|  |  |
| --- | --- |
| Politechnika Warszawska  Wydział Elektroniki i Technik Informacyjnych  Instytut Informatyki | Rok akademicki 2010/2011 |

Praca Dyplomowa Inżynierska

Adrian Wiśniewski

System Service Desk zgodny z zaleceniami ITIL:

Zarządzanie komponentami usług i konfiguracją oraz zarządzanie zmianą

Opiekun pracy:

dr inż. Michał Rudowski

|  |  |
| --- | --- |
| Ocena |  |
| Podpis przewodniczącego  Komisji Egzaminu Dyplomowego |  |

|  |  |
| --- | --- |
|  | Kierunek: Informatyka  Specjalność: Inżynieria Systemów Informatycznych  Data urodzenia: 24 grudnia 1987 r.  Data rozpoczęcia studiów: październik 2006 r. |

Życiorys

Urodziłem się 24 grudnia 1987 r. w Nowym Dworze Mazowieckim. Mieszkam w Wołominie, gdzie ukończyłem Sportową Szkołę Podstawową nr 5 im. C. K. Norwida i Sportowe Gimnazjum nr 3 im. Polskich Olimpijczyków. Następnie uczęszczałem do klasy o profilu matematyczno-fizycznym w IV Liceum Ogólnokształcącym im. Adama Mickiewicza w Warszawie. W październiku 2006 r. rozpocząłem studia na Politechnice Warszawskiej na Wydziale Elektroniki i Technik Informacyjnych na kierunku Informatyka.

Pierwszą pracę jako programista aplikacji internetowych podjąłem rok później —w 2007 r. — w firmie Psiloc Sp. z o. o., gdzie rozwijałem aplikację www.locatik.pl. Następnie przez kilka lat pracowałem na zlecenie przy wielu różnych projektach dla klientów takich jak McDonald’s, Allegro, czy gazeta.pl. Obecnie jestem zatrudniony jako programista gier komputerowych w firmie 11 bit studios S.A.

Ponadto od 2006 roku należę do społeczności twórców gier komputerowych www.gamedev.pl, a w latach 2009 – 2010 należałem do Koła Naukowego Twórców Gier Komputerowych „Polygon”, gdzie pełniłem rolę administratora i członka zarządu koła.

|  |  |
| --- | --- |
|  | Podpis studenta |

Egzamin dyplomowy

Złożył egzamin dyplomowy dn. 2011 r.

Z wynikiem

Ogólny wynik studiów

Dodatkowe wnioski i uwagi Komisji

Streszczenie

Tutaj streszczenie

Abstract

Tutaj streszczenie ang

Spis treści

[Rozdział 1. Wstęp 9](#_Toc299356053)

[1.1. Cel pracy 10](#_Toc299356054)

[1.2. Motywacja 10](#_Toc299356055)

[1.3. Zakres 11](#_Toc299356056)

[1.4. Układ 11](#_Toc299356057)

[1.5. Nazewnictwo 11](#_Toc299356058)

[Rozdział 2. ITIL 13](#_Toc299356059)

[2.1. Pojęcia podstawowe 14](#_Toc299356060)

[2.1.1. Usługa 14](#_Toc299356061)

[2.1.2. Zarządzanie usługami IT (ITSM) 14](#_Toc299356062)

[2.1.3. Funkcja 14](#_Toc299356063)

[2.1.4. Proces 14](#_Toc299356064)

[2.2. Historia 15](#_Toc299356065)

[2.2.1. Wersja pierwsza 15](#_Toc299356066)

[2.2.2. Wersja druga 16](#_Toc299356067)

[2.2.3. Wersja trzecia 18](#_Toc299356068)

[2.3. Struktura 19](#_Toc299356069)

[2.3.1. Strategia usług (Service Strategy) 21](#_Toc299356070)

[2.3.2. Projektowanie usług (Service Design) 22](#_Toc299356071)

[2.3.3. Przekazanie usług (Service Transition) 23](#_Toc299356072)

[2.3.4. Eksploatacja usług (Service Operation) 24](#_Toc299356073)

[2.3.5. Ustawiczne doskonalenie usług (Continual Service Improvement) 26](#_Toc299356074)

[2.4. Wyszczególnione procesy 27](#_Toc299356075)

[2.4.1. Zarządzanie komponentami usług i konfiguracją 27](#_Toc299356076)

[2.4.2. Zarządzanie zmianą 31](#_Toc299356077)

[2.5. Inne metodyki 33](#_Toc299356078)

[Rozdział 3. Firma RedHost 35](#_Toc299356079)

[3.1. Charakterystyka 35](#_Toc299356080)

[3.1.1. Świadczone usługi 35](#_Toc299356081)

[3.2. Wprowadzenie zarządzania konfiguracją 35](#_Toc299356082)

[3.2.1. Przykładowe elementy konfiguracji 35](#_Toc299356083)

[3.3. Wprowadzenie zarządzania zmianą 35](#_Toc299356084)

[3.3.1. Przykładowe żądania zmian 35](#_Toc299356085)

[Rozdział 4. Wymagania 37](#_Toc299356086)

[4.1. Technologia (TECH) 37](#_Toc299356087)

[4.2. Architektura (ARCH) 38](#_Toc299356088)

[4.3. Moduł zarządzania konfiguracją (KONF) 38](#_Toc299356089)

[4.4. Moduł zarządzania zmianą (ZMIA) 41](#_Toc299356090)

[4.5. Rdzeń systemu (CORE) 43](#_Toc299356091)

[Rozdział 5. Projekt 45](#_Toc299356092)

[Rozdział 6. Implementacja 47](#_Toc299356093)

[Rozdział 7. Użytkowanie systemu 49](#_Toc299356094)

[Rozdział 8. Testy i ocena wydajności 51](#_Toc299356095)

[Rozdział 9. Podsumowanie 53](#_Toc299356096)

[Bibliografia 55](#_Toc299356097)

[Dodatek A. Zawartość płyty CD 57](#_Toc299356098)

[Wykaz skrótów 59](#_Toc299356099)

[Spis Tabel 61](#_Toc299356100)

[Spis ilustracji 63](#_Toc299356101)

# Wstęp

W dobie społeczeństwa informacyjnego komputery stały się nieodłącznym elementem naszego życia. Są dziś atrybutem nowoczesności i motorem napędowym postępu, a ich ogromny wpływ na zmiany w gospodarce jest niekwestionowany. Przedsiębiorcy od największych korporacji do małych firm korzystają z informatyki, która jest już nierozerwalnym elementem biznesu. Systemów tworzonych na potrzeby przedsiębiorczości przybywa każdego dnia i są one coraz bardziej skomplikowane biorąc pod uwagę zarówno złożoność rozwiązywanych problemów, ilość wykorzystywanych zasobów, jak i integrację z innymi systemami.

Dynamiczny rozwój infrastruktury informatycznej przedsiębiorstw stawia nowe wyzwania w dziedzinie zarządzania. Dotychczas zadania działu IT skupiały się na technologii i dostarczaniu pojedynczych elementów takich jak aplikacja, serwer, sieć, czy baza danych. Postępowanie takie skutkuje rozproszeniem infrastruktury w przedsiębiorstwie, co z kolei znacznie utrudnia zarządzanie. Brak jednolitych procedur uniemożliwia efektywne przeprowadzanie audytów oraz skutkuje obniżeniem jakości świadczonych usług, co kończy się awariami. Ponadto zasoby systemów są od siebie odizolowane i niemożliwy jest ich transfer, a w konsekwencji dopasowanie do zmiennego w czasie zapotrzebowania użytkowników. Systemy przez większość czasu nie korzystają z pełni dostępnej mocy, co jest marnotrawstwem i może być postrzegane jako strata dla przedsiębiorstwa.

Model ITSM (ang. IT service management) podchodzi do problemu w zupełnie odwrotny sposób. Traktuje on infrastrukturę IT jako całość i skupia się na postrzeganiu jej z perspektywy użytkownika. Nie jest ważne jak i z czego zbudowane są poszczególne systemy, natomiast kluczowe znaczenie ma fakt, że świadczą one usługi, istotne dla osiągnięcia celów biznesowych przedsiębiorstwa. Usługą IT nazywamy zbiór sprzętu komputerowego, oprogramowania, łącz oraz innych elementów infrastruktury; które razem zapewniają spójną funkcjonalność pozwalającą uzyskać zamierzony efekt. Warto zwrócić uwagę, że definicja ta ma charakter uniwersalny. Pozwala to na grupowanie typowych czynności związanych z zarządzaniem usługami IT w procesy, które dzięki swojej ogólności mogą być stosowane niezależnie od specyfiki przedsiębiorstwa. Ponadto funkcjonowanie działu IT w roli dostawcy usług pozwala skupić się bezpośrednio na celach biznesowych, ponoszonych kosztach oraz wymaganej jakości; podczas gdy kwestie techniczne są rozstrzygane wewnątrz departamentu informatycznego przez odpowiednio wykwalifikowany personel.

ITSM jedynie definiuje sposób w jaki należy postrzegać infrastrukturę informatyczną. Nie jest gotową receptą na problemy zarządzania IT, lecz jako nowatorskie podejście stanowi punkt wyjścia dla wielu konkretnych metodyk. Bez wątpienia obecnie najbardziej popularną i najszerzej stosowaną z nich jest ITIL (ang. Information Technology Infrastructure Library). Jest to kodeks postępowania opracowany przez OGC[[1]](#footnote-1), znane z wprowadzenia innych modeli zarządzania takich jak PRINCE2, MSP, czy M\_o\_R. ITIL składa się ze zbioru najlepszych praktyk, wypróbowanych w wielu przedsiębiorstwach, co zarazem stanowi jego główną zaletę. Nie narzuca sztywnych wytycznych, przez co może być przystosowany do różnych potrzeb. Ponadto — jako metodyka wytworzona przez organizacje rządowe — nie jest własnością żadnej firmy, dzięki czemu jest niezależna od platformy technologicznej i specyfiki branży.

## Cel pracy

Stworzenie prototypu systemu do zarządzania infrastrukturą informatyczną przedsiębiorstwa, opartego o ITIL w wersji 3. Ponadto zobrazowanie zastosowań tego systemu na podstawie przypadków użycia w przykładowej firmie.

## Motywacja

Na rynku istnieją gotowe, skomplikowane i duże produkty do zarządzania usługami IT oparte o ITIL, a ogromne koszty licencji sprawiają, że są one dostępne jedynie dla największych firm. Dodatkowo często występującym problemem jest brak możliwości adaptacji systemu do zmieniającej się charakterystyki przedsiębiorstwa. Jest to spowodowane przechowywaniem informacji na temat elementów konfiguracji w relacyjnych bazach danych w tradycyjny sposób, gdzie informacje o elementach jednego typu znajdują się w jednej tabeli. Schemat bazy danych tworzony jest na sztywno, a w konsekwencji dodanie nowych atrybutów lub typów elementów wymaga zarówno zmian w bazie danych jak i modyfikacji systemu. Moja praca dyplomowa jest próbą stworzenia niedrogiego, intuicyjnego i adaptowalnego narzędzia, które będzie mogło być stosowane także przez średnie i małe przedsiębiorstwa.

## Zakres

ITIL jest standardem złożonym z bardzo dużej liczby procesów. Wykonanie systemu wspierającego wszystkie byłoby zadaniem przerastającym siły niejednej firmy, nie wspominając o możliwościach dyplomanta. W związku z tym z całego standardu zostały wybrane najważniejsze procesy związane z przekazaniem usług i ich eksploatacją.

Następnie procesy z obu tych grup zostały równolegle zaimplementowane w ramach dwóch prac dyplomowych inżynierskich. Procesy z grupy przekazania usług — zarządzanie komponentami usług i konfiguracją oraz zarządzanie zmianą — są opisane przez autora w niniejszej pracy. Pozostałe procesy z drugiej grupy — zarządzanie incydentami i problemami — zostały opisane przez Piotra Kalańskiego[[2]](#footnote-2).

Należy zaznaczyć, że w ramach współpracy autorów obu prac powstał jeden system, w którym wszystkie zaimplementowane procesy współdziałają ze sobą. Prace związane z projektowaniem i implementacją współużytkowanych części systemu oraz opisaniem przykładowej firmy zostały wykonane wspólnie.

## Układ

Rozdział drugi opisuje zbiór zaleceń ITIL z szczególnym uwzględnieniem zarządzani konfiguracją i zmianą. W rozdziale trzecim przedstawiono profil przykładowej firmy, która zamierza wdrożyć ITIL wraz z systemem informatycznym automatyzującym jego procesy. Rozdziały czwarty, piąty i szósty omawiają kolejne etapy wykonania tego systemu, odpowiednio: zebranie wymagań funkcjonalnych i niefunkcjonalnych, projekt oraz implementację. Następujący rozdział siódmy prezentuje zastosowania systemu po jego wdrożeniu w opisanej wcześniej firmie. Rozdział ósmy skupia się na omówieniu testów poprawności i wydajności implementacji. Kończący pracę rozdział dziewiąty stanowi domknięcie i podsumowanie całości.

## Nazewnictwo

Wszystkie nazwy związane z ITIL występujące w tej pracy zostały przetłumaczone zgodnie z oficjalnym glosariuszem wydanym przez OGC[[3]](#footnote-3).

# ITIL

2.5 + 1.5 + 6 + 7 = 17

Wspomnieć o tym że piszę firma, a może to też równie dobrze być wewnętrzny departament.

## Pojęcia podstawowe

### Usługa

### Zarządzanie usługami IT (ITSM)

### Funkcja

### Proces

## Historia

Prace nad ITIL zaczęły się ponad trzydzieści lat temu — w drugiej połowie lat osiemdziesiątych. Rząd Wielkiej Brytanii szukał sposobów na ograniczenie wydatków związanych z utrzymaniem systemów informatycznych, które sięgały kwoty około 8 miliardów funtów w skali roku[[4]](#footnote-4). Źródła wysokich kosztów dopatrywano się przede wszystkim w fakcie, że wszystkie departamenty i instytucje rządowe zarządzały infrastrukturą IT we własnym zakresie. Nie istniały jednolite procedury i standardy, a co za tym idzie możliