklem życia procesów – od ich koncepcji, przez implementację, aż po bieżące nadzorowanie i wprowadzanie usprawnień. Szczególną wartość stanowi możliwość łączenia działań manualnych z automatycznymi, integracji z różnorodnymi systemami zewnętrznymi oraz elastyczne przypisywanie ról i uprawnień, co sprzyja zachowaniu przejrzystości realizacji zadań.

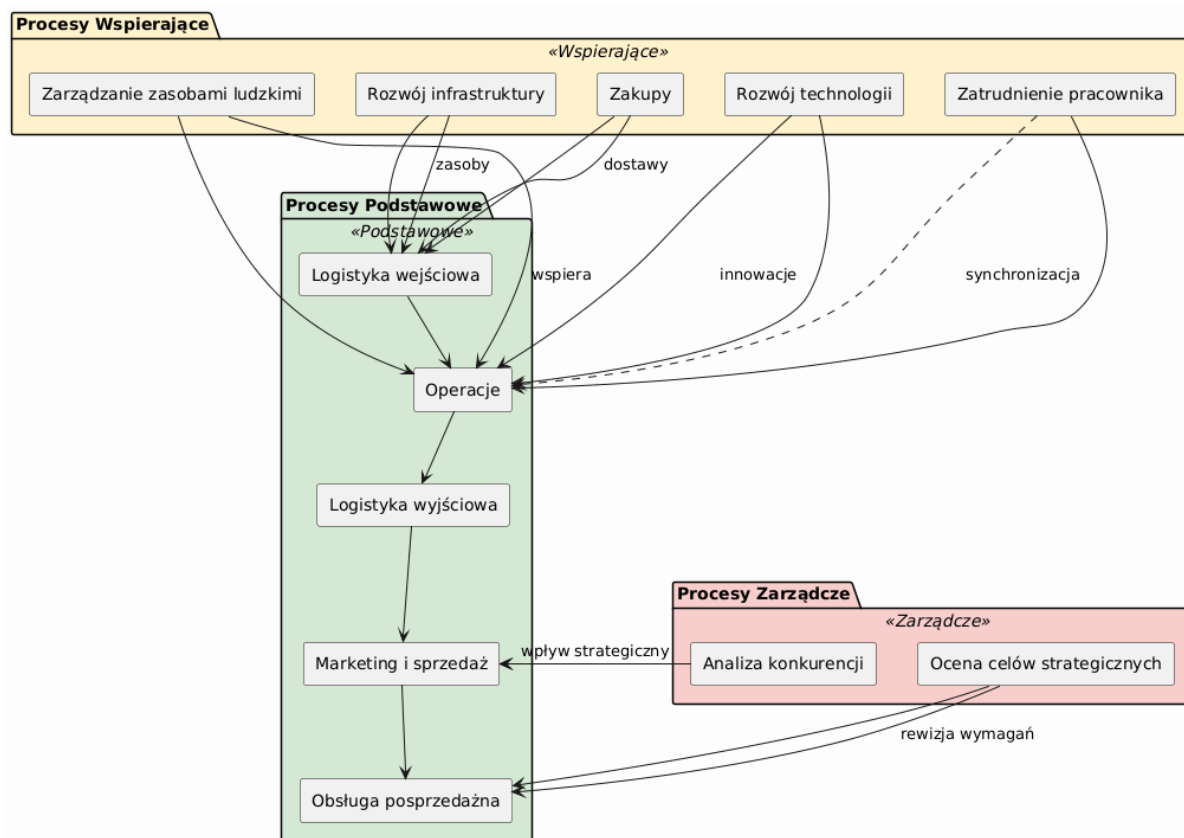
Opracowanie integruje aspekty teoretyczne z praktycznymi narzędziami, umożliwiającymi tworzenie aplikacji procesowych, które łączą modelowanie logiczne z rzeczywistym wykonaniem w systemie informatycznym.

1. Zarządzanie procesami biznesowymi BPM

1.1. Cykl zarządzania procesami biznesowymi

Zarządzanie procesami biznesowymi (*Business Process Management*, BPM) to systematyczne podejście do identyfikowania, analizowania, wdrażania, monitorowania i doskonalenia działań w organizacji, mające na celu osiągnięcie określonych wyników. Opiera się na łączeniu perspektywy strategicznej i operacyjnej – procesy stanowią most między celami firmy a codziennymi zadaniami pracowników. Na Rys. 1 przedstawiono zależności między trzema grupami procesów w organizacji.

[Tekst alternatywny: Diagram przedstawia zależności między trzema grupami procesów w organizacji: procesami wspierającymi (blok żółty), podstawowymi (blok zielony) i zarządczymi (blok różowy). Strzałki obrazują powiązania pomiędzy poszczególnymi elementami grup.]



Rys. 1. Kluczowe procesy w organizacji.

Źródło: opracowanie własne.



Procesy w organizacji możemy pogrupować ze względu na ich wpływ na działalność podstawową organizacji na:

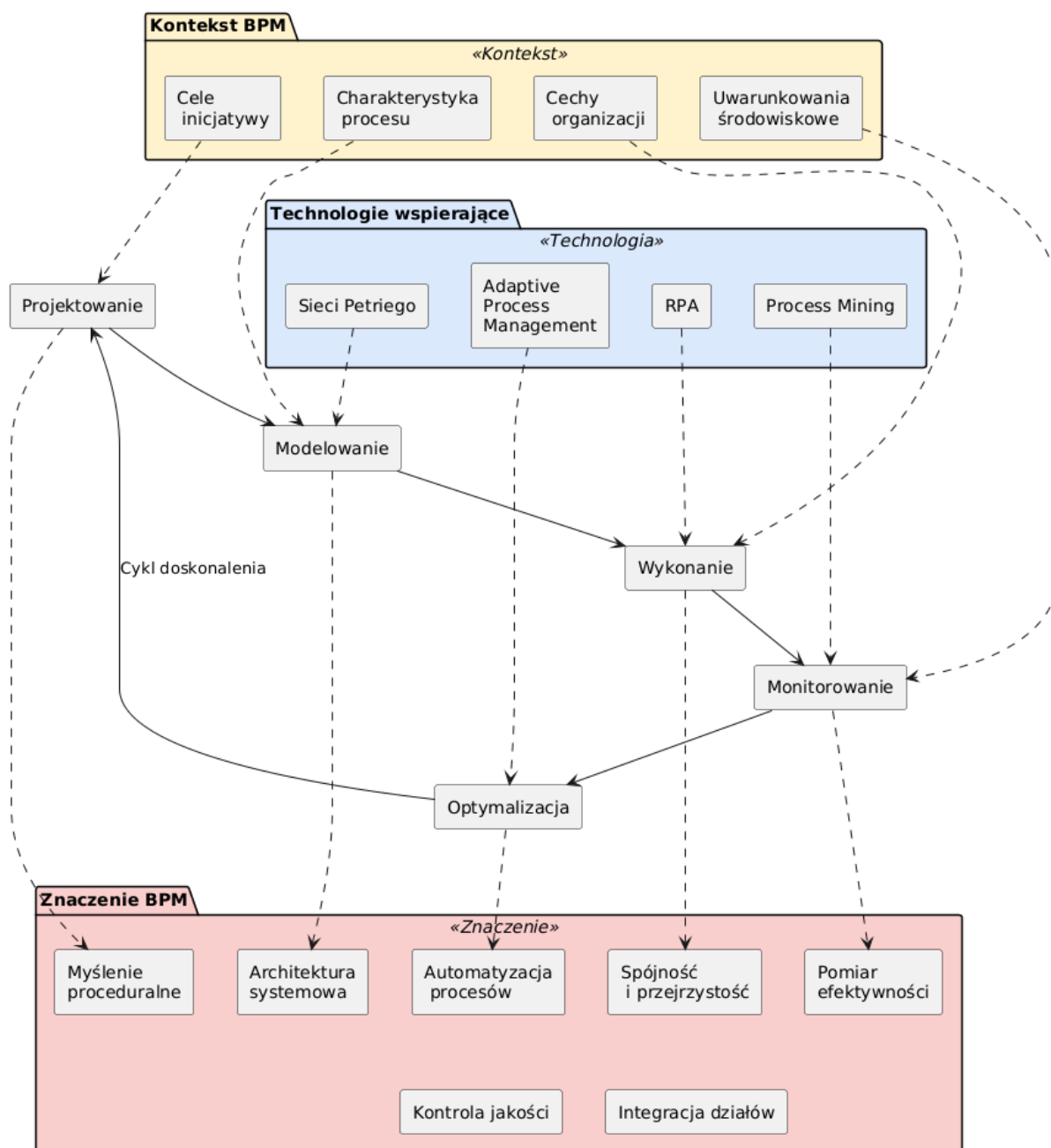
1. **Procesy wspierające** - obejmują zarządzanie zasobami ludzkimi, rozwój infrastruktury, zakupy, rozwój technologii i zatrudnienie pracownika. Dostarczają zasobów, innowacji i wsparcia dla procesów podstawowych.
2. **Procesy podstawowe** - tworzą główny łańcuch wartości organizacji poczynając od logistyki wejściowej przez kolejno: procesy operacyjne, logistykę wyjściową, marketing i sprzedaż po obsługę posprzedażną. Procesy podstawowe są ściśle związane z kluczową wartością, jaką organizacja oferuje klientom. Obejmują działania, które bezpośrednio prowadzą do powstania i dostarczenia produktów lub usług, za które odbiorcy są gotowi zapłacić.
3. **Procesy zarządcze** mają charakter nadrzędny – wpływają strategicznie na pozostałe procesy, a także dokonują rewizji wymagań procesów podstawowych na podstawie analizy konkurencji i oceny celów strategicznych.

BPM można rozumieć na dwa sposoby: jako metodykę proceduralnego myślenia na poziomie organizacji, która tworzy wspólny język dla menedżerów i specjalistów, oraz jako narzędzie do pomiaru i nadzorowania struktur. Holistyczne podejście obejmuje nie tylko aktualny stan procesu, lecz także prognozy przyszłych zmian. Takie długoterminowe spojrzenie pozwala na zachowanie zgodności między strategią a codziennym funkcjonowaniem operacyjnym.

BPM to koncepcja obejmująca metody, techniki i praktyki organizacyjne oraz technologie IT. Proces zarządzania procesami przebiega w cyklu obejmującym **projektowanie, modelowanie, realizację, monitorowanie i optymalizację** (

Rys. 2).

[Tekst alternatywny: Diagram przedstawia systemowy model zarządzania procesami biznesowymi (BPM), łączący trzy obszary: kontekst BPM, technologie wspierające i znaczenie BPM. W górnej części pokazano kontekst inicjatywy - cele, charakterystykę procesu, cechy organizacji i uwarunkowania środowiskowe. Środkowa część prezentuje technologie wspierające, takie jak sieci Petriego, *Adaptive Process Management*, RPA i *Process Mining*, które wspomagają kolejne etapy cyklu BPM: projektowanie, modelowanie, wykonanie, monitorowanie i optymalizację. Dolna część obrazuje znaczenie BPM – obejmuje myślenie proceduralne, architekturę systemową, automatyzację procesów, spójność i przejrzystość działań, pomiar efektywności, kontrolę jakości oraz integrację działów.]



Rys. 2. Cykl zarządzania procesami biznesowymi.
Źródło: opracowanie własne.



Podejście cykliczne oznacza, że po każdej turze wdrożenia i analizy wyników następuje powrót do wcześniejszych etapów w celu ich korekty i dalszego udoskonalania. Obecnie coraz częściej można zaobserwować łączenie BPM z automatycznymi rozwiązaniami, takimi jak *proces mining* czy *adaptive process management*, co odzwierciedla postęp technologiczny i zmiany w sposobie pracy.

Kluczowym elementem BPM jest koncentracja na pomiarze efektywności i skuteczności działań. Pomiar ten może przybierać formę ilościową lub jakościową - od prostych metod statystycznej kontroli procesu po oceny uwzględniające koszty lub analizę przepływów pracy. Wdrożenie takich rozwiązań wymaga opracowania systemu monitorowania, który odzwierciedla rzeczywiste kryteria oceny procesu. Wyznaczenie odpowiednich wskaźników jest kluczowe dla kontroli procesu: dane są zbierane na bieżąco lub okresowo i analizowane pod kątem ich zgodności z celami. Obszar ten ma również silny komponent kontekstowy - przed rozpoczęciem projektu automatyzacji procesu konieczna jest analiza warunków specyficznych dla danego przedsięwzięcia procesowego.

Koncepcja tak zwanej ramy kontekstowej BPM wyróżnia cztery główne wymiary:

1. Cele inicjatywy (czy dążymy do optymalizacji istniejących działań, czy szukamy nowych rozwiązań).
2. Charakterystykę samego procesu (np. stopień złożoności czy kreatywności).
3. Cechy organizacji (branża, wielkość, kultura).
4. Uwarunkowania środowiskowe (np. poziom konkurencyjności lub niepewności otoczenia).

Uwzględnienie tych zmiennych umożliwia lepsze dopasowanie metodyki BPM do specyfiki danego kontekstu przedsiębiorstwa.

Znaczenie BPM można również widzieć w jego roli jako narzędzia łączącego różne działy w organizacji. Badania wskazują, że wartość procesów różni się w zależności od obszaru biznesowego – na przykład w finansach właściciele procesów występują znacznie częściej niż w działach R&D czy magazynowych. Oznacza to zróżnicowany poziom odpowiedzialności za jakość procesu, zależny od natury procesu i lokalnej kultury zarządzania.

Kluczowe technologie wspierające BPM to:

1. **Sieci Petriego (Petri Nets)**. Formalny model matematyczny używany do opisu, analizy i symulacji procesów współbieżnych, czyli takich, w których wiele działań może się odbywać równocześnie z synchronizacją i zależnościami między nimi. Są szeroko wykorzystywane w BPM, automatyzacji produkcji, informatyce



i systemach sterowania, ponieważ umożliwiają dokładne modelowanie przepływu pracy, zasobów oraz warunków logiki między zdarzeniami.

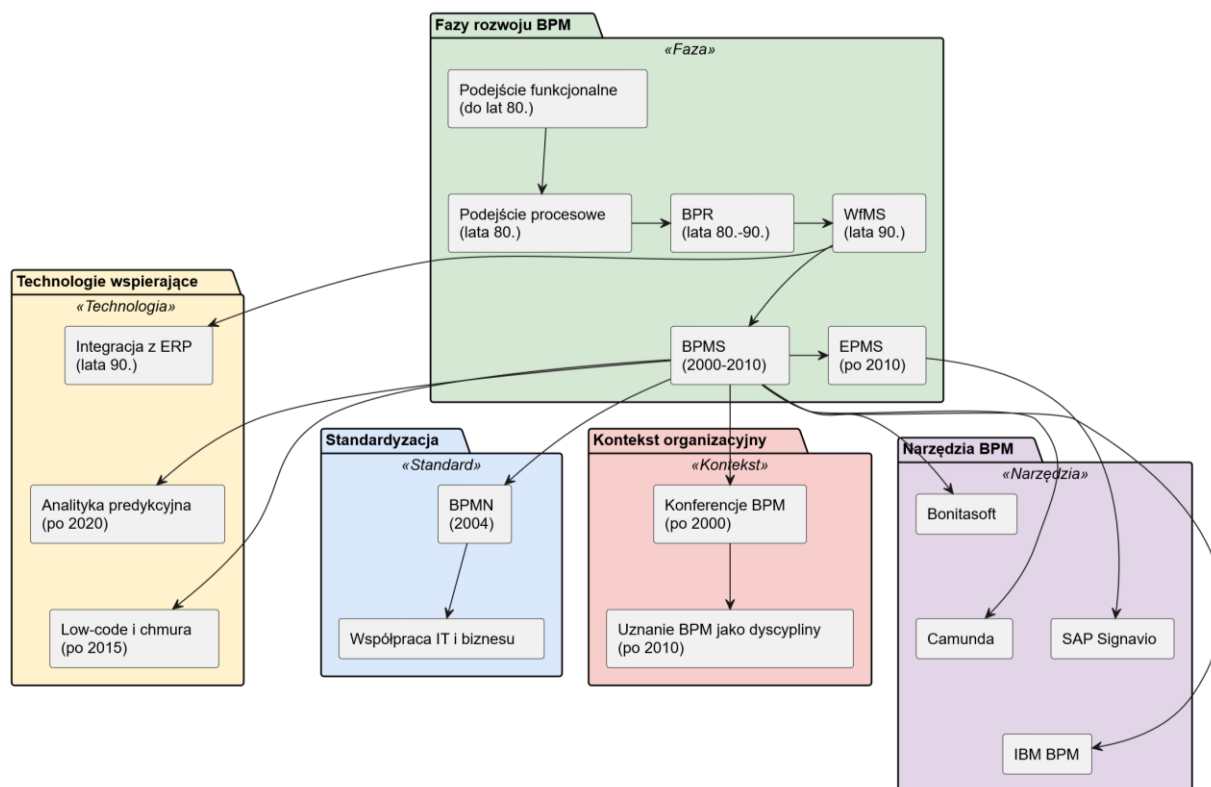
2. **Adaptacyjne zarządzanie procesami** APM (*Adaptive Process Management*). Podejście do zarządzania procesami biznesowymi, które pozwala na elastyczne dostosowywanie procesu w trakcie jego realizacji - bez konieczności pełnego przeprojektowania lub ponownego wdrożenia modelu. W odróżnieniu od tradycyjnego BPM, opierającego się na sztywnych, wstępnie zdefiniowanych przepływach, APM jest zorientowane na adaptację, uczenie się i reagowanie na zmiany kontekstu biznesowego w czasie rzeczywistym.
3. **Robotyzacja procesu automatyzacji** za pomocą RPA (*Robotic Process Automation*). Technologia, która służy do automatyzacji powtarzalnych zadań wykonywanych przez ludzi w systemach informatycznych. Zamiast fizycznego robota, w RPA działa „robot programowy” (*software robot*), który naśladuje działania użytkownika w interfejsie aplikacji (np. klikanie, kopiowanie danych, wypełnianie formularzy, wysyłanie e-maili czy generowanie raportów). RPA często stanowi etap wstępny w rozwoju bardziej zaawansowanych systemów automatyzacji, takich jak wykorzystanie sztucznej inteligencji (AI) lub systemów *workflow*, na przykład BonitaSoft.
4. Eksploracja procesów PM (*Process Mining*) to technologia analizy procesów biznesowych, która odkrywa, wizualizuje i optymalizuje rzeczywiste procesy na podstawie danych z systemów informatycznych (np. ERP, CRM, BPM). Logi zdarzeń w systemach zawierają informacje, kto, kiedy i co wykonał w procesie. W eksploracji procesów te dane służą do odkrywania procesów w logach (automatyczne generowanie modeli na podstawie logów), sprawdzania zgodności (porównanie z modelem referencyjnym) oraz do udoskonalania procesów, poprzez identyfikację opóźnień, duplikatów i błędów.

Koncepcja BPM obejmuje element systemowy, w którym wszystkie procesy są osadzone w szerszej architekturze przedsiębiorstwa. Ta architektura łączy jednostki organizacyjne i wspierające je technologie informacyjne. Dzięki temu można tworzyć przejrzyste mapy interakcji między procesami głównymi a pomocniczymi oraz identyfikować punkty styku kluczowe dla działalności organizacji.

1.2. Rozwój BPM

Historia zarządzania procesami biznesowymi wywodzi się z wcześniejszych koncepcji organizacji pracy, w których początkowo dominowało podejście funkcjonalne (Rys. 3).

[Tekst alternatywny: Tekst diagram ilustruje historyczny rozwój zarządzania procesami biznesowymi (BPM), ukazując, jak na przestrzeni lat zmieniały się fazy, technologie, standardy, kontekst organizacyjny i narzędzia informatyczne. Całość podzielona jest na pięć bloków tematycznych, połączonych strzałkami symbolizującymi ich wzajemne zależności. Pokazuje ewolucję BPM — od podejścia funkcjonalnego w latach 80. do dzisiejszych systemów EPMS. Wskazuje zależności między etapami rozwoju BPM, takimi jak technologie (ERP, low-code, chmura, analityka predykcyjna), standardy (BPMN), kontekst organizacyjny (konferencje, uznanie BPM) oraz narzędziami, takimi jak Bonitasoft, Camunda, SAP Signavio i IBM BPM.



Rys. 3. Rozwój zarządzania procesami biznesowymi.

Źródło: opracowanie własne.

Pierwotne struktury organizacyjne opierały się na działach i specjalizacjach, skupiając się na optymalizacji poszczególnych zadań w ramach wydzielonych jednostek. Z czasem pojawiła się koncepcja patrzenia na działalność przez pryzmat procesów - kolejnych działań zmierzających do osiągnięcia określonego celu w najefektywniejszy sposób. To był ważny krok, ponieważ przesunął nacisk z lokalnej optymalizacji na poziomie jednostek na optymalizację całego strumienia wartości.



W latach 90. XX wieku rozwój technologii informatycznych i popularyzacja systemów zarządzania przepływem pracy (WfMS) wpłynęły na automatyzację przepływu pracy w zespołach. Początkowo WfMS realizowały zadania według ustalonych reguł, a potem zostały wzbogacone o funkcje monitorowania i analizy procesów. Integracja z systemami ERP pozwoliła na powiązanie procesów z danymi operacyjnymi. Rozwój tych narzędzi doprowadził do powstania BPMS, które obejmują cały cykl życia procesu- od projektowania po analizę wyników. Historia BPM to kontynuacja koncepcji reinżynierii procesów (BPR) z lat 80. i 90., choć z przesunięciem nacisku z radykalnej przebudowy na ciągłe doskonalenie, realizowane iteracyjnie i wspierane narzędziami informatycznymi, co zwiększa elastyczność zmian. Równocześnie narzędzia BPM ewoluowały od prostych edytorów do kompleksowych BPMS dostępnych w chmurze i infrastrukturze *on-premises*. Współczesne rozwiązania pozwalają na projektowanie, wdrażanie i monitorowanie procesów w jednym środowisku, co minimalizuje ryzyko utraty informacji.

Koncepcja systemu zarządzania procesami w przedsiębiorstwie EPMS (*Enterprise Process Management System*) wykracza poza automatyzację działań, traktując proces jako główną jednostkę w architekturze korporacyjnej, ważniejszą od danych. To przesunięcie zmieniło postrzeganie integracji systemów i roli procesów jako spoiwa firmy. W początkach XXI wieku liczne konferencje i rosnąca dojrzałość dziedziny sprawiły, że BPM zaczął być postrzegany jako odrębna dyscyplina, obejmująca aspekty techniczne i kulturowe. Z czasem skupiono się na standaryzacji, co doprowadziło do powstania notacji BPMN, ułatwiającej współpracę IT i biznesu bez tłumaczeń, skracając czas wdrożeń.

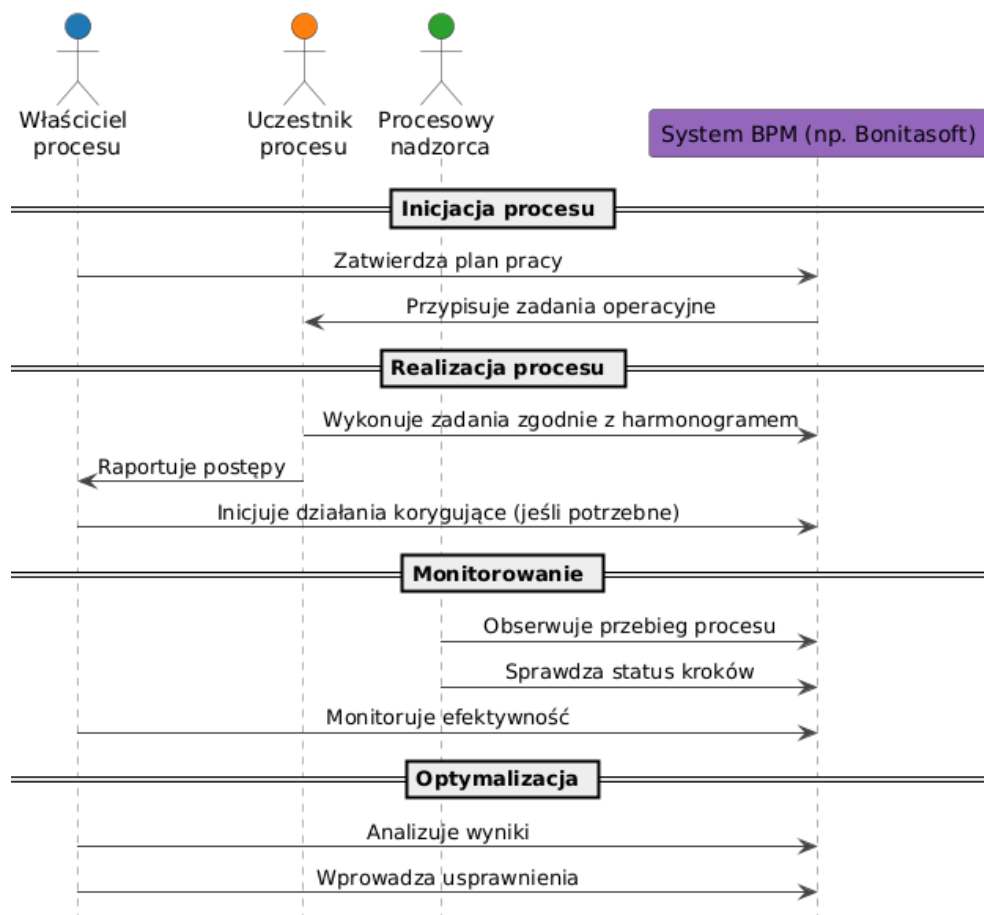
Obecnie BPM integruje się z technologiami *low-code* i analityką predykcyjną, co poprawia zdolność szybkiego reagowania na zmiany w otoczeniu i obniża ryzyko związane z inwestycjami w systemy BPMS. Platformy oferują komponenty do integracji i monitorowania KPI w czasie rzeczywistym, pozwalając na szybkie wprowadzanie usprawnień procesów.

1.3. Role i odpowiedzialności w BPM

Analizując strukturę procesów, należy uwzględnić osoby i role zaangażowane w ich realizację. W BPM funkcje są zróżnicowane: od właścicieli procesów, przez wykonawców, po osoby nadzorujące i analizujące. Każda rola wiąże się z innymi obowiązkami, a ich odpowiednie zdefiniowanie jest kluczowe dla skuteczności wdrożenia (Rys. 4).

[Tekst alternatywny: Diagram sekwencji ról i odpowiedzialności w BPM z czterema rolami w układzie kolumnowym: Właściciel procesu, Uczestnik procesu, Procesowy

nadzorca, System BPM. Fazy są wyraźnie wizualizowane jako poziome sekcje oznaczone jako etykiety z obramowaniem - „Inicjacja procesu”, „Realizacja procesu”, „Monitorowanie”, „Optymalizacja”. Przepływ komunikatów przebiega od lewej do prawej, zgodnie z sekwencją faz od góry do dołu.].



Rys. 4. Role i odpowiedzialności w BPM.
Źródło: opracowanie własne.

Diagram ukazuje cykliczny charakter zarządzania procesami - współpracę ludzi i systemu, której celem jest doskonalenie organizacji poprzez planowanie, realizację, kontrolę i optymalizację działań. Każdy z aktorów pełni odrębną funkcję w cyklu życia procesu - od inicjacji, przez realizację i monitorowanie, aż po optymalizację:

- W fazie inicjacji właściciel procesu zatwierdza plan pracy, a system BPM przypisuje zadania uczestnikowi procesu.
- W fazie realizacji uczestnik wykonuje przypisane zadania zgodnie z harmonogramem i raportuje postępy właścicielowi procesu, który w razie potrzeby inicjuje działania korygujące. W etapie monitorowania nadzorca



procesu obserwuje przebieg i status działań, a właściciel monitoruje efektywność całego procesu.

- Faza optymalizacji obejmuje analizę wyników oraz wprowadzanie usprawnień przez właściciela procesu w systemie BPM.

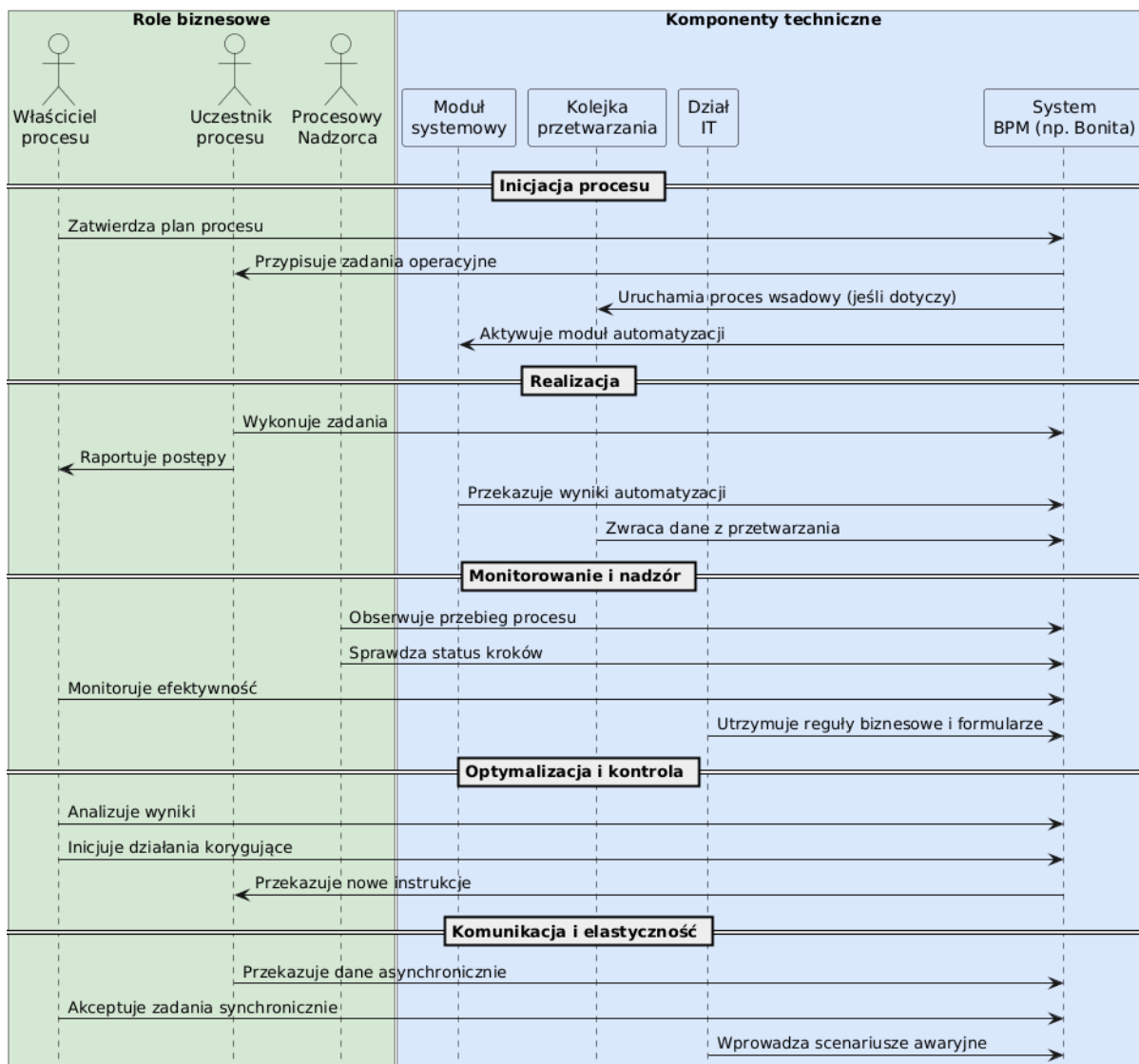
Podczas projektowania aplikacji procesowej w systemie klasy BPMS nadawanie ról przebiega etapami:

1. Utworzenie listy wszystkich zadań występujących w procesie.
2. Określenie wymaganych kompetencji dla każdego zadania - technicznych, organizacyjnych lub decyzyjnych.
3. Grupowanie zadań pod kątem możliwości przypisania ich do jednej roli.
4. Skonfigurowanie ról w systemie klasy BMS wraz z przypisaniem użytkowników lub grup.

Przedstawiony układ ról umożliwia zachowanie spójności pomiędzy teoretycznym modelem a jego praktyczną realizacją. Organizacyjne podejście w ramach BPM akcentuje znaczenie jednoznacznego przypisania odpowiedzialności, które wykracza poza formalny zakres stanowisk – powinno również gwarantować, że każdy agent, zarówno osoba jak i system, dysponuje niezbędnymi zasobami do wykonania powierzonych obowiązków. W kontekście procesów wsadowych funkcję agenta może pełnić kolejka przetwarzająca zadania, natomiast w działaniach zautomatyzowanych – dedykowany moduł systemowy obsługujący daną operację.

Na Rys. 5 przedstawiono role i odpowiedzialności w zarządzaniu procesami biznesowymi z uwzględnieniem komponentu technicznego i komunikacji.

[Tekst alternatywny: Diagram pokazuje sekwencję współpracy ról biznesowych (po lewej, zielone tło: Właściciel procesu, Uczestnik procesu, Procesowy Nadzorca) z komponentami technicznymi (po prawej, niebieskie tło: Moduł systemowy, Kolejka przetwarzania, Dział IT, System BPM/Bonita). Fazy są wyraźnie wizualizowane jako poziome sekcje oznaczone jako etykiety z obramowaniem - „Inicjacja”, „Realizacja procesu”, „Monitorowanie i nadzór”, „Optymalizacja i kontrola”, „Komunikacja i elastyczność”. Przepływ komunikatów przebiega od lewej do prawej, zgodnie z sekwencją faz od góry do dołu.].



Rys. 5. Role, odpowiedzialności i komunikacja w BPA.

Źródło: opracowanie własne.

W fazie inicjacji właściciel zatwierdza plan, system przypisuje zadania, a dział IT uruchamia proces wsadowy i moduł automatyzacji. W fazie realizacji uczestnik wykonuje zadania, system zwraca dane i wyniki automatyzacji. W fazie monitorowania nadzorca obserwuje przebieg i status, właściciel śledzi efektywność, a dział IT utrzymuje reguły i formularze. W fazie optymalizacji właściciel analizuje wyniki, inicjuje korekty, a system przyjmuje nowe instrukcje. W fazie komunikacji uczestnik akceptuje zadania synchronicznie i przekazuje dane asynchronicznie, a system obsługuje scenariusze awaryjne.



Dział IT pełni rolę zaplecza, odpowiadając za wdrażanie reguł biznesowych i zarządzanie przepływem formularzy, poza samym workflow. Chociaż formalnie nie uczestniczą bezpośrednio w procesie biznesowym i wykonywaniu działań operacyjnych, ich praca jest kluczowa dla dostępności i efektywności całego systemu. Wdrożenie aplikacji procesowej wymaga również ustalenia mechanizmów komunikacji między rolami. Te powiązania mogą mieć charakter synchroniczny, na przykład bezpośrednia akceptacja zadania przez właściciela, lub asynchroniczny, polegający na przekazaniu wyników pracy do kolejnego etapu bez konieczności natychmiastowej interakcji.

Przydzielanie członków zespołu do określonych ról zwykle opiera się na ich bieżących kompetencjach, ale równie ważne jest planowanie przyszłych rotacji i scenariuszy awaryjnych. Elastyczność struktury ról minimalizuje ryzyko zakłóceń, na przykład w przypadku nieobecności pracownika lub awarii systemu; dobrze zdefiniowane role i jasne odpowiedzialności są kluczowe dla skutecznej automatyzacji procesów biznesowych. Brak przejrzystości w tym zakresie może prowadzić do nieporozumień i błędów, które można uniknąć poprzez staranne zaprojektowanie tej warstwy już na etapie modelowania aplikacji procesowej.

Dobrze zaplanowane role w praktyce ułatwiają wdrażanie metod analizy i kontroli jakości procesu. Można wyróżnić mechanizmy przypisywania obowiązków interesariuszom oraz stosowania reguł kontrolnych na różnych etapach. Zrozumienie tych zależności wspiera późniejsze tworzenie dashboardów, które monitorują efektywność i identyfikują potencjalne opóźnienia.



2. Automatyzacja procesów biznesowych

2.2. Pojęcie automatyzacji procesów biznesowych

Automatyzacja procesów to zastosowanie technologii informatycznych i telekomunikacyjnych w celu realizacji zadań wcześniej wykonywanych manualnie. Opiera się na regułach biznesowych, danych wejściowych i jasno zdefiniowanych sekwencjach działań, tak aby system mógł przejąć zadania rutynowe lub wymagające ścisłej powtarzalności.

Automatyzacja może być realizowana przez systemy informatyczne typu BPMS, narzędzia RPA lub integrację usług zewnętrznych poprzez API. Proces implementacji można opisać krokami:

1. Identyfikacja czynności powtarzalnych oraz wymagających dużej ilości wejść danych; odnotowujemy te o wysokim ryzyku błędu manualnego.
2. Projektowanie elementów wykonawczych procesu z przypisaniem technologii obsługowej – np. RPA do obsługi formularzy bankowych czy BPMS do kontroli przepływu informacji między modułami IT.
3. Ustawienie reguł decyzyjnych zapisanych w silniku workflow umożliwiających wybór ścieżki przebiegu zależnie od wartości danych procesowych.

W projektowaniu aplikacji procesowych należy uwzględnić następujące aspekty:

- **Zgodność regulacyjna.** Automatyczna kontrola przestrzegania zasad wewnętrznych organizacji czy norm branżowych pozwala uniknąć kosztownych błędów i naruszeń. Implementacja tej funkcjonalności może polegać przykładowo na sprawdzeniu kompletności dokumentacji przed akceptacją zamówienia – jeśli brakuje któregoś wymagania formalnego, system kieruje sprawę na tor uzupełnienia informacji zanim trafi ona do klienta.
- **Analiza wydajności.** Projektując aplikacje, warto integrować mechanizmy analityczne, które umożliwiają monitorowanie wydajności automatów w czasie rzeczywistym. Dzięki temu można śledzić, jak często uruchamiany jest dany bot lub zadanie oraz ile czasu im to zajmuje. Analiza tych informacji pozwala zidentyfikować miejsca przeciążenia lub niewydolności kodu konektorów.
- **Inteligentna automatyzacja** dla zapewnienia ciągłości biznesu. Systemy automatycznie decydują o ponownym uruchomieniu usług po awarii lub wyborze ścieżki priorytetowej dla krytycznych zamówień firmy. W praktyce mogą współpracować z monitorami infrastruktury IT, wysyłając sygnały do procesów naprawczych bez potrzeby zaangażowania człowieka.
- **Rozszerzalność** - możliwość dodania nowych kroków lub zmiany logiki bez konieczności gruntownej przebudowy całego modelu procesu. Rekomenduje



się projektowanie komponentowe: małe moduły wykonawcze łatwe do wymiany czy aktualizacji działające jako część większej architektury procesu.

2.1. Architektura systemów automatyzacji procesów

Automatyzacja procesów biznesowych (BPA, *Business Process Automation*) umożliwia standaryzację działań oraz zwiększa ich przejrzystość dla osób zarządzających. Umożliwia powtarzalną realizację zadań przy zachowaniu stałego poziomu jakości. Rozwój zarządzania procesami biznesowymi (BPM) przeszedł od modelowania przepływów pracy z wykorzystaniem diagramów i narzędzi typu sieci Petriego do skupienia się na doskonaleniu i innowacyjności procesowej. Zmiana ta była związana z koniecznością dostosowania procesów do zmieniających się warunków rynkowych i technologicznych. Architektura BPA opiera się na kilku współpracujących warstwach (Tabela 1).

Tabela 1. Model warstwowy architektury automatyzacji procesów BPA.

Nr	Warstwa	Opis
1	Warstwa prezentacji	Interfejsy web/mobile i formularze, przez które użytkownicy inicjują sprawy, realizują zadania i wprowadzają dane. Obejmuje portale zadań, powiadomienia, dostępność (a11y) i UX. Zapewnia spójne doświadczenie użytkownika i walidację danych na wejściu.
2	Warstwa orkiestracji procesów	Silnik BPMN 2.0 zarządza przebiegiem procesu: start/stop instancji, bramki, równoległość, eskalacje, SLA i przypisania. Koordynuje ludzi, systemy i wyjątki, zapewniając stan i audit trail każdego kroku.
3	Warstwa decyzji	Reguły i decyzje biznesowe modelowane w DMN/DRD (np. tabele decyzyjne, FEEL) oddzielone od przepływu. Umożliwia niezależne wersjonowanie, testy i szybkie zmiany polityk, bez modyfikacji modeli procesów.
4	Warstwa integracji i automatyzacji zadań	Realizacja prac technicznych przez mikroserwisy, integratory, skrypty i roboty RPA (gdy brak API). Typowy wzorzec: External Task/worker pool — silnik publikuje zadanie, worker wykonuje i zwraca wynik; łatwa skalowalność i odporność.
5	Warstwa komunikacji	Połączenia synchroniczne (REST/gRPC) dla zapytań i aktualizacji oraz architektura zdarzeniowa (EDA) z brokerem wiadomości (np. Kafka) dla luźnego powiązania, kolejek pracy i komunikacji międzydomenowej. Obsługuje także scenariusze wsadowe i asynchroniczne.
6	Warstwa danych	Systemy źródłowe (ERP/CRM/MES), bazy transakcyjne, magazyn/jezioro danych (DWH/Data Lake) i repozytoria historii procesów. Odpowiada za jakość danych, wersjonowanie schematów, retencję i udostępnianie danych do analityki.



7	Warstwa obserwowalności i doskonalenia	Monitoring, logi, metryki, śledzenie end-to-end i audyt. Wykorzystuje Process Mining do odkrywania rzeczywistych ścieżek, conformance checking i identyfikacji wąskich gardeł, zamykając pętlę ciągłego doskonalenia (CIP).
8	Zarządzanie i bezpieczeństwo	IAM/SSO/RBAC, separacja ról, polityki zgodności (np. RODO), szyfrowanie, kopie zapasowe i DR. Obejmuje też ład modelowy: wersjonowanie BPMN/DMN, cykle przeglądów, testy i kontrolę zmian na środowiskach (dev/test/prod).

Integracja BPMS z systemami przedsiębiorstwa polega na tym, że silnik procesu nie zastępuje ERP/CRM/MES, lecz koordynuje ich funkcje. Silnik orkiestruje ich funkcje: decyduje co i kiedy wywołać, zbiera wyniki i przechowuje ślad audytowy.

Przykładowe opcje integracji to:

- API synchroniczne (REST/gRPC) do szybkich zapytań i aktualizacji,
- zdarzenia domenowe (Kafka lub podobne) dla komunikacji o luźnym powiązaniu i dużej przepustowości,
- kolejki wsadowe do przetwarzania dużych zadań,
- boty RPA tam, gdzie brak API lub interfejs GUI jest jedynym rozwiązaniem (tymczasowe „mosty”). Wzorzec *External Task* upraszcza zarządzanie RPA w procesach, bez „wpychania” kodu do silnika.

Przykładowa implementacja wzorca *External Task*:

1. Silnik *workflow* wystawia zadanie z tematem (np. *zamówienie wysyłki*).
2. Zewnętrzny *worker* (Python/Java/JS) *polluje* silnik, pobiera pracę, realizuje odpowiednie połączenie (REST, RPA, SQL), a potem zwraca wynik.

Stosowane topologie wdrożeniowe:

- BPMS jako centralny silnik (monolityczno-hubowe podejście) – szybszy start, prosta administracja; integracje przez adaptery i API.
- Microservices + „orchestrator” – silnik procesu lekki, a logika domenowa w usługach. Zadania techniczne realizowane przez *External Task*; komunikacja przez REST + zdarzenia. Skuteczne przy dużej skali i niezależnym rozwijaniu domen.

2.2. Silnik workflow

Silnik *workflow* (dalej: silnik) jest centralnym komponentem większości systemów klasy BPMS i odgrywa rolę koordynatora wykonania zdefiniowanego w modelu procesu. W praktyce oznacza to, że model przygotowany w Bonita Studio (z wykorzystaniem BPMN) zostaje załadowany do silnika, który następnie steruje

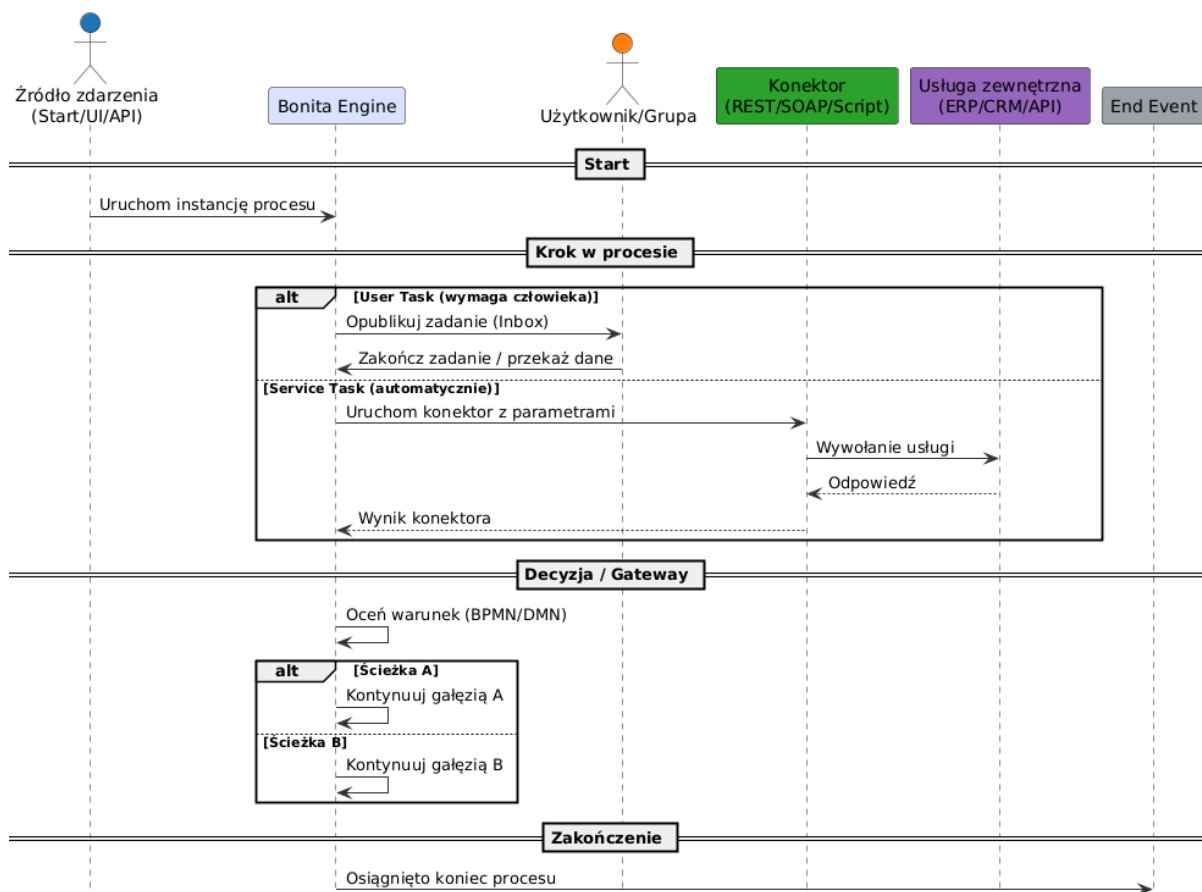


kolejnymi krokami wykonywania - przydziela zadania uczestnikom, inicjuje usługi systemowe oraz monitoruje ich wyniki.

Z technicznego punktu widzenia silnik obsługuje tzw. instancje procesów, czyli konkretne realizacje schematu uruchamiane na podstawie zdarzenia początkowego. Każda instancja zawiera tokeny wskazujące aktualną pozycję w przebiegu procesu; przesuwanie tych tokenów pomiędzy elementami modelu to główne zadanie silnika. Kluczowym zadaniem silnika jest odróżnianie operacji wymagających udziału człowieka od tych, które są w pełni zautomatyzowane. Zadania manualne są przypisywane wybranym użytkownikom lub grupom, z jasno określonym terminem realizacji i priorytetem.

W platformie Bonitasoft odbywa się to poprzez mechanizm przypisania aktorów do zadań – gdy token dociera do elementu typu *user task*, silnik publikuje go w skrzynce zadań odpowiednich osób, oczekując na ich interakcję. Dopiero po odebraniu wyniku czy danych wejściowych następuje aktualizacja stanu instancji i kontynuacja przebiegu. W przypadku działań automatycznych jak *service task*, to silnik zarządza komunikacją z interfejsami API lub usługami integracyjnymi. Może uruchomić konektor REST, SOAP czy skrypt systemowy bez udziału użytkownika, a następnie zinterpretować wynik - np. przekazać dane do kolejnego zadania albo uruchomić bramkę warunkową decydującą o dalszym przebiegu. Rola silnika obejmuje więc także kontrolę logiki warunkowej: to on wybiera ścieżkę zgodną z kryteriami zapisanymi w modelu (Rys. 6).

[Tekst alternatywny: Diagram sekwencji pokazuje uproszczony bieg instancji w Bonitasoft: start procesu, wykonanie kroku jako zadanie użytkownika lub automatyczny service task (konektor → usługa zewnętrzna), następnie decyzja w bramce (gałąź A lub B) i zakończenie procesu. Poziome linie podziału wyróżniają etapy: Start, Krok w procesie, Decyzja oraz Zakończenie.]



Rys. 6. Działanie silnika *workflow* w Bonitasoft.

Źródło: opracowanie własne.

Ważnym aspektem jest traktowanie silnika *workflow* go jako „uczestnika” procesu posiadającego własny zasób i zakres odpowiedzialności. W Bonita Studio można zastosować odrębną pulę procesów (*pool*) dla silnika *workflow*, aby oddzielić działania typowo systemowe od czynności ludzkich. Pozwala to analizować i optymalizować obciążenie technologiczne niezależnie od obciążenia personelu.

Projektowanie aplikacji procesowej w Bonitasoft odbywa się w trzech krokach:

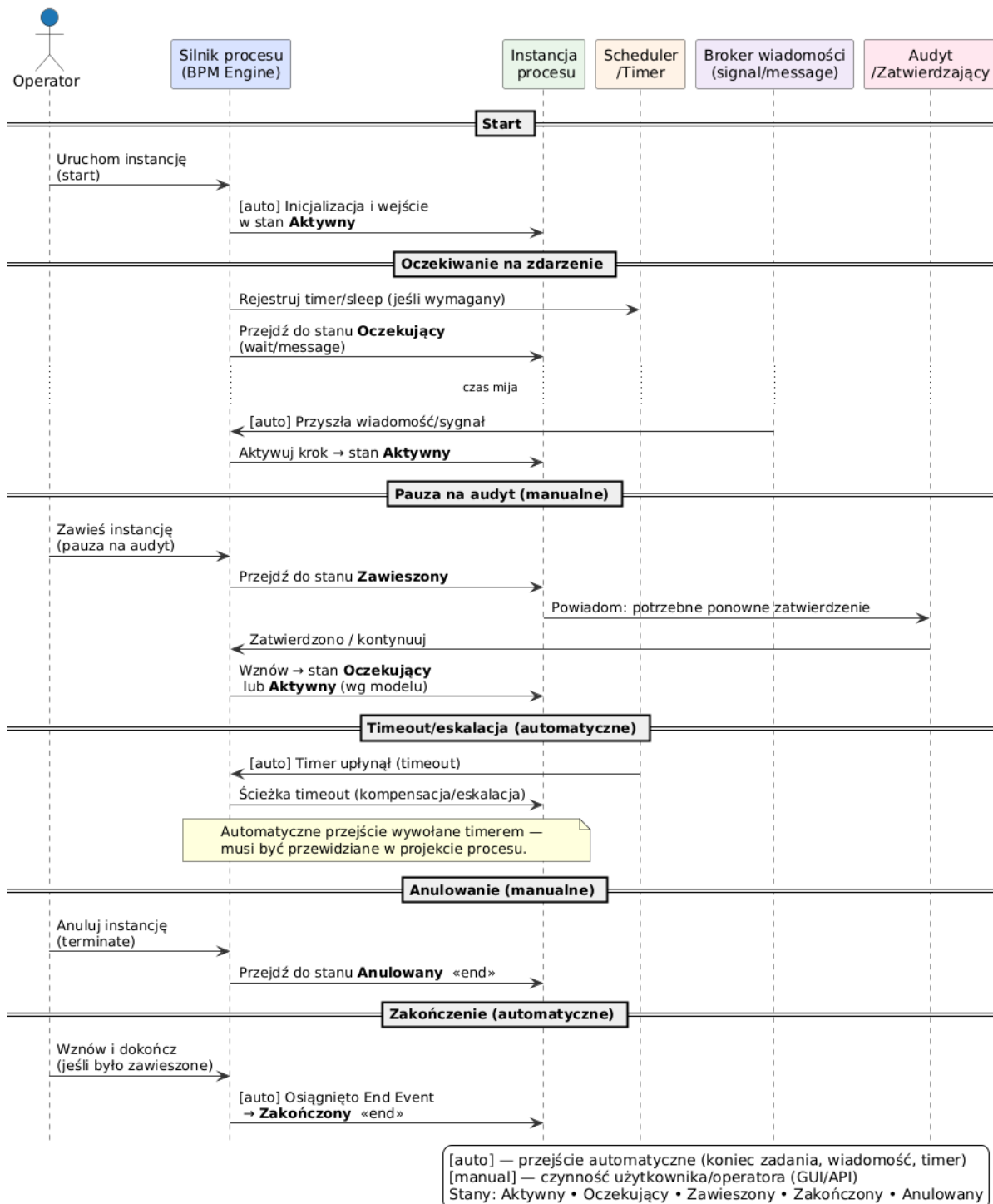
1. Umieść w diagramie zarówno zadania manualne jak i automatyczne, konfigurując je zgodnie ze scenariuszem biznesowym.
2. Określ parametry każdego działania – ustal formularze dla zadań manualnych (user task), walidatory i powiązane role; przygotuj konektory lub skrypty integracyjne wraz z parametrami wywołań dla zadań automatycznych (service task).
3. Po zakończeniu modelowania publikuj proces w Bonita Portal – jest to moment, gdy silnik zaczyna faktyczne wykonanie instancji.



Znajomość pojęcia stanu procesu jest kluczowa dla zrozumienia technicznych aspektów funkcjonowania silnika workflow. Każda instancja procesu może występować jako aktywna, oczekująca lub zakończona; istnieje również możliwość zawieszenia lub anulowania jej przebiegu przez administratora systemu. Silnik umożliwia zarówno manualne, jak i automatyczne przejścia pomiędzy poszczególnymi stanami, które powinny zostać uwzględnione w projekcie aplikacji procesowej. Takie rozwiązania pozwalają na implementację mechanizmów wstrzymywania działań, na przykład na czas audytu lub ponownej akceptacji etapów procesu.

Warto podkreślić, że mimo możliwości pełnej kontroli nad przepływem czynności automatycznych i półautomatycznych przez silnik, decyzje podejmowane przez użytkowników nadal wpływają na sekwencję realizowanych działań. Tym samym przebieg procesu stanowi połączenie technicznej orkiestracji z aktywną współpracą uczestników. Mechanizm zastosowania stanów instancji procesu do sterowania przepływem procesu przedstawiono na Rys. 7.

[Tekst alternatywny: Diagram sekwencji pokazuje przebieg stanów instancji procesu w systemie BPM z udziałem: Operatora, Silnika procesu, Instancji procesu, Scheduler/Timer, Brokera wiadomości oraz Audytu/Zatwierdzającego. Po uruchomieniu przez Operatora silnik inicjuje instancję w stanie Aktywny. Gdy proces musi czekać na zdarzenie lub upływanie czasu, silnik rejestruje timer i przełącza instancję w stan Oczekujący. Po nadejściu wiadomości z Brokera (np. sygnał, komunikat) instancja wraca do stanu Aktywny. Diagram wyróżnia przejścia automatyczne (oznaczone „[auto]”: koniec zadania, sygnał, timer/timeout) oraz manualne (akcje użytkownika/operatora). Operator może zawiesić instancję (stan Zawieszony) na czas audytu; po zatwierdzeniu przez Audyt instancja jest wznowiana i kontynuuje bieg. Timer może wywołać ścieżkę timeout/eskalacji. Operator ma też możliwość anulowania instancji (stan końcowy Anulowany). W scenariuszu podstawowym proces kończy się automatycznie po osiągnięciu *End Event* (stan Zakończony). Diagram akcentuje, że projekt aplikacji procesowej powinien przewidywać zarówno przejścia automatyczne, jak i manualne.]



Rys. 7. Stany instancji procesu i przejścia (manualne vs automatyczne).

Źródło: opracowanie własne.



Silnik *workflow* pełni funkcję centralnego komponentu odpowiedzialnego za koordynację komunikacji pomiędzy różnymi systemami, zarówno wewnętrznymi, jak i zewnętrznymi. Realizuje tzw. usługi orkiestracji (*orchestration services*) zgodnie ze specyfikacją BPMN, umożliwiając sekwencyjne zestawianie wielu wywołań serwisów aplikacyjnych lub ich równoległe wykonanie w oparciu o reguły synchronizacji określone w równoległych bramkach. Istotnym aspektem implementacji jest uwzględnienie statusów sukcesu oraz błędów zwracanych przez integrowane API, a także zaprojektowanie alternatywnych scenariuszy obsługi tych zdarzeń.

Podczas projektowania architektury procesu należy uwzględnić kwestie wydajności. Zbyt duża liczba równoległych zadań automatycznych oraz intensywne zapytania do serwisów zewnętrznych mogą prowadzić do przeciążenia zasobów silnika. Wskazane jest wykorzystanie dostępnych w Bonita Portal narzędzi monitorujących do pomiaru wydajności oraz analiza raportów dotyczących czasu i pamięci wykorzystywanych przez poszczególne kroki.

Ostatnim etapem projektowania aplikacji procesowej jest analiza danych uzyskanych z silnika workflow i ich wykorzystanie do modyfikacji modelu BPMN. Może to obejmować skrócenie sekwencji działań w zależności od czasu odpowiedzi serwisów, dodanie asynchronicznych podprocesów w miejscach, gdzie występują nieuniknione opóźnienia oraz wdrożenie rozwiązań zwiększających bezpieczeństwo danych przesyłanych między elementami poprzez zastosowanie kontroli dostępu opartej na rolach dostępnych na platformie.

2.3. Robotyzacja procesów a zarządzanie procesami biznesowymi

Relacja pomiędzy robotyzacją procesów biznesowych (RPA) a zarządzaniem procesami biznesowymi (BPM) jest zazwyczaj interpretowana jako powiązanie dwóch kluczowych poziomów funkcjonowania organizacji: operacyjnego oraz strategicznego:

1. RPA dotyczy realizacji ściśle określonych, powtarzalnych procesów w środowisku cyfrowym z wykorzystaniem programowych botów. Te narzędzia umożliwiają automatyczne wykonywanie takich zadań jak kopiowanie oraz aktualizacja danych, wypełnianie formularzy czy generowanie raportów.
2. BPM to podejście, które obejmuje cały proces od początku do końca i stanowi strukturę organizacyjną umożliwiającą optymalizację przebiegu strumieni wartości.

Oba podejścia są komplementarne – BPM pozwala precyzyjnie zidentyfikować obszary w procesie, gdzie automatyzacja przyniesie największe korzyści, natomiast



RPA jest narzędziem umożliwiającym skuteczne wdrożenie tej automatyzacji. Synergia pomiędzy BPM a RPA polega również na tym, że starannie zaprojektowany proces integruje automatyzację tam, gdzie zapewni ona istotną poprawę efektywności – mierzoną skróceniem czasu realizacji lub ograniczeniem liczby błędów wynikających z pracy manualnej. Dodatkowo, wysoki poziom jakości modeli BPMN tworzy stabilne punkty integracji dla botów, dzięki czemu przebieg procesu jest klarownie określony i przewidywalny dla mechanizmów inicjujących działania robotyczne. Kluczowe znaczenie ma także właściwe zarządzanie projektem wdrożeniowym obejmującym obie technologie.

Przykładowo: w środowisku Bonitasoft możliwe jest osadzenie rozwiązań RPA jako elementów większego modelu procesowego. Scenariuszem może być wdrożenie bota obsługującego przenoszenie danych między dwoma aplikacjami pozbawionymi wspólnego API – na etapie projektowania procesu w Bonita Studio dodaje się wtedy *service task* odpowiadający za uruchomienie bota RPA i przetworzenie wymaganych pól. Efekt działania bota zapisywany jest do zmiennych procesu, co umożliwia ich dalsze wykorzystanie w kolejnych krokach definiowanych przez model BPM.

Istnieje rozróżnienie pomiędzy integracją narzędziową a integracją metodyczną:

1. Integracja narzędziowa: silnik *workflow* współpracuje z komponentem RPA w tym samym przepływie zadań.
2. Integracja metodyczna - BPM pełni rolę mapy wartości biznesowej, dzięki której wiadomo, które kroki warto zautomatyzować za pomocą botów.

Taki podział upraszcza późniejszą analizę wydajności: wskaźniki z modułu monitorującego instancje procesu można uzupełnić o logi wykonania bota RPA.

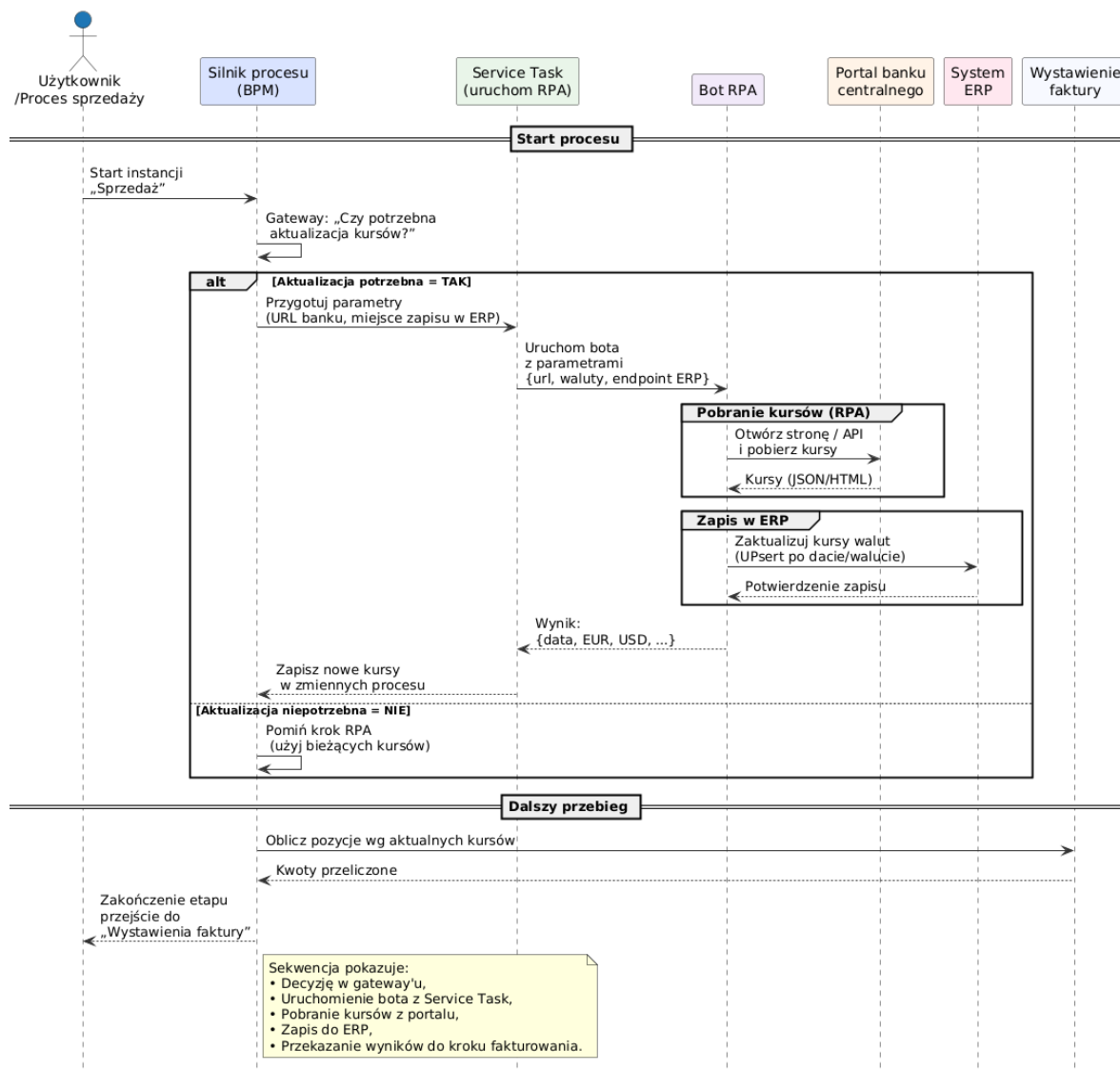
Przykład wdrożenia (aktualizacja kursów walut w systemie ERP):

- Proces sprzedaży sprawdza potrzebę aktualizacji kursów (*gateway*).
- *Service Task* uruchamia bota, przekazując adres strony banku centralnego i lokalizację zapisu w ERP.
- Bot pobiera kursy z portalu i zapisuje je w odpowiednich polach systemu ERP
- Wyniki automatyzacji (np. nowe kursy) trafiają do następnego etapu procesu, np. wystawienia faktury według zaktualizowanej stawki.

Na Rys. 8 przedstawiono diagram sekwencji dla przykładowej implementacji procesu.

[Tekst: diagram sekwencji ilustruje zautomatyzowany proces aktualizacji kursów walut w ERP w trakcie procesu sprzedaży, obejmujący konieczność wystawienia faktury. Po decyzji w bramce BPM o konieczności zaktualizowania kursów, silnik

uruchamia bota RPA, który pobiera najnowsze kursy z portalu banku centralnego i zapisuje je w systemie ERP. Zaktualizowane kursy są następnie przekazywane do kolejnego kroku – wystawienia faktury według zaktualizowanych kursów wymiany.]



Rys. 8. Diagram sekwencji dla automatyzacji procesu aktualizacji kursów walut.

Źródło: opracowanie własne.

Podczas projektowania w środowisku Bonitasoft aplikacji procesowej integrującej oba podejścia pomocne jest zastosowanie następującej procedury:

1. Identyfikacja kandydatów do automatyzacji

Wybierz zadania powtarzalne, o jasno zdefiniowanych regułach i najlepiej oparte na źródłach cyfrowych.



2. Specyfikacja reguł i danych wejściowych/wyjściowych

Zapisz warunki uruchomienia, wymagane parametry oraz oczekiwane rezultaty i format danych.

3. Modelowanie procesu w BPMN (Bonita Studio)

Umieść zadanie jako *Service Task* w odpowiednim miejscu przepływu.

4. Dodanie logiki sterującej (warunki)

Przed *Service Task* wstaw bramkę warunkową (*Gateway*), która decyduje o potrzebie uruchomienia automatyzacji.

5. Wybór/opracowanie robota RPA

Przygotuj lub wybierz istniejący skrypt/bota (np. *UiPath*, *Automation Anywhere*), który wykona zadanie na podstawie przekazanych parametrów procesu.

6. Integracja procesu z botem (konektory)

Skonfiguruj w Bonita konektor wywołujący bota (przekazanie parametrów) oraz konektor zwrotny odbierający wyniki pracy bota.

7. Obsługa scenariuszy awaryjnych

Zaprojektuj ścieżki na wypadek błędów typowych dla RPA (zmiana UI, opóźnienia, brak elementu). Uwzględnij komunikaty błędu zwracane przez bota.

8. Implementacja obsługi wyjątków w BPMN

Na zadaniu uruchamiającym bota umieść zdarzenie brzegowe *Boundary Event (Error/Timer)*. W razie błędu przekieruj przepływ na tor manualnej obsługi lub ponownej próby.

9. Testy i walidacja end-to-end

Sprawdź ścieżki pozytywne/negatywne, poprawność danych, idempotencję, retry oraz czasy odpowiedzi. Idempotencja to właściwość operacji, która daje taki sam wynik niezależnie od liczby powtórzeń. Jeśli wykonasz ją raz czy pięć razy – stan systemu pozostaje taki sam, jak po pierwszym poprawnym wywołaniu. Idempotencja pozwala powtarzać kroki bez skutków ubocznych, co jest kluczowe w automatyzacji procesów i integracjach rozproszonych.

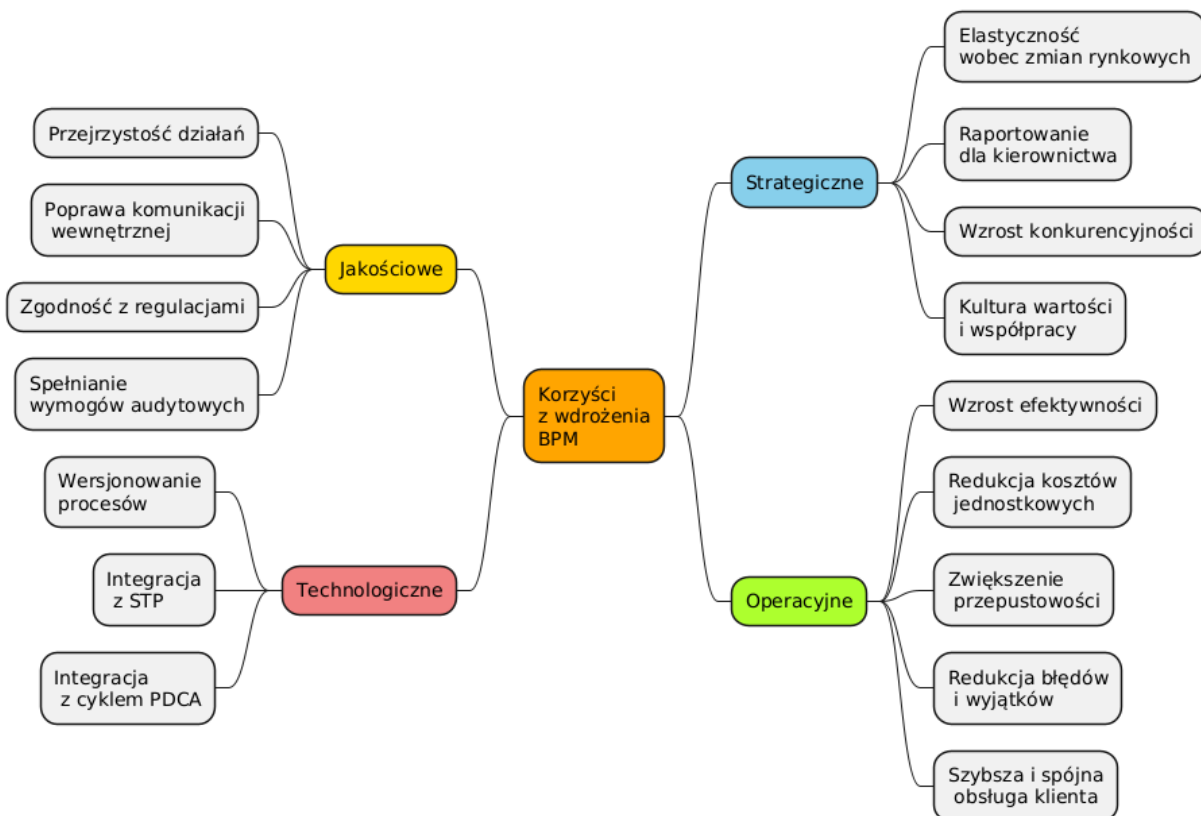
10. Monitorowanie i doskonalenie

Włącz logowanie i metryki; na podstawie raportów procesu i zachowania bota usprawniaj reguły, parametry i obsługę wyjątków.

2.4. Korzyści z automatyzacji procesów

Korzyści płynące z automatyzacji procesów biznesowych w środowisku takim jak Bonitasoft można rozpatrywać zarówno w kategoriach wymiernych efektów operacyjnych, jak i mniej oczywistych skutków organizacyjnych czy kulturowych (Rys. 9).

[Tekst alternatywny: Mapa myśli przedstawia korzyści z wdrożenia BPM w czterech obszarach: strategicznym, operacyjnym, jakościowym i technologicznym. Centralny węzeł „Korzyści z wdrożenia BPM” łączy te kategorie w spójny obraz wartości dla organizacji]



Rys. 9. Mapa korzyści płynących z automatyzacji procesów.

Źródło: opracowanie własne.

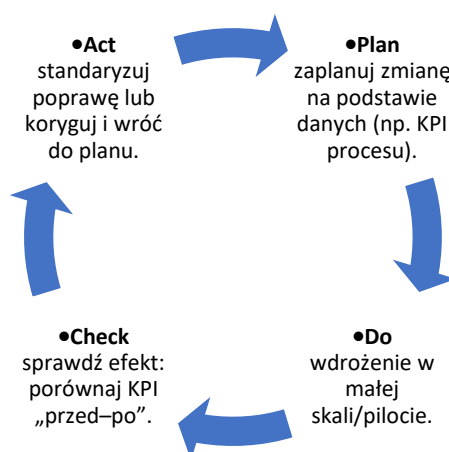
W wymiarze strategicznym automatyzacja procesów biznesowych powoduje większą elastyczność wobec zmian rynkowych, lepsze raportowanie dla kierownictwa, wzrost konkurencyjności oraz rozwój kultury wartości i współpracy. Operacyjnie BPM przynosi wzrost efektywności, redukcję kosztów jednostkowych, zwiększenie przepustowości, ograniczenie błędów i wyjątków oraz szybszą, spójną obsługę klienta. W obszarze jakościowym akcentowana jest przejrzystość działań, poprawa komunikacji wewnętrznej, zgodność z regulacjami i spełnianie wymogów audytowych. Z perspektywy technologicznej korzyści obejmują wersjonowanie procesów oraz integrację ze STP i cyklem PDCA.

Przetwarzanie bezpośrednie (STP, *Straight-Through Processing*) to metoda przetwarzania danych polegająca na automatycznym przesyłaniu informacji z wejścia do wyjścia w jednym ciągłym strumieniu.



Cykl ciągłego doskonalenia procesu PDCA (*Plan–Do–Check–Act*) wywodzi się z cyklu Shewharta, opracowanego w latach 30. XX wieku przez statystyka Waltera A. Shewharta (Bell Labs) jako iteracyjna metoda uczenia się na podstawie danych oraz podnoszenia jakości procesów, znana współcześnie jako cykl Deminga (Rys. 10).

[Tekst alternatywny: diagram przedstawia zamknięty cykl utworzony z czterech niebieskich, zakrzywionych strzałek ułożonych w okrąg (góra, prawa strona, dół, lewa strona). Strzałki wskazują kierunek zgodny z ruchem wskazówek zegara, ilustrując sekwencyjne i powtarzalne fazy cyklu PDCA: Plan – planowanie zmiany w oparciu o dane (np. wskaźniki KPI procesu); Do – wdrożenie rozwiązania w ograniczonej skali lub w formie pilotażu; Check – weryfikacja efektów poprzez porównanie wskaźników KPI „przed–po”; Act – standaryzacja usprawnienia bądź korekta działań i powrót do etapu planowania.]



Rys. 10. Cykl ciągłego doskonalenia procesu PDCA.
Źródło: opracowanie własne.

W praktyce zarządzania procesami biznesowymi, koncepcja STP umożliwia maksymalizację automatyzacji realizowanych zadań, podczas gdy wdrożenie cyklu PDCA zapewnia metodyczne doskonalenie procesu w sposób iteracyjny, zgodnie z ustalonym harmonogramem (np. kwartalnie) i w oparciu o pomiary oraz audyty.



3. Inteligentne zarządzanie procesami biznesowymi iBPM

3.1. Definicja iBPM

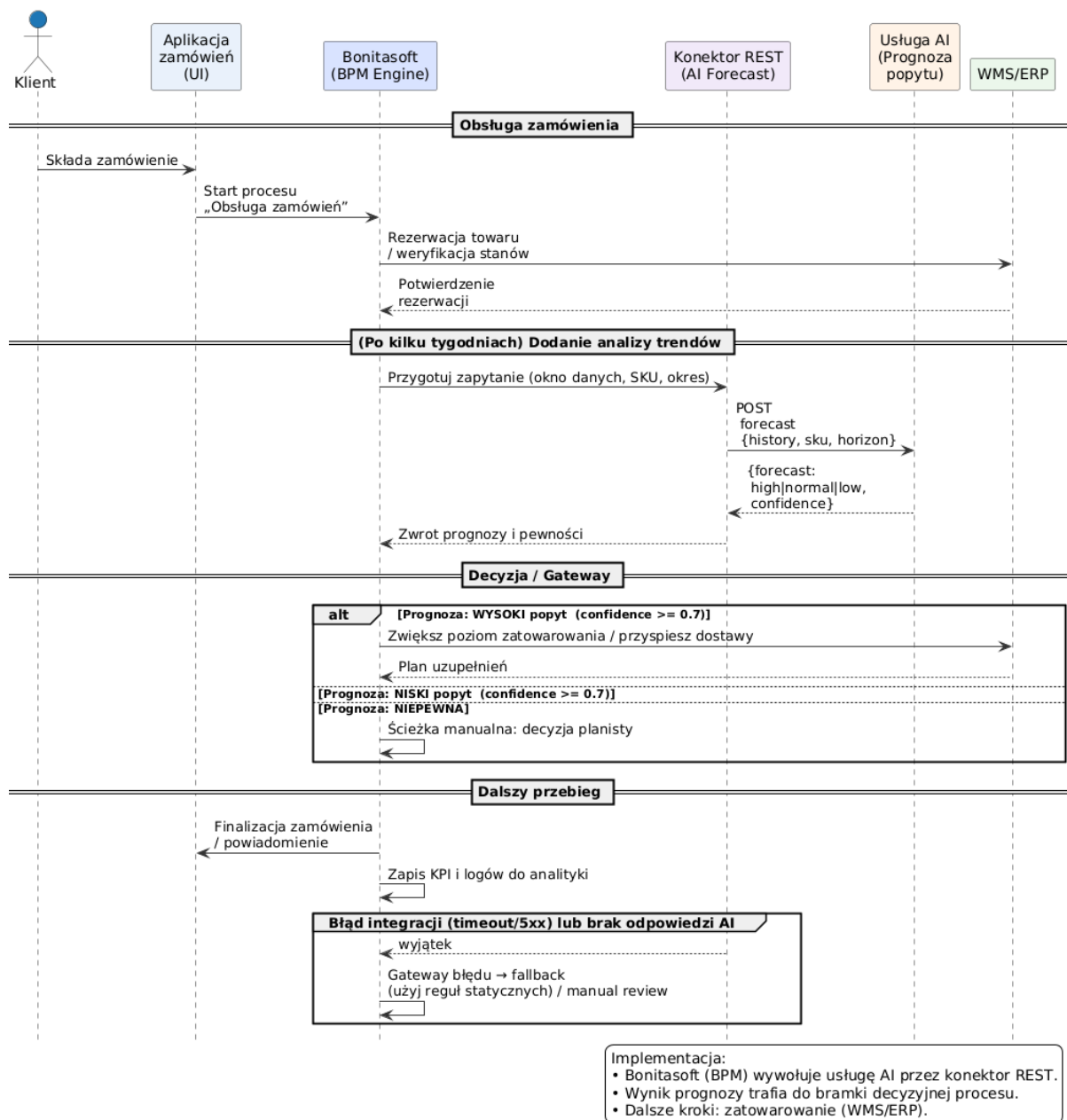
Inteligentne zarządzanie procesami biznesowymi (iBPM) stanowi zaawansowaną ewolucję tradycyjnego modelu BPM, w której do standardowych mechanizmów modelowania i realizacji procesów dodano elementy bazujące na technologiach sztucznej inteligencji, analizie danych w czasie rzeczywistym oraz szeroko zakrojonej automatyzacji. W praktyce oznacza to, że system BPMS zostaje wzbogacony o dodatkową warstwę inteligencji, umożliwiającą nie tylko realizację zdefiniowanych ścieżek działań, lecz także ich dynamiczną modyfikację w odpowiedzi na bieżące dane oraz zmieniające się warunki otoczenia.

W literaturze przedmiotu opisuje się iBPM jako zestaw funkcji integrujących reguły biznesowe, analizę predykcyjną oraz zdolność uczenia maszynowego w taki sposób, aby realizacja procesów przebiegała optymalnie nawet przy dużej zmienności otoczenia organizacji. Narzędzia tej klasy operują często w architekturze chmurowej z dostępem do rozproszonych źródeł danych. Dzięki temu możliwe jest gromadzenie i przetwarzanie informacji niemal w czasie rzeczywistym – od logów produkcyjnych po wskaźniki zachowań konsumentów z aplikacji mobilnych.

Silnik wykonawczy procesu może nie tylko realizować przewidywalne przesuwanie tokena po modelu BPMN, ale także modyfikować sekwencję działań w oparciu o aktualny kontekst, na przykład proponując alternatywne czynności lub ustalając priorytety zadań. Takie rozszerzenie znajduje odzwierciedlenie również w interfejsach użytkownika – panele konfiguracyjne typu low-code umożliwiają zespołom biznesowym dokonywanie zmian bez konieczności pełnego udziału programistów.

Przykładowo, projektując proces obsługi zamówień online, po kilku tygodniach od uruchomienia można dodać element analizy trendów sprzedaży, który będzie wpływał na poziom zatowarowania magazynu lub uruchamianie kampanii marketingowych zależnie od prognozy popytu. Tego rodzaju logikę można oprzeć na zewnętrznej usłudze AI połączonej z Bonitasoft przez konektor REST, której wynik stanowi dane wejściowe do bramki decyzyjnej decydującej o dalszym przebiegu (Rys. 11).

[Tekst alternatywny: Diagram sekwencji pokazuje proces obsługi zamówień online rozszerzony o krok analizy trendów sprzedaży. Silnik Bonitasoft wywołuje usługę AI przez konektor REST, otrzymuje prognozę popytu i na bramce decyzyjnej kieruje bieg: przy wysokim popycie zwiększa zatowarowanie w WMS/ERP, a przy niepewnej prognozie przekazuje sprawę do decyzji manualnej.]



Rys. 11. Diagram sekwencji dla przykładowej implementacji iBPM.

Źródło: opracowanie własne.



Równolegle coraz częściej wdrażana jest tzw. inteligentna automatyzacja (*intelligent automation*), łącząca robotyzację procesów (RPA) z uczeniem maszynowym. Bot wykonuje czynność techniczną, a moduł AI ocenia sens jej kolejnych powtórzeń lub sugeruje poprawki.

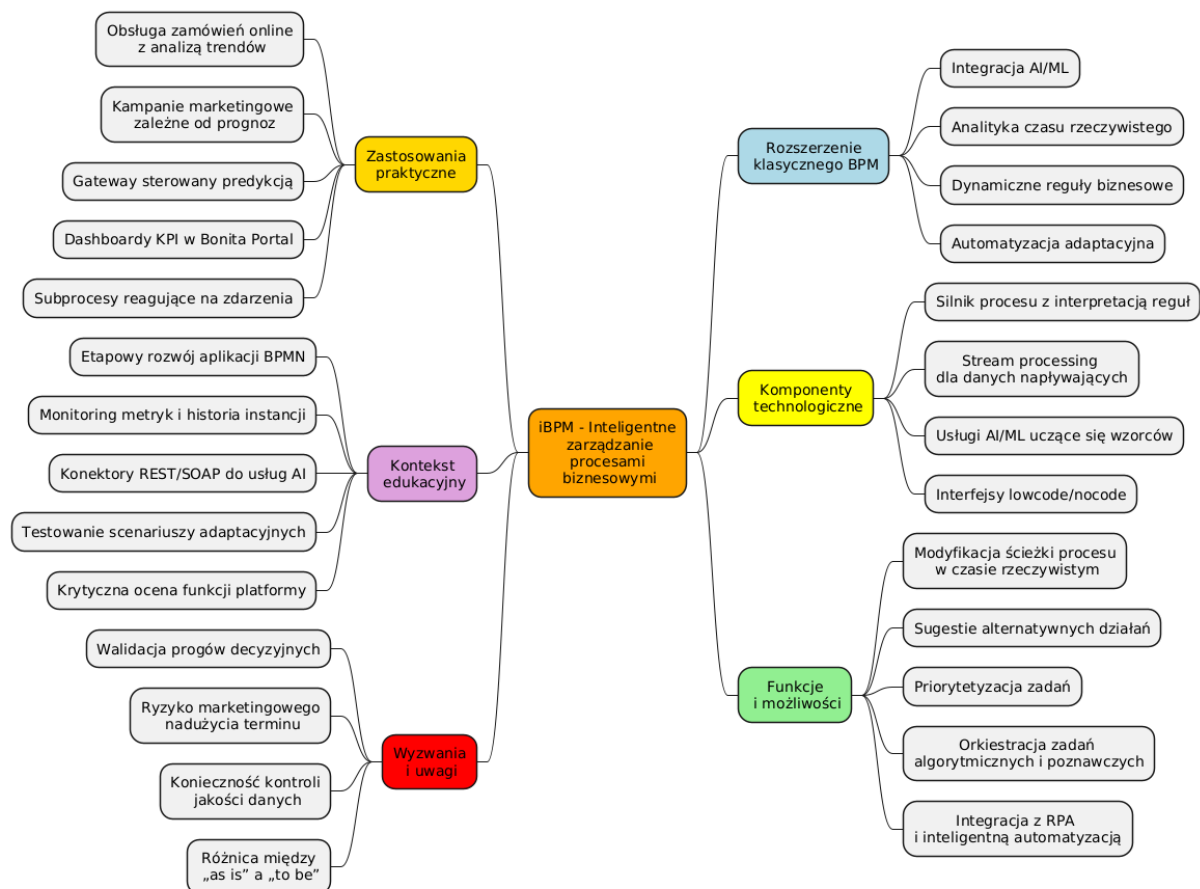
Podstawowe składniki ekosystemu iBPM obejmują:

- Silnik procesu zdolny do interpretacji dynamicznych reguł.
- Warstwę analityczną z modułem *stream processing* dla danych napływających.
- Integrację z usługami AI/ML uczącymi się wzorców postępowań.
- Interfejsy niskokodowe/bezkodowe (*lowcode/nocode*) do szybkich zmian logiki działania.

W systemie Bonitasoft wybrane funkcjonalności mogą być odwzorowane za pomocą integracji, takich jak konektory REST/SOAP umożliwiające wywoływanie usług chmurowych odpowiedzialnych za generowanie rekomendacji. Procesy podrzędne o zasięgu globalnym (*event subprocess*) mogą być inicjowane sygnałem informującym o przekroczeniu określonych wskaźników KPI. Natomiast w portalu Bonita możliwe jest tworzenie *dashboardów* prezentujących dane w czasie rzeczywistym.

Na Rys. 11 przedstawiono koncepcję iBPM w formie mapy pojęć.

[Tekst alternatywny: Mapa pojęć przedstawia iBPM — inteligentne zarządzanie procesami biznesowymi — jako połączenie klasycznego BPM z AI/ML i analityką czasu rzeczywistego. Pokazuje rozszerzenia (dynamiczne reguły, automatyzacja adaptacyjna, interpretacja reguł przez silnik), komponenty technologiczne (przetwarzanie strumieniowe, usługi uczące się wzorców, interfejsy low/no-code) oraz funkcje, takie jak modyfikacja ścieżki procesu w locie, sugestie działań, priorytetyzacja zadań, integracja z RPA. W zastosowaniach pojawiają się m.in. obsługa zamówień z analizą trendów, kampanie oparte na prognozach, bramki sterowane predykcją i dashboardy KPI. W kontekście edukacyjnym akcentowane są metryki, historia instancji, konektory REST/SOAP i testy scenariuszy adaptacyjnych, a w wyzwaniach — jakość danych, ryzyko nadużyć marketingowych i różnica między stanem „as is” a „to be”.]



Rys. 12. Mapa ekosystemu iBPM.

Źródło: opracowanie własne.

Poszczególne gałęzie drzewa mapy obejmują:

- „Rozszerzenie klasycznego BPM” przedstawia integrację z AI/ML, dynamiczne reguły biznesowe, automatyzację adaptacyjną oraz procesowy silnik interpretujący reguły.
- „Komponenty technologiczne” dotyczą przetwarzania strumieniowego danych, usług wykrywających wzorce oraz interfejsów niskokodowych/bezkodowych wspierających tworzenie rozwiązań.
- „Funkcje i możliwości” obejmują modyfikację ścieżki procesu w czasie rzeczywistym, sugerowanie alternatywnych działań, priorytetyzację zadań, orkiestrację zadań algorytmicznych i poznawczych oraz integrację z RPA i automatyzacją inteligentną.



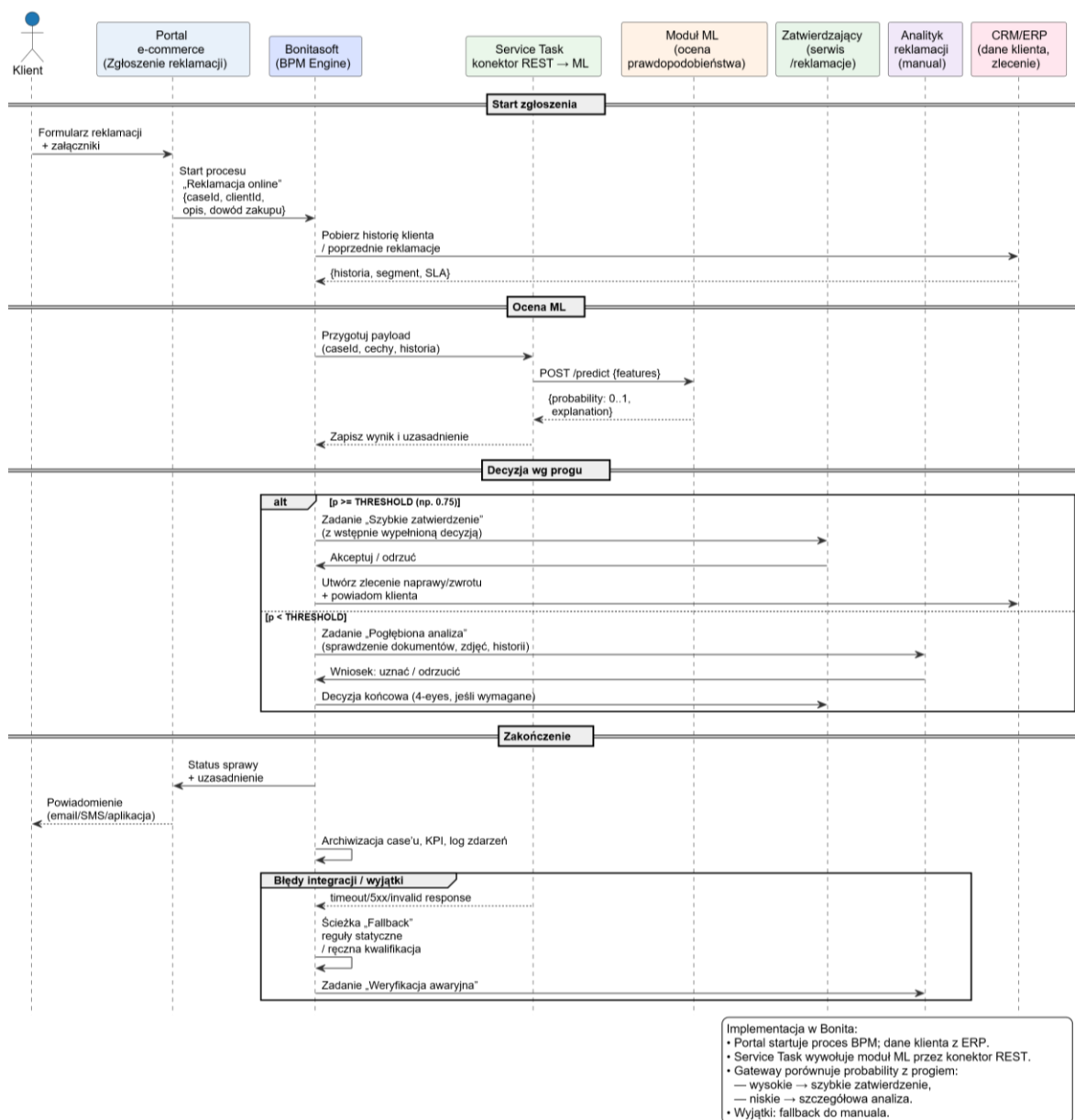
- „Zastosowania praktyczne” obejmują scenariusze, takie jak obsługa zamówień online z analizą trendów, kampanie marketingowe oparte na prognozach, bramki sterowane predykcją, dashboardy KPI w Bonita Portal, podprocesy reagujące na zdarzenia czy etapowy rozwój aplikacji BPMN.
- „Kontekst edukacyjny” wskazuje na monitoring metryk i historii instancji, wykorzystanie konektorów REST/SOAP do usług AI, testowanie scenariuszy adaptacyjnych oraz ocenę możliwości platformy, w tym walidację progów decyzyjnych.
- „Wyzwania i uwagi” obejmują ryzyko nadużycia terminu „AI” w działaniach marketingowych, potrzebę kontroli jakości danych oraz konieczność rozróżnienia między bieżącym stanem a docelowym.

3.2. Uczenie maszynowe w analizie procesów

Uczenie maszynowe w analizie procesów biznesowych umożliwia zastosowanie modeli, które wykorzystują dane historyczne oraz aktualne wyniki systemu do samodzielnego uczenia się. Takie podejście pozwala na odejście od statycznych reguł sterujących przebiegiem iBPM i wdrożenie architektury adaptacyjnej, w której decyzje dotyczące ścieżki procesu są podejmowane automatycznie na podstawie wzorców wykrywanych przez algorytmy ML. Systemy te wykorzystują różnorodne techniki, takie jak regresja liniowa, lasy losowe czy głębokie sieci neuronowe. Wybór metody uzależniony jest od typu analizowanych danych oraz założonych celów biznesowych.

Przykładowo, automatyzacja procesu obsługi reklamacji produktów online. Po otrzymaniu zgłoszenia system przesyła szczegóły do modułu ML oceniającego prawdopodobieństwo uznania reklamacji na podstawie historii klienta i wcześniejszych przypadków. Jeśli przewidywana wartość jest wysoka, proces kierowany jest bezpośrednio do zatwierdzenia; jeśli niska – trafia do pogłębionej analizy ręcznej (Rys. 13).

[Tekst alternatywny: diagram sekwencji pokazuje automatyzację obsługi reklamacji w Bonitasoft. Po złożeniu zgłoszenia portal uruchamia proces, który pobiera dane klienta z ERP i wywołuje moduł ML przez konektor REST, aby obliczyć prawdopodobieństwo uznania. Jeśli wynik przekracza próg, sprawa trafia do szybkiego zatwierdzenia; w przeciwnym razie przechodzi do pogłębionej analizy ręcznej, z obsługą ścieżki awaryjnej w razie błędów integracji.]



Rys. 13. Diagram sekwencji dla przykładowej implementacji iBML.

Źródło: opracowanie własne.

Rysunek przedstawia sekwencję automatyzacji procesu obsługi reklamacji. Klient składa zgłoszenie w portalu e-commerce, które inicjuje proces BPM wraz z danymi sprawy (ID, opis, załączniki). Silnik procesu pobiera z ERP/CRM historię klienta i wcześniejsze rozstrzygnięcia, po czym w kroku *Service Task* wywołuje zewnętrzny moduł ML przez konektor REST, przekazując cechy sprawy i kontekst. Moduł zwraca prawdopodobieństwo uznania reklamacji oraz ewentualne wyjaśnienia modelu; wynik jest zapisywany w zmiennych procesu.



Na bramce decyzyjnej porównuje się wartość prognozy z ustalonym progiem. Jeśli prognoza jest wysoka, sprawa jest zatwierdzana szybko przez uprawnionego pracownika (wariant „*straight-through*” z minimalną interwencją). Po akceptacji system generuje w ERP zlecenie naprawy lub zwrotu i wysyła klientowi powiadomienie. W przypadku niskiej prognozy sprawa kierowana jest do bardziej szczegółowej, ręcznej analizy (weryfikacja dokumentów i zdjęć), z możliwością zastosowania zasady „czterech oczu” przy podejmowaniu ostatecznej decyzji. Diagram obejmuje także ścieżkę wyjątków: gdy wystąpi błąd integracji (*timeout*, *5xx*, niepoprawna odpowiedź), proces przełącza się na I - reguły statyczne lub ręczną kwalifikację - aby zapewnić ciągłość obsługi. Całość procesu wspiera rejestrowanie KPI oraz logów zdarzeń i umożliwia późniejszą kalibrację progu decyzji oraz jakości modelu ML.

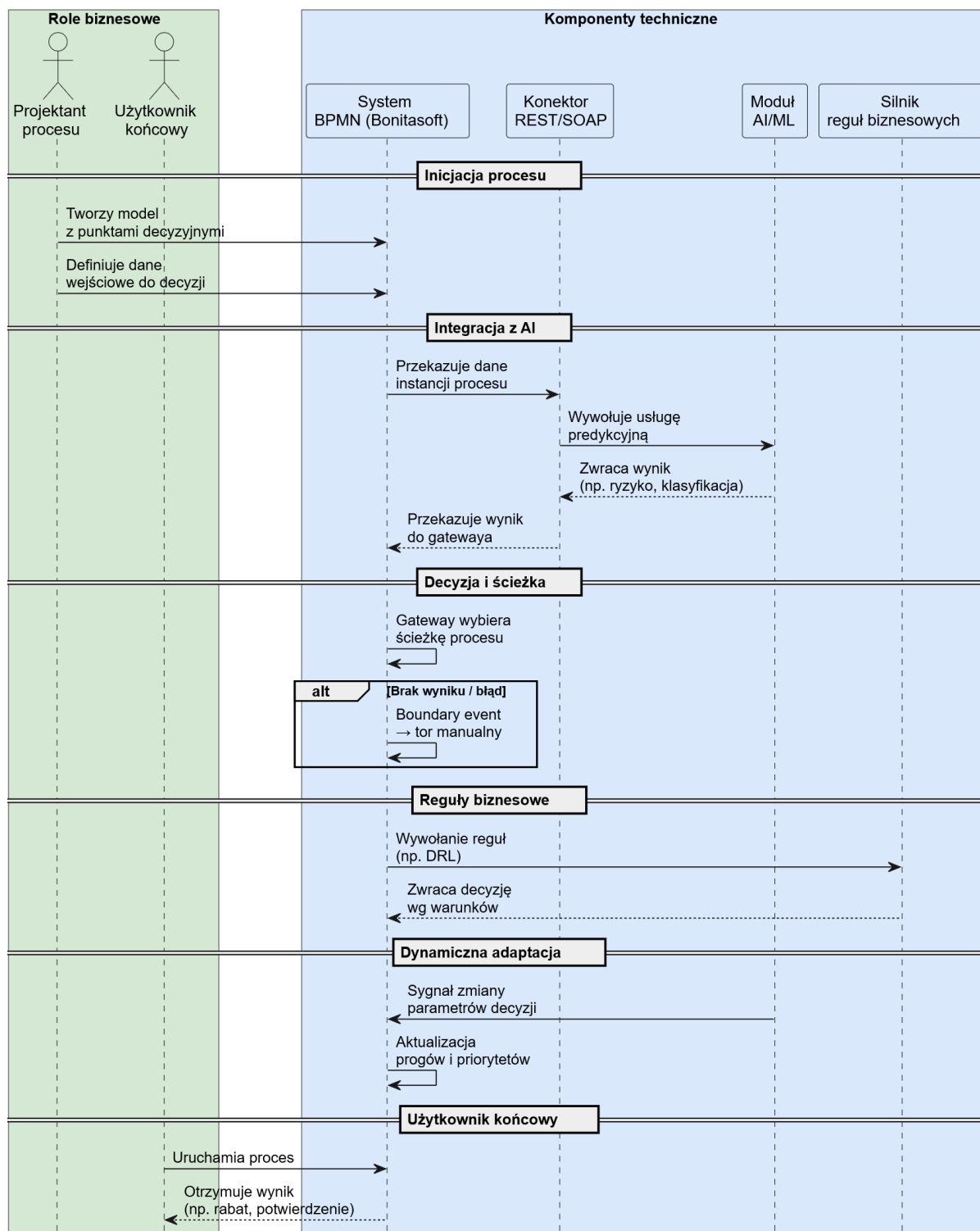
3.3. Automatyczne podejmowanie decyzji

Automatyzacja podejmowania decyzji to jedna z kluczowych funkcji systemów iBPM. System procesowy przestaje pełnić rolę biernego wykonawcy zdefiniowanej logiki, stając się aktywnym „uczestnikiem” procesu, który samodzielnie determinuje dalsze jego etapy na podstawie kontekstu oraz napływających danych. Od strony technicznej mechanizm ten integruje reguły biznesowe zapisane w silniku procesowym z rezultatami analiz predykcyjnych generowanych przez moduły AI/ML

Projektant definiuje niezbędne dane wejściowe (np. parametry transakcji, historię klienta, wyniki wcześniejszych kroków) i przekazuje je z instancji procesu do modułu analitycznego przez konektor REST/SOAP — lokalny lub chmurowy. Odpowiedź (klasyfikacja, prognoza czasu, ocena ryzyka) steruje wyborem gałęzi w bramce BPMN. Równocześnie przewidziana jest ścieżka awaryjna: zdarzenie brzegowe na zadaniu integracyjnym przejmuje błąd lub brak wyniku i kieruje instancję do obsługi manualnej. Dzięki temu proces nie blokuje się przy niedostępności modułu AI (Rys. 14).

[Tekst alternatywny: Diagram pokazuje automatyczne podejmowanie decyzji w iBPM. Po lewej role biznesowe: projektant procesu i użytkownik końcowy; po prawej komponenty techniczne: system BPM (Bonitasoft), konektor REST/SOAP, moduł AI/ML i silnik reguł. Przepływ obejmuje: inicjację procesu, przekazanie danych do AI, zwrot predykcji i decyzję bramki (z obejściem manualnym przy błędzie), ewentualne wywołanie reguł biznesowych oraz dynamiczną adaptację progów. Użytkownik uruchamia proces i otrzymuje wynik.]

Automatyczne podejmowanie decyzji w iBPM



Rys. 14. Automatyczne podejmowanie decyzji w iBML.

Źródło: opracowanie własne.

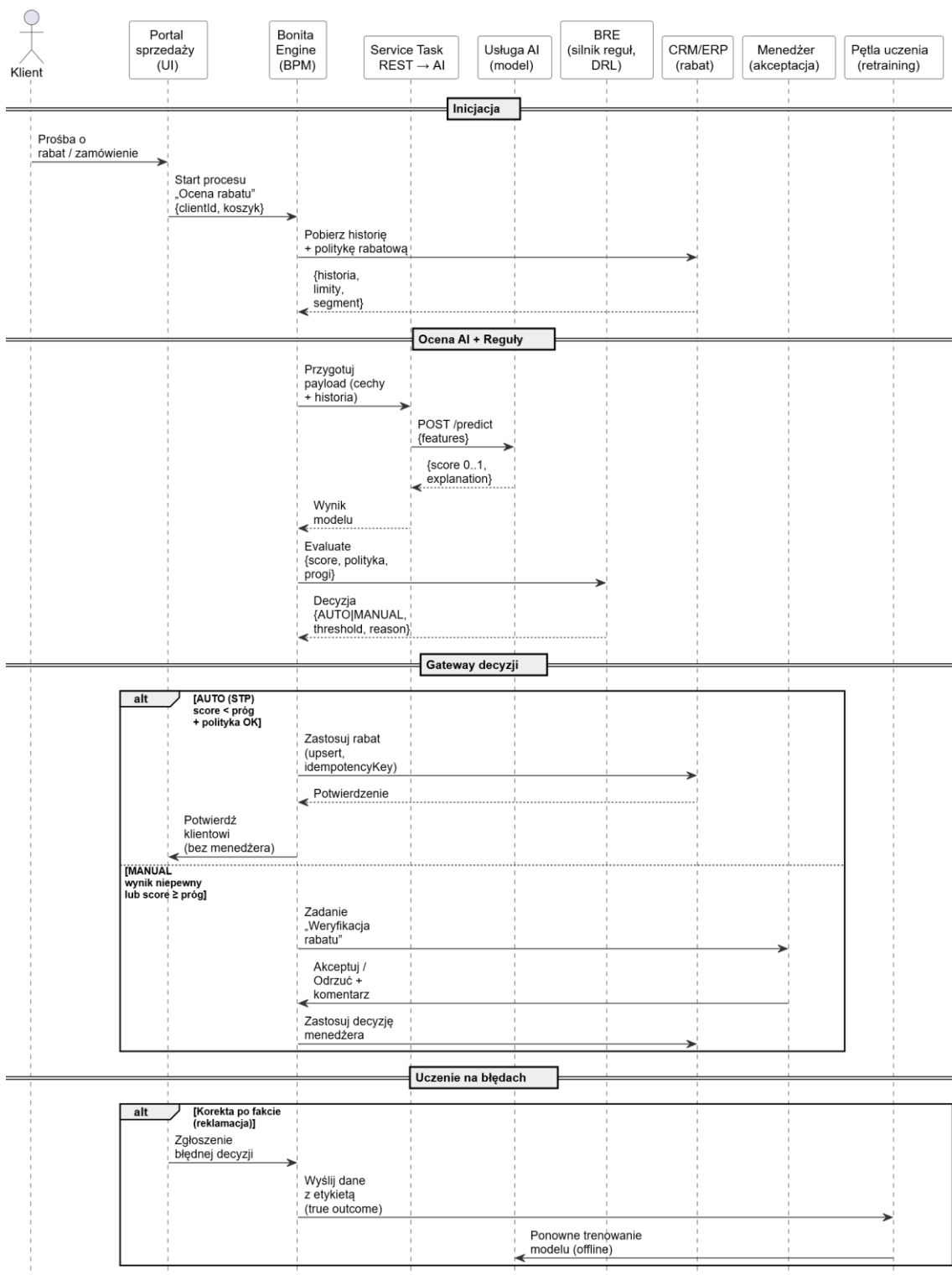


W zaawansowanych wdrożeniach reguły decyzyjne podlegają dynamicznej modyfikacji na podstawie bieżącej analizy efektywności procesu. Silnik workflow może otrzymywać zewnętrzne dyspozycje dotyczące zmiany parametrów progowych lub priorytetów realizacji wybranych typów spraw. Przykładowo, w przypadku stwierdzenia wzrostu wolumenu zamówień ekspresowych przy ograniczonych zasobach system automatycznie podwyższa próg wartości zamówienia kwalifikującego je do trybu „VIP”, aby utrzymać zdolność terminowej realizacji tych zamówień.

Automatyzacja podejmowania decyzji może opierać się zarówno na modelach uczących się, jak i klasycznych silnikach reguł biznesowych (*Business Rules Engine*). Silnik reguł pozwala na oddzielenie logiki warunkowej od modelu BPMN oraz jej modyfikację bez konieczności ingerencji w diagram procesu. Projektując tego typu rozwiązania, należy zwrócić szczególną uwagę na jakość danych wykorzystywanych w procesie decyzyjnym oraz właściwą dokumentację mechanizmu podejmowania decyzji. W środowiskach korporacyjnych automatyzacja decyzji jest często realizowana przy wykorzystaniu techniki STP, gdzie spełnienie określonych warunków umożliwia systemowi samodzielne zakończenie całej procedury bez udziału człowieka.

Przykładowo, algorytm AI ocenia ryzyko przyznania rabatu sprzedażowego klientowi; jeśli wartość ta nie przekracza ustalonego progu, rabat stosowany jest automatycznie, a potwierdzenie przesyłane do klienta bez interwencji menedżera. Osoba opracowująca taki projekt powinna również przewidzieć mechanizmy uczenia się na błędach – w sytuacji nietrafnej decyzji automatycznej odpowiednie dane należy oznaczyć i wykorzystać podczas ponownego trenowania modelu. Dzięki temu możliwe jest poprawianie jakości predykcji oraz redukcja liczby pomyłek (Rys. 15).

[Tekst alternatywny: Diagram sekwencji przedstawia wąski proces automatyzacji decyzji rabatowej w Bonitasoft. Po zgłoszeniu przez Klienta w Portalu sprzedaży (UI) uruchamiany jest proces w Bonita Engine (BPM), który pobiera historię z CRM/ERP. W fazie „Ocena AI + Reguły” silnik wywołuje przez *Service Task* REST->AI zewnętrzny model AI, otrzymuje wynik (*score*) i przekazuje go do BRE (silnik reguł, DRL) w celu interpretacji polityki. W sekcji „Gateway decyzji” dla niskiego ryzyka system stosuje rabat automatycznie (STP) i odsyła potwierdzenie do UI; dla wyższego ryzyka tworzy zadanie dla Menedżera do ręcznej akceptacji, a decyzję zapisuje w ERP. Ostatnia sekcja „Uczenie na błędach” pokazuje pętlę zwrotną: przypadki uznane za błędne trafiają do Pętli uczenia (*retraining*), która dostarcza dane do ponownego trenowania modelu AI.]



Rys. 15. Automatyzacja decyzji z użyciem AI i BRE w iBML.

Źródło: opracowanie własne.



3.4. Przyszłość iBPM

Rozwój inteligentnego zarządzania procesami biznesowymi wiąże się z postrzeganiem iBPM jako koncepcji funkcjonującej w ekosystemach hybrydowych, które łączą różne technologie automatyzacyjne i analityczne w ramach jednego pakietu usług. Obserwuje się tendencję do integracji rozwiązań wykraczających poza klasyczne RPA, obejmujących uczenie maszynowe, analitykę predykcyjną czy mechanizmy integracyjne typu iPaaS. Platforma integracyjna jako usługa iPaaS (*Integration Platform as a Service*) – to chmurowa platforma do budowania, uruchamiania i zarządzania integracjami między aplikacjami, danymi oraz procesami. Dostarcza gotowe konektory, mapowanie danych, orkiestrację przepływów, obsługę zdarzeń/API, harmonogramy, monitorowanie i mechanizmy bezpieczeństwa - wszystko jako usługę.

Systemy iBPM będą wymagały współpracy z wieloma źródłami danych, od baz relacyjnych po strumienie z urządzeń IoT. Zdolności adaptacyjne tych systemów mogą obejmować modyfikowanie przebiegów procesów w odpowiedzi na aktualne warunki, prognozowanie skutków decyzji oraz symulację alternatywnych wariantów działań przed wdrożeniem. Prognozy będą oparte na analizie wzorców zachowań klientów, trendów rynkowych i wyników audytów jakości. Narzędzia iBPM mogą wspierać podejmowanie decyzji poprzez analizę dużych ilości danych. Dostawcy usług chmurowych udostępniają moduły analityki w czasie rzeczywistym powiązane z silnikami workflow.

Obecnie obserwuje się zwiększające się znaczenie środowisk umożliwiających edycję logiki procesowej w trybie niskokodowym lub bez kodu LCNC (*lowcode / nocode*). Platformy tworzenia aplikacji przy użyciu niewielkiej ilości kodu są wykorzystywane przez zespoły biznesowe do opracowywania przebiegu działania. W przyszłości narzędzia takie jak Bonitasoft mogą oferować możliwość interaktywnej modyfikacji procesu w czasie rzeczywistym, bez zatrzymywania instancji, poprzez zmianę reguł lub parametrów bramek zgodnie z aktualnym stanem KPI.

Kolejnym kierunkiem oczekiwanych zmian jest intensyfikacja wykorzystania semantycznego przetwarzania informacji jako narzędzia wspierającego procesy decyzyjne. System będzie miał możliwość analizy nie tylko danych liczbowych czy prostych warunków logicznych, lecz również rozpatrywania kontekstu tekstowego zgłoszeń klientów lub statusów prawnych transakcji, na przykład poprzez analizę sentymentu wiadomości e-mail.

Można również przewidzieć stopniowe zwiększanie udziału procesów autonomicznych, czyli takich, które po uruchomieniu samodzielnie wybierają działania uzupełniające bez udziału człowieka. Od strony projektowej oznacza to konieczność przygotowania obszernego zbioru reguł startowych oraz zestawu algorytmów adaptacyjnych, umożliwiających reakcję na wszelkie zmiany stanu zmiennych procesowych.



Na poziomie organizacyjnym prognozuje się, że iBPM będzie zmierzać w kierunku modelu holistycznego, w którym wszystkie funkcje kontrolne, wykonawcze i analityczne są zintegrowane w jednym panelu operacyjnym.

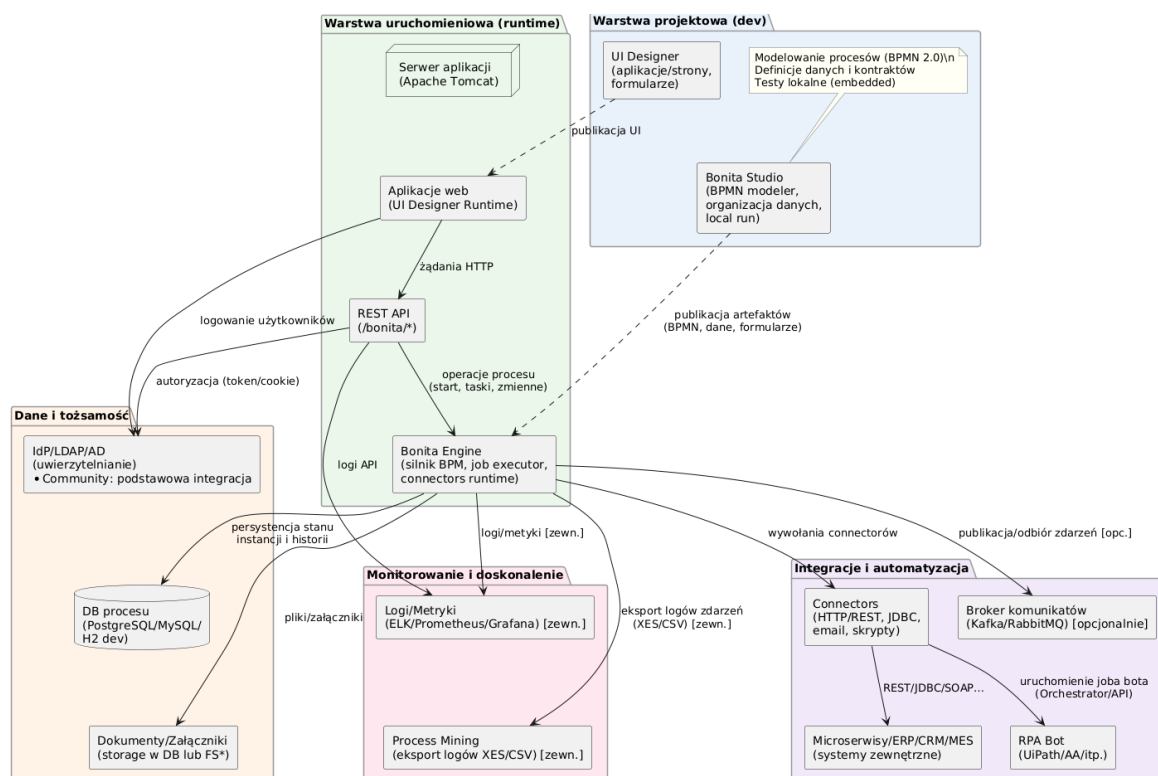
Przykładowo, Bonitasoft może rozwijać takie rozwiązania, oferując rozbudowane dashboardy czasu rzeczywistego obejmujące nie tylko status instancji procesu, ale również prognozy jego przebiegu generowane przez moduły AI oraz rekomendacje dotyczące optymalizacji dla menedżera projektu. Taka integracja umożliwi stworzenie środowiska projektowego pełniącego rolę centrum zarządzania działaniami operacyjnymi organizacji.

W nadchodzących latach kluczowe znaczenie będzie miało połączenie kompetencji w zakresie modelowania procesów z umiejętnościami integracji narzędzi analitycznych oraz ze znajomością wyzwań i potencjalnych zagrożeń wynikających z zastosowania tych technologii.

4. Platforma Bonitasoft Community Edition

4.1. Charakterystyka

Platforma Bonitasoft Community Edition (<https://www.bonitasoft.com>) to oprogramowanie typu open source służące do projektowania, uruchamiania oraz monitorowania procesów biznesowych zgodnych ze standardem BPMN 2.0. Zawiera narzędzia, takie jak Bonita Studio do edycji, silnik *workflow* oraz Bonita Portal, który zapewnia użytkownikom dostęp do zarządzania instancjami procesów i zadaniami. Architektura platformy bazuje na serwerze aplikacyjnym Tomcat, zawiera wbudowaną bazę danych oraz mechanizmy konfiguracji ról i uprawnień, co pozwala na organizację struktury wykonawczej od projektu do wdrożenia (Rys. 16). [Tekst alternatywny: Diagram pokazuje logiczną architekturę Bonitasoft CE w ujęciu warstwowym: warstwa projektowa, warstwa uruchomieniowa, warstwa danych i uwierzytelniania, warstwa monitorowania i doskonalenia procesów, warstwa integracji i automatyzacji procesów.]



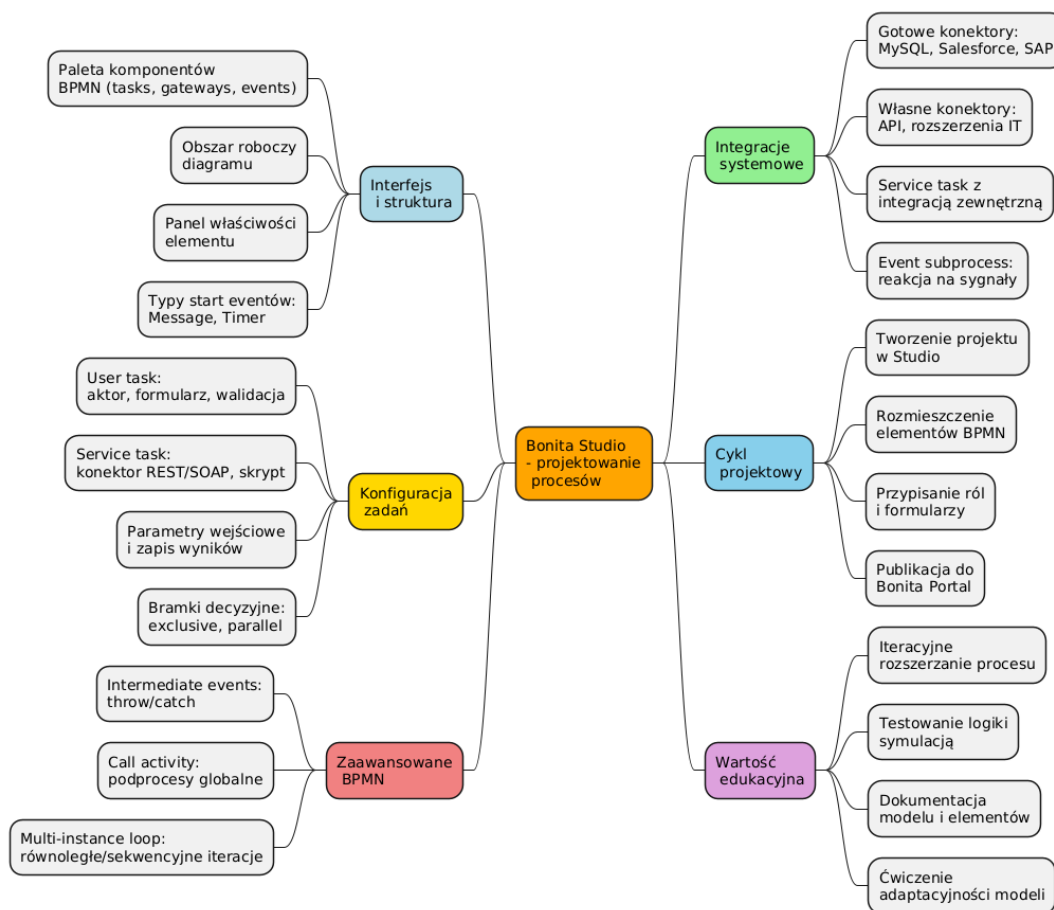
Rys. 16. Architektura Bonitasoft CE.

Źródło: opracowanie własne.

4.2. Projektowanie procesów

Bonita Studio to narzędzie projektowe platformy Bonitasoft CE, umożliwiające przekształcenie koncepcji procesu biznesowego w model możliwy do uruchomienia przez silnik workflow (Rys. 17).

[Tekst alternatywny: Mapa myśli „Bonita Studio – projektowanie procesów” streszcza kluczowe obszary pracy: interfejs (paleta BPMN, obszar diagramu, właściwości, start eventy), konfigurację zadań (*user/service task*, parametry I/O, bramki), elementy zaawansowane BPMN, oraz integracje systemowe (gotowe i własne konektory. Uzupełnia je cykl projektowy: tworzenie i publikacja projektu, przypisanie ról i formularzy, iteracyjne rozszerzanie i testy, wraz z wartością edukacyjną w postaci dokumentacji i ćwiczenia adaptacyjności modeli.



Rys. 17. Projektowanie procesu w Bonitasoft Studio.
Źródło: opracowanie własne.

Interfejs edytora umożliwia projektowanie diagramów BPMN, konfigurowanie elementów procesowych oraz definiowanie ich powiązań z warstwą użytkownika



i systemami zewnętrznymi. Realizacja przykładowej aplikacji procesowej obejmuje szereg powtarzalnych etapów: rozmieszczenie elementów BPMN zgodnie z wymaganiami biznesowymi, przypisanie ról uczestnikom, przygotowanie formularzy i konektorów, ustawienie warunków w bramkach decyzyjnych, dodanie zdarzeń końcowych zamykających przebieg procesu, a także zapis i publikację modelu do Bonita Portal. Moment publikacji stanowi przejście od fazy projektowania do wykonania — silnik workflow interpretuje diagram i zarządza przebiegiem działań według zadanej konfiguracji.

Kluczowym aspektem funkcjonalności Bonita Studio jest integracja modułów. Platforma oferuje gotowe konektory do wielu popularnych systemów oraz pozwala na tworzenie własnych rozwiązań poprzez API lub narzędzia deweloperskie. Integracje te są implementowane bezpośrednio w procesie jako część zadań serwisowych lub podprocesów obsługujących globalne zdarzenia.

Bonita Studio wspiera iteracyjny charakter cyklu BPM, umożliwiając łatwe modyfikowanie modeli oraz publikowanie kolejnych wersji bez konieczności inicjowania projektu od podstaw. Możliwość utrzymywania równoległych wersji procesu ułatwia testowanie zmian i planowanie migracji aktywnych instancji.

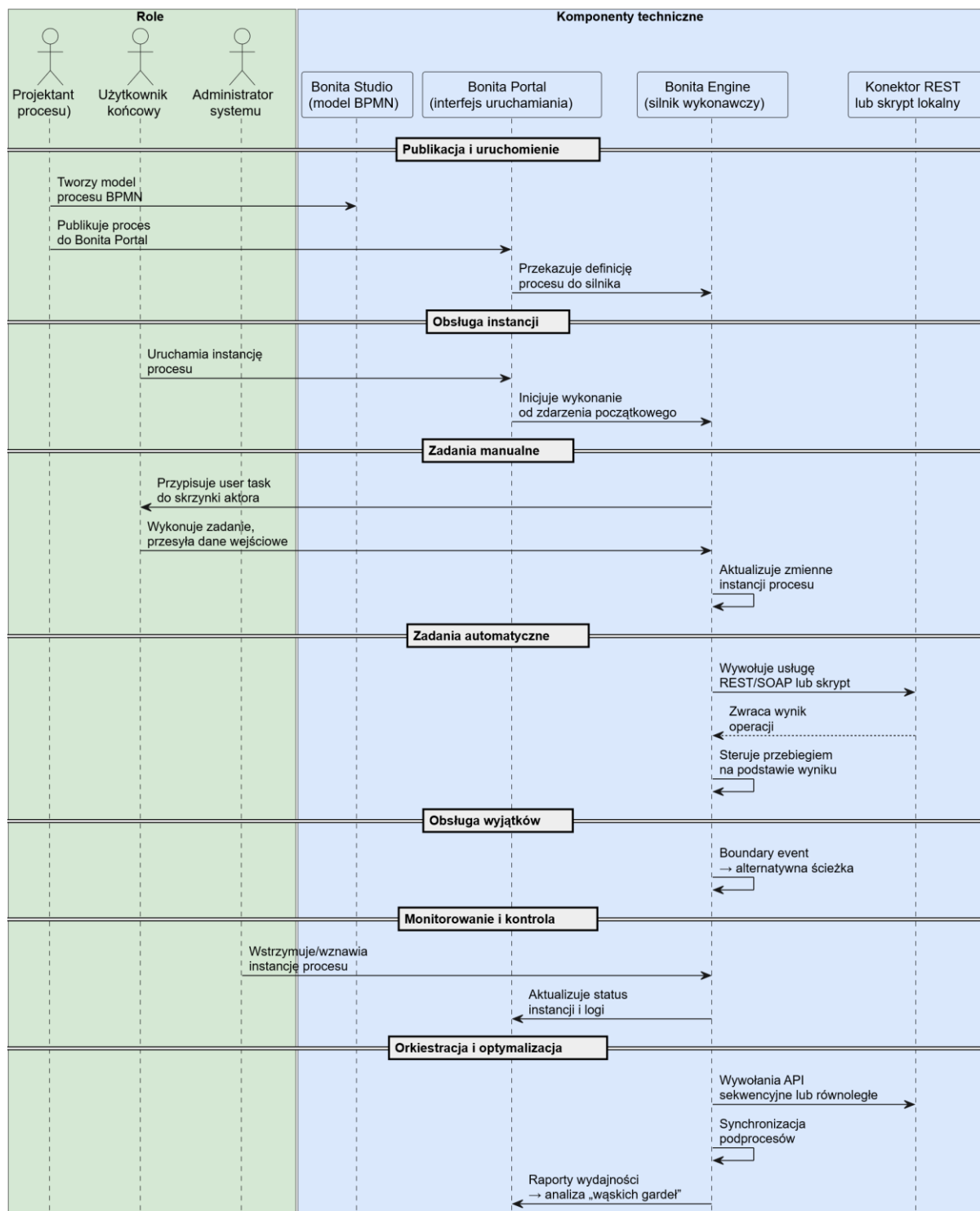
Narzędzie wspomaga również dokumentowanie modeli – każdy element można opisać, podając szczegółowe informacje dotyczące jego funkcji czy sposobu działania. Taka dokumentacja uznawana jest za dobrą praktykę wdrożeniową, usprawniając utrzymanie aplikacji i szkolenie nowych członków zespołu projektowego. Dzięki tym funkcjonalnościom Bonita Studio stanowi kompleksowe środowisko do budowy aplikacji procesowych od koncepcji po wdrożenie.

4.3. Realizacja procesów

Bonita Engine stanowi kluczowy komponent wykonawczy platformy, odpowiedzialny za inicjowanie oraz kontrolę instancji procesów opracowanych w Bonita Studio.

Pełni funkcję silnika workflow, który interpretuje modele BPMN na poziomie implementacyjnym, zarządza przepływem tokenów pomiędzy elementami diagramu, alokuje zadania uczestnikom oraz koordynuje realizację usług systemowych (Rys. 18).

[Tekst alternatywny: Diagram sekwencji opisuje wykonanie procesu w BonitaSoft z uwzględnieniem ról, komponentów technicznych w układzie pionowym oraz poszczególne interakcje pogrupowane w poziomie: Publikacja i uruchomienie, obsługa instancji procesu, zadania manualne, zadania automatyczne, obsługa wyjątków, monitorowanie i kontrola, orkiestracja i optymalizacja.]



Rys. 18. Wykonanie procesu w Bonitasoft.
Źródło: opracowanie własne.



Po stworzeniu modelu procesu w Bonita Studio warto:

- upewnić się, że każdy element wykonawczy ma przypisanych aktorów (dla zadań manualnych) lub poprawnie skonfigurowane konektory (dla automatów).
- ustawić warunki przejścia dla bramek tak, aby korzystały one ze zmiennych aktualizowanych przez wyniki działań silnika.
- Skonfigurować obsługę sytuacji wyjątkowych, np. przez zdarzenia brzegowe reagujące na błędy w wywołaniach usług.

Po opublikowaniu odpowiednio przygotowanego modelu w Bonita Portal możliwe jest uruchomienie instancji oraz monitorowanie jej przebiegu. W trakcie realizacji procesu istnieje także możliwość modyfikacji stanu instancji - administrator ma uprawnienia do jej wstrzymania, wznowienia bądź anulowania.

Ścisła integracja pracy silnika z mechanizmem ról gwarantuje ochronę danych - dostęp do wybranych aktywności oraz zmiennych procesowych mają wyłącznie osoby upoważnione. Istnieje możliwość zastosowania zaawansowanych reguł kontrolnych, takich jak dwuetapowa akceptacja decyzji czy rozdzielenie obowiązków pomiędzy różne grupy aktorów.

Z perspektywy technicznej kluczowe znaczenie ma również monitoring wydajności silnika: szczegółowa analiza raportów dotyczących zużycia pamięci oraz czasu realizacji poszczególnych aktywności umożliwia identyfikację potencjalnych „wąskich gardeł” oraz efektywną optymalizację projektu procesu.

5. Cykl budowy aplikacji procesowej

Cykl budowy aplikacji procesu w Bonitasoft składa się z następujących etapów:

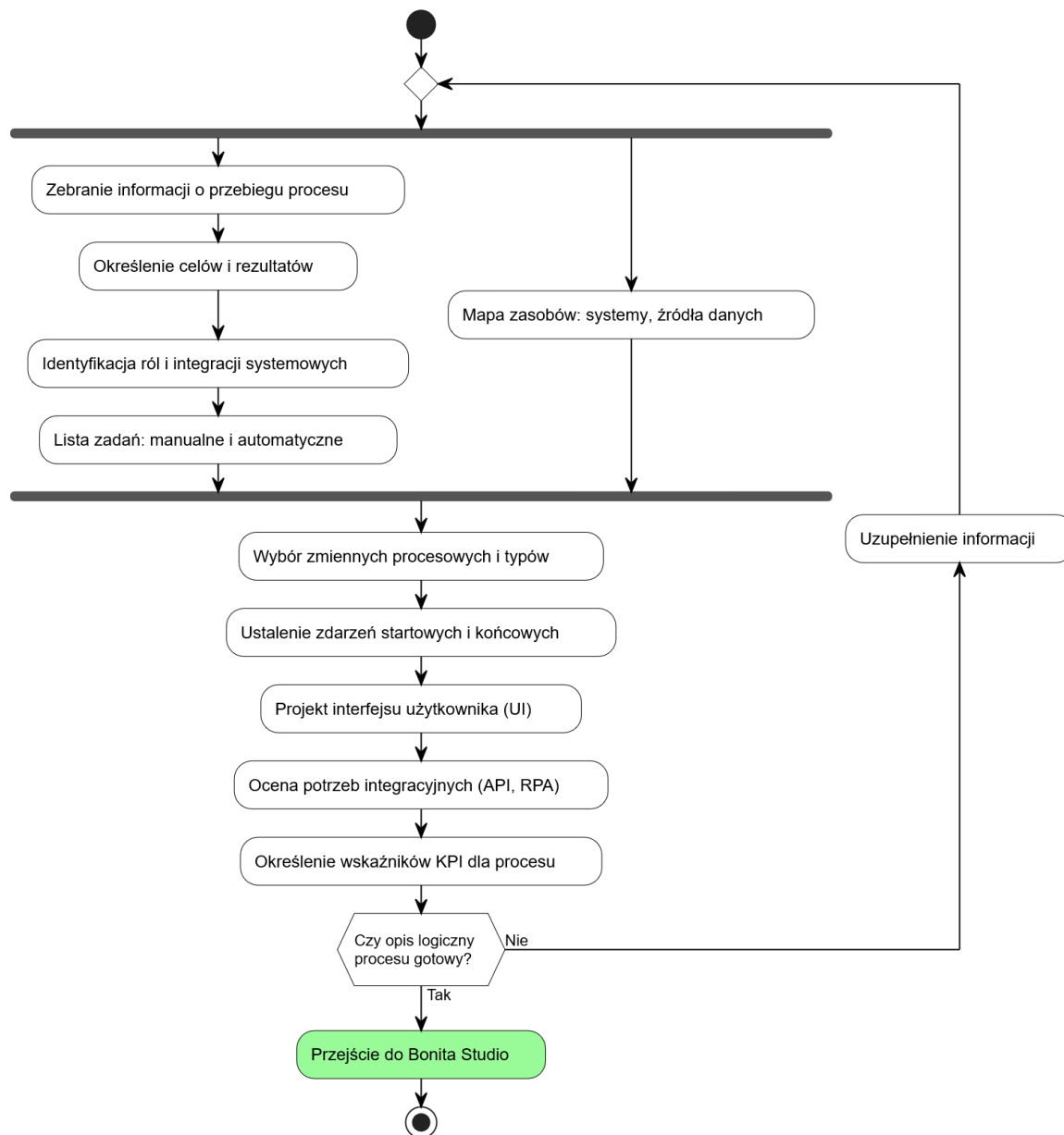
1. Przygotowanie projektu.
2. Tworzenie diagramu procesu.
3. Definiowanie ról i uprawnień.
4. Testowanie aplikacji procesowej.
5. Publikacja i uruchomienie procesu.

5.1. Przygotowanie projektu

W fazie przygotowania projektu aplikacji procesowej w Bonitasoft kluczowe jest przełożenie założeń biznesowych na pierwszy szkic techniczny, który będzie punktem startowym pracy w Bonita Studio. Przygotowanie projektu przebiega w sposób przedstawiony nad diagramie na

Rys. 19.

[Tekst alternatywny: diagram aktywności pokazujący sekwencję czynności do wykonania na etapie przygotowania projektu aplikacji procesowej w celu sformułowania opisu logiki procesu i jego powiązań ze źródłami danych.]



Rys. 19. Przygotowanie projektu aplikacji procesowej .
Źródło: opracowanie własne.

- Zbieranie informacji o procesie**

Projektant powinien zebrać kluczowe dane o procesie, jego celach i oczekiwanych efektach. Na tym etapie wystarczy ustalić role, integracje



systemowe oraz główne działania i punkty styku; szczegóły nie są jeszcze wymagane.

- **Identyfikacja i klasyfikacja zadań**

Należy wypisać wszystkie zadania procesu (manualne i automatyczne), określić ich typ oraz przypisać każdemu zadaniu odpowiednią rolę wykonawcy według logiki biznesowej.

- **Opracowanie mapy zasobów**

Warto stworzyć mapę zasobów z wykazem systemów i źródeł danych potrzebnych na każdym etapie. Jeśli proces wymaga zapisania zamówienia w ERP, należy określić wymagania dotyczące połączenia, rekordu i dostępnych konektorów Bonitasoft.

- **Wybór zmiennych procesowych**

Przed rozpoczęciem projektu należy określić zmienne procesowe, ich typy oraz nazwę odpowiadającą roli i zawartości danych. W Bonita Studio najlepiej przygotować listę zmiennych wraz z typami (tekst, liczba, data itp.) i wartościami domyślnymi przed modelowaniem BPMN, co ułatwia późniejszą wymianę informacji między zadaniami, bramkami i formularzami.

- **Definiowanie punktów początkowych i końcowych**

W projekcie należy jasno wyznaczyć zdarzenia początkowe i końcowe każdej ścieżki działania, np. uruchomienie przez wiadomość od partnera lub operatora. Punkty startowe będą konfigurowane później w Bonita Studio, ale już teraz trzeba zaznaczyć je na mapie przebiegu jako potencjalne wyzwalacze (*trigger*).

- **Szkic interfejsu użytkownika**

Projektowanie interfejsu użytkownika dla zadań manualnych polega na określeniu pól wejściowych i reguł walidacyjnych, które są potrzebne przed przekazaniem danych dalej. Pozwala to połączyć formularze zaprojektowane w UI Designer z zadaniami typu *user task* podczas wdrażania.

- **Analiza potrzeb integracyjnych**

Na tym etapie należy ocenić potrzeby dotyczące integracji z usługami zewnętrznymi, na przykład poprzez wykorzystanie dostępnych konektorów lub opracowanie własnego rozszerzenia API. W przypadku, gdy część procesu zostaje obsługiwana przez narzędzie RPA, powinny zostać określone szczegóły scenariusza pracy bota, takie jak dane pobierane z Bonita Engine, miejsce ich umieszczenia oraz sposób przekazania wyniku operacji do instancji procesu.

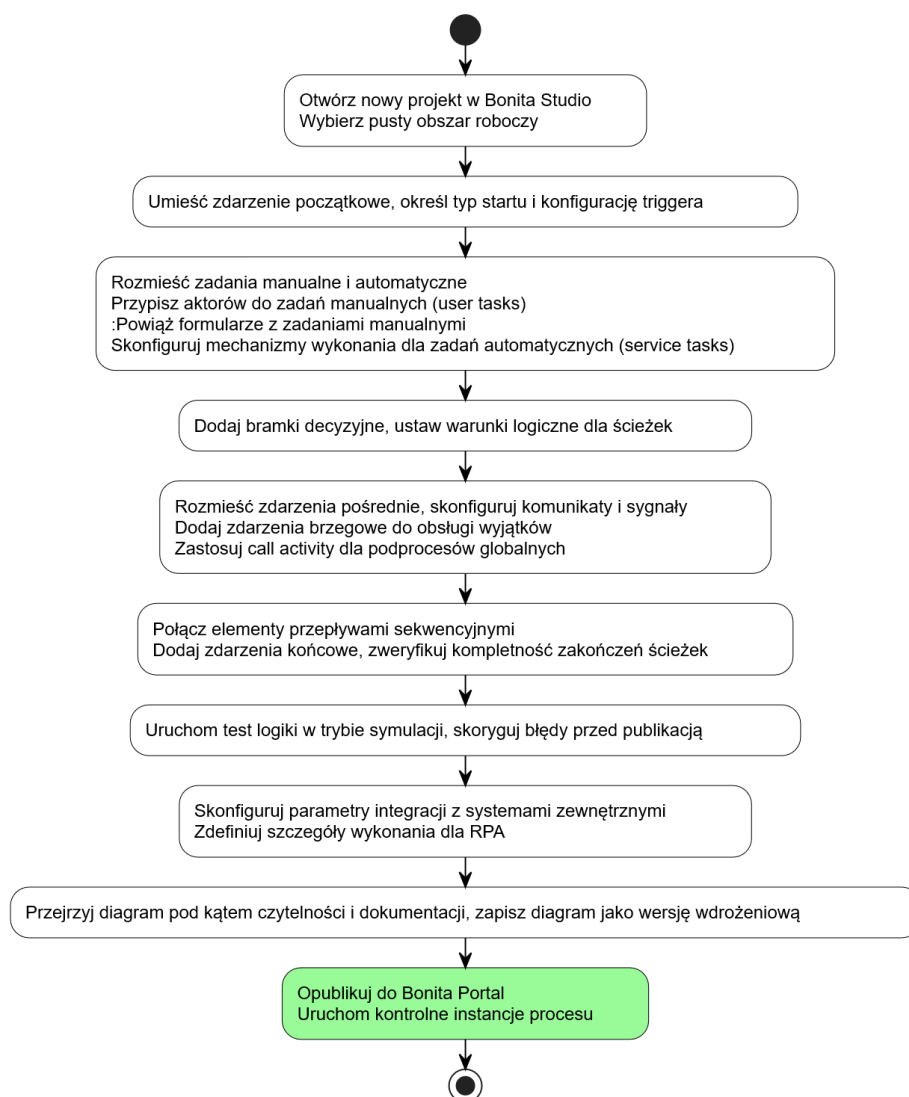
- **Ustalenie wskaźników efektywności procesu**

Dla zachowania spójności projektu w kontekście przyszłego monitorowania efektów, warto już na etapie przygotowania określić kluczowe wskaźniki KPI powiązane z procesem. Przykładowo: czas trwania całego przebiegu procesu od *start event* do *end event*, liczba wyjątków obsługiwanych przez ścieżki alternatywne czy średni czas reakcji uczestnika na przydzielone zadanie.

5.2. Tworzenie diagramu procesu

Po skompletowaniu opisu logicznego procesu następuje przejście do fazy budowy modelu procesu w Bonita Studio. Tworzenie diagramu procesu jest momentem przełożenia koncepcji powstałej w fazie przygotowania projektu na model, który zostanie później uruchomiony przez silnik *workflow* (Rys. 20).

[Tekst alternatywny: diagram aktywności pokazujący sekwencję czynności do wykonania na etapie tworzenia diagramu procesu w Notacji BPMN 2.0.]



Rys. 20. Tworzenie diagramu procesu w Bonita Studio.

Źródło: opracowanie własne.



Pierwszym krokiem podczas modelowania procesu w Bonita Studio jest dodanie odpowiedniego zdarzenia początkowego, które odpowiada za inicjację przebiegu. Może nim być start wywołany komunikatem z innego systemu, sygnałem czasowym lub akcją użytkownika, który uruchamia instancję ręcznie.

Kolejnym etapem jest rozmieszczenie aktywności zgodnie z przebiegiem procesu. Przy zadaniach manualnych (*user tasks*) przypisz aktorów w zakładce „Actors”, mapując ich na realnych użytkowników lub grupy w Bonita Portal. Każde zadanie powinno być powiązane z formularzem z UI Designer. Zadania automatyczne (*service tasks*) wymagają wyboru mechanizmu wykonania - konektora REST/SOAP, operacji bazodanowej lub skryptu - oraz ustalenia parametrów wejściowych i sposobu zapisu wyników.

Po dodaniu głównych czynności do diagramu należy ustawić bramki decyzyjne, które rozdzielają lub łączą ścieżki procesu. Warunki logiczne, bazujące na danych wejściowych lub wynikach działań, definiuje się w panelu właściwości bramki. Niewłaściwa konfiguracja może zatrzymać token lub zakłócić przebieg procesu.

Równolegle do głównych zadań rozmieszcza się zdarzenia pośrednie (*intermediate events*), które służą do obsługi komunikatów lub sygnałów wewnątrz procesu. Każde takie zdarzenie należy odpowiednio skonfigurować, określając źródło lub odbiorcę oraz typ wiadomości. W przypadku konieczności obsługi wyjątków podczas realizacji zadania dodaje się zdarzenia brzegowe (*boundary events*) na obrysie aktywności. Ich typ (przerywający lub nieprzerywający) determinuje dalszy wpływ na przebieg procesu.

Wykorzystanie *call activity* do podprocesów globalnych umożliwia wskazanie modelu i mapowanie zmiennych w panelu właściwości. Dzięki temu główny diagram jest krótszy, a logika powtarzanych funkcji biznesowych spójna.

Po rozmieszczeniu i połączeniu wszystkich elementów przepływami sekwencyjnymi dodaje się zdarzenia końcowe (*end events*), które domykają każdą ścieżkę procesu. Ich obecność jest kluczowa dla monitorowania czasu trwania instancji oraz kontroli prawidłowego zakończenia procesu. Warto zweryfikować, czy każdy możliwy wariant przebiegu kończy się przypisanym punktem końcowym.

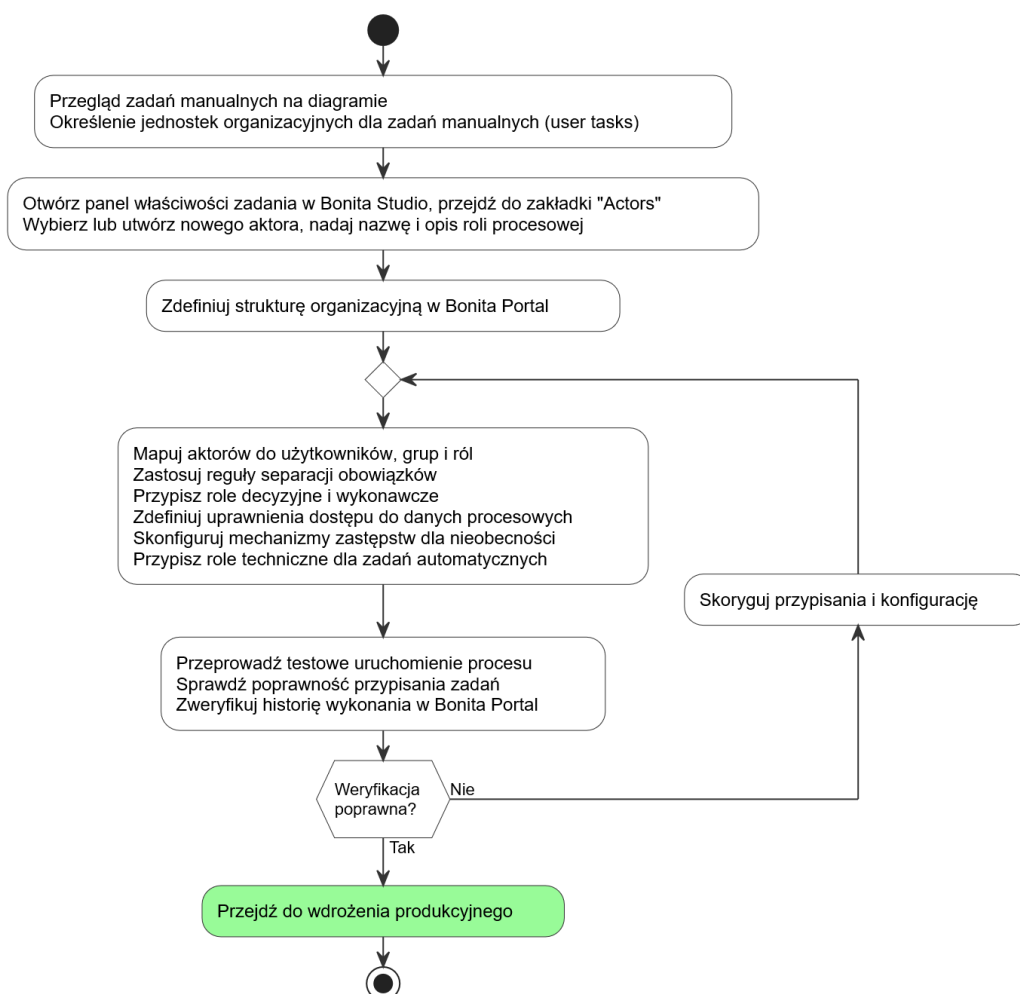
W przypadku integracji z systemami zewnętrznymi należy pamiętać o ustawieniu parametrów połączeń i testowaniu odpowiedzi usług jeszcze przed publikacją całego procesu. Jeśli część integracji realizowana jest przez rozwiązania RPA uruchamiane wewnątrz zadania automatycznego (*service task*), projektant powinien wpisać szczegóły wykonania: skrypt bota, przekazywane wartości oraz sposób odbioru wyniku przez Bonita Engine.

Przed zatwierdzeniem diagramu warto sprawdzić jego czytelność, stosować spójne nazwy ról i opisy w polach „Documentation”. Po przygotowaniu diagram można zapisać jako wersję wdrożeniową, opublikować w Bonita Portal i przeprowadzić testowe uruchomienia, monitorując działanie procesu.

5.3. Definiowanie ról i uprawnień

Po utworzeniu diagramu i rozmieszczeniu elementów BPMN należy określić, kto i w jakim zakresie będzie mógł uczestniczyć w realizacji poszczególnych zadań (Rys. 21).

[Diagram przedstawia proces definiowania ról i uprawnień w Bonita Studio: od identyfikacji zadań manualnych i przypisywania ról aktorom, przez stworzenie struktury organizacyjnej i mapowanie użytkowników, aż po testowanie, korekty i wdrożenie produkcyjne.]



Rys. 21. Definiowanie ról i uprawnień w Bonita Studio.

Źródło: opracowanie własne.

Każda aktywność powinna mieć przypisaną rolę, a ta rola musi być następnie powiązana z faktycznym kontem lub grupą użytkowników istniejącą w organizacji. Role procesowe ustala się przez przypisanie aktorów do manualnych zadań na

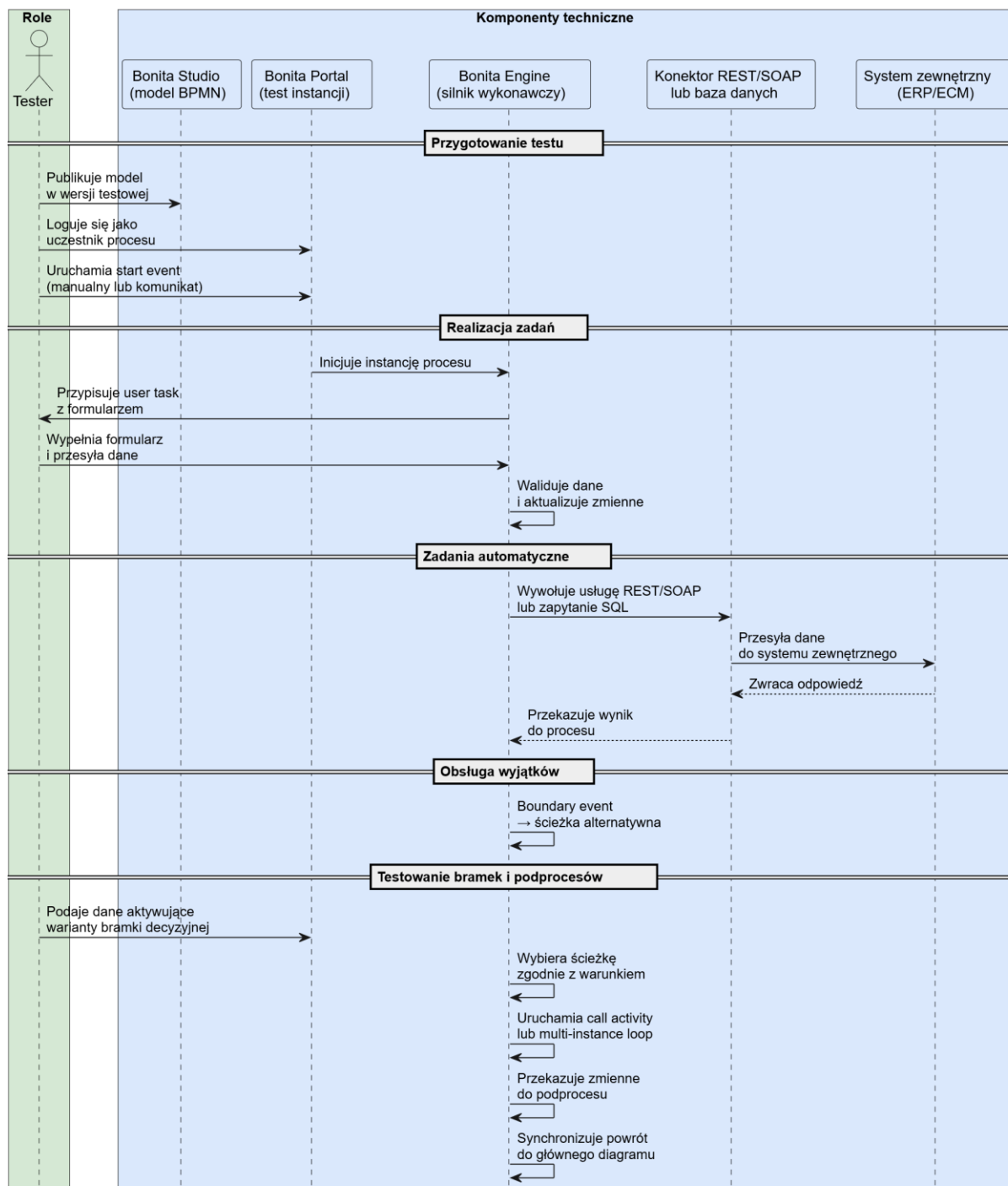


podstawie jednostek organizacyjnych lub stanowisk. Aktorzy są logicznymi reprezentantami ról odwzorowywanych w strukturze Bonita Portal i przypisywani użytkownikom, grupom (działom) lub rolam (funkcjom niezależnym od grupy). Wyróżnia się role decyzyjne i wykonawcze, stosując separację obowiązków dla bezpieczeństwa. Przydział aktorów możliwy jest poprzez plik organizacji lub portal, a strukturę można zaimportować. Testy procesu potwierdzają prawidłowość przypisań, a dostęp do danych ogranicza się do wybranych ról. System obsługuje zastępstwa podczas nieobecności i przydziela role techniczne, np. „System”, do automatycznych zadań.

5.4. Testowanie aplikacji procesowej

Celem tego etapu jest sprawdzenie, czy aplikacja procesowa uruchamiana w Bonita Portal zachowuje się zgodnie z założeniami biznesowymi i technicznymi ustalonymi wcześniej w fazie projektowania. Testowanie pozwala wykryć błędy logiki przepływu, nieprawidłową konfigurację konektorów, niespójności danych procesowych czy problemy z integracją systemów zewnętrznych (Rys. 22).

[Tekst alternatywny: Diagram sekwencji przedstawia scenariusz testów procesu w Bonitasoft z udziałem testera oraz komponentów: Bonita Studio, Bonita Portal, Bonita Engine, konektor REST/SOAP lub baza danych i system zewnętrzny. Pokazuje etapy: przygotowanie testu (publikacja modelu, start instancji), realizację zadań (wypełnianie formularzy, walidacja i aktualizacja zmiennych), wykonanie zadań automatycznych z integracją z systemem zewnętrznym, obsługę wyjątków przez boundary event oraz testowanie bramek i podprocesów (wybór ścieżki, call activity, pętla multi-instance).]



Rys. 22. Testowanie aplikacji procesowej.
Źródło: opracowanie własne.



Poniżej przedstawiono przykładowy plan testów aplikacji procesowej:

1. Planowanie przypadków

- Opracowanie listy przypadków obejmujących ścieżki główne, alternatywne i wyjątki (np. błąd API).
- Dla każdego przypadku określić należy warunki wejściowe: zmienne startu, dane formularzy oraz zewnętrzne sygnały lub wiadomości.
- Definiowanie oczekiwanych wyników: docelowe zdarzenie końcowe (*end event*), wartości zmiennych po ukończeniu zadania oraz treści odpowiedzi konektorów (Author, 2023a).

2. Uruchomienie testu

- Model jest publikowany w Bonita Portal (tryb testowy), a użytkownik loguje się w odpowiedniej roli.
- Proces jest inicjowany (przycisk *Start* lub *Message Start*); następuje realizacja zadań manualnych z weryfikacją walidatorów.

3. Weryfikacja automatyzacji i integracji

- Analiza realizacji zadań automatycznych (*service task*) i konektorów (REST/SOAP/SQL) na podstawie zapisów w logach wraz z czasami i błędami.
- Zdarzenia brzegowe (*Boundary events*) obsługują błędy – sprawdzenie, czy po zasymulowaniu awarii proces przechodzi na ścieżkę wyjątków.

4. Logika sterująca

- Dla bramek decyzyjnych podawane są dane aktywujące wszystkie gałęzie; wynik powinien odpowiadać zdefiniowanym warunkom, a rozbieżności wskazują na błędy konfiguracji.

5. Podprocesy i pętle

- W przypadku *call activity/global subprocess* potwierdzana jest poprawna wymiana zmiennych między procesem głównym a podprocesem.
- Sprawdzenie, czy dla *multi-instance* liczba iteracji i zakończenia odpowiadają parametrom definicji.

6. Testy integracyjne i wydajnościowe

- Współdziałanie systemów (np. ERP i ECM) przebiega zgodnie z kolejnością czynności i **synchronizacją** w bramkach równoległych.
- Czasy ścieżek są mierzone i odnoszone do KPI przyjętych w projekcie.

7. Obciążenie ról i monitoring

- Rozłożenie zadań na aktorów jest analizowane w raportach Portalu; ewentualna nierównowaga jest identyfikowana.
- Moduł monitoringu instancji prezentuje pozycję tokena i historię działań ze znacznikami czasu.

Końcowym etapem testów aplikacji procesowej jest sporządzenie raportu, który obejmuje listę przypadków, wyniki oraz uwagi (dotyczące m.in. bramek, konektorów,



ról i walidacji). Następnie w Bonita Studio wprowadza się poprawki, publikuje wersję poprawioną i powtarza kluczowe przypadki aż do uzyskania pełnej zgodności ze specyfikacją. W efekcie uzyskuje się stabilny, audytowalny proces odporny na typowe problemy integracyjne i organizacyjne.

5.5. Publikacja i uruchomienie procesu

Publikacja zaczyna się od zapisania aktualnej wersji procesu w Bonita Studio i nadania jej numeru wersji w polu „Version” w panelu właściwości głównych procesu. Dzięki temu możliwe jest późniejsze równoległe utrzymywanie kilku wersji tego samego schematu, co ułatwia aktualizacje i testy porównawcze.

Następnie wybiera się opcję „Deploy” lub „Publish” z menu Studio – mechanizm ten pakuje definicję procesu wraz z powiązаныmi formularzami, skryptami i zasobami do formatu wdrożeniowego czytelnego dla Bonita Engine. W trakcie tej operacji Studio łączy się z instancją Bonita Portal określoną w konfiguracji serwera. Jeśli wymagane są dane uwierzytelniające, trzeba zalogować się kontem administratora Portalu.

Po udanej publikacji nowa definicja procesu pojawia się na liście dostępnych procesów administracyjnych w module „Processes” w Portalu. Na tym etapie można zweryfikować poprawność wdrożenia – sprawdzić nazwę, opis, numer wersji oraz to, czy status jest ustawiony jako „Enabled”.

Konfiguracja przed uruchomieniem obejmuje mapowanie aktorów na konkretnych użytkowników lub grupy istniejące w środowisku Portalu. Nawet jeśli podczas projektowania nadaliśmy nazwę roli i przypisaliśmy jej funkcjonalność, dopiero teraz wskazujemy faktyczne konta lub grupy uprawnione do wykonywania danych zadań. W tym celu używamy opcji „Actor mapping”, wybierając aktora z listy i przypisując mu jeden lub kilka dostępnych podmiotów (np. grupa „Sprzedaż - Oddział Kielce”). Brak poprawnego mapowania spowoduje zatrzymanie instancji na pierwszym zadaniu oczekującym na wykonawcę. Aby uruchomić instancję procesu ręcznie (jeżeli start jest inicjowany przez użytkownika), należy wyszukać go w menu „Process list” i wybrać opcję „Start”.



Literatura

1. Van der Aalst, W.M.P. and Carmona, J. (eds.) (2022) Process Mining Handbook. Cham: Springer. DOI: <https://doi.org/10.1007/978-3-031-08848-3>.
2. Apache Software Foundation (n.d.) Apache Kafka Documentation [online]. <https://kafka.apache.org/documentation>.
3. Bonitasoft (2025).[online] <https://www.bonitasoft.com>.
4. Camunda (n.d.) External Task Client — User Guide [online]. <https://docs.camunda.org/manual/latest/user-guide/ext-client/>. Camunda (n.d.)
5. Camunda (2023) 'Microservice Orchestration vs Choreography — what's the difference?' Camunda Blog. <https://camunda.com/blog/2023/02/orchestration-vs-choreography>. https://camunda.com/blog/2023/02/orchestration-vs-choreography/?utm_source=chatgpt.com
6. Confluent (n.d.) Event-Driven Architecture (EDA): A Complete Introduction [online]. <https://www.confluent.io/learn/event-driven-architecture>.
7. Drejewicz, Sz. (2012), *Zrozumieć BPMN modelowanie procesów biznesowych*. Helion, Gliwice.
8. Enache, Maria. (2022). Technologies For Business Process Automation. Annals of Dunarea de Jos University of Galati. Fascicle I. Economics and Applied Informatics. 28. 21-26. DOI: <https://doi.org/10.35219/eai15840409263>.
9. Gawin, B. (2013) Symulacja procesów biznesowych: standardy BPMS i BPMN w praktyce. Helion, Gliwice.
10. IBM (2025) *What is Business Process Management?*, IBM. <https://www.ibm.com/think/topics/business-process-management>.
11. IBM (2025) *What is Robotic Process Automation (RPA)?*, IBM. <https://www.ibm.com/think/topics/rpa>.
12. J., B. (2024) *The Complete Guide To Business Process Management (BPM) Lifecycle*, Datarundown. <https://datarundown.com/bpm-lifecycle/>
13. Kossak, F. et al. (2015) *A Rigorous Semantics for BPMN 2.0 Process Diagrams*, A Rigorous Semantics for BPMN 2.0 Process Diagrams. Cham: Springer International Publishing. <https://doi.org/10.1007/978-3-319-09931-6>.
14. McClintock, M. (2023) *Robotic Process Automation (RPA) vs Business Process Management (BPM)*, ProcessMaker. <https://www.processmaker.com/blog/difference-between-rpa-bpm>.
15. Misiak, Z. (2024) Modelowanie procesów biznesowych : BPMN 2.0 od podstaw. Helion, Gliwice.
16. Piotrowski M. (2014) *Procesy biznesowe w praktyce. Projektowanie*,





testowanie i optymalizacja. Helion, Gliwice.

17. Red Hat (2015) *What's business automation?* <https://www.redhat.com/en/topics/automation/whats-business-automation>.
18. Schlegel, D., Rosenberg, B., Fundanovic, O., & Kraus, P. (2024). How to conduct successful business process automation projects? An analysis of key factors in the context of robotic process automation. *Business Process Management Journal*, 30(8), 99–119. <https://doi.org/10.1108/BPMJ-06-2023-0465>
19. Tucci, L. (2023) *What is Business Process Management? A Guide to BPM*, Search CIO. TechTarget. <https://www.techtarget.com/searchcio/definition/business-process-management>.
20. Vu, H., Leopold, H. and van der Aa, H. (2023) "What is Business Process Automation Anyway?," *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2023-January, pp. 5462–5471. <https://doi.org/10.24251/hicss.2023.666>.





Spis rysunków

Rys. 1. Kluczowe procesy w organizacji.	5
Rys. 2. Cykl zarządzania procesami biznesowymi.	7
Rys. 3. Rozwój zarządzania procesami biznesowymi.	10
Rys. 4. Role i odpowiedzialności w BPM.	12
Rys. 5. Role, odpowiedzialności i komunikacja w BPA.	14
Rys. 6. Działanie silnika <i>workflow</i> w Bonitasoft.	20
Rys. 7. Stany instancji procesu i przejścia (manualne vs automatyczne).	22
Rys. 8. Diagram sekwencji dla automatyzacji procesu aktualizacji kursów walut.	25
Rys. 9. Mapa korzyści płynących z automatyzacji procesów.	27
Rys. 10. Cykl ciągłego doskonalenia procesu PDCA.	28
Rys. 11. Diagram sekwencji dla przykładowej implementacji iBPM.	30
Rys. 12. Mapa ekosystemu iBPM.	32
Rys. 13. Diagram sekwencji dla przykładowej implementacji iBML.	34
Rys. 14. Automatyczne podejmowanie decyzji w iBML.	36
Rys. 15. Automatyzacja decyzji z użyciem AI i BRE w iBML.	38
Rys. 16. Architektura Bonitasoft CE.	41
Rys. 17. Projektowanie procesu w Bonitasoft Studio.	42
Rys. 18. Wykonanie procesu w Bonitasoft.	44
Rys. 19. Przygotowanie projektu aplikacji procesowej	46
Rys. 20. Tworzenie diagramu procesu w Bonita Studio.	48
Rys. 21. Definiowanie ról i uprawnień w Bonita Studio.	50
Rys. 22. Testowanie aplikacji procesowej.	52

Spis tabel

Tabela 1. Model warstwowy architektury automatyzacji procesów BPA.	17
---	----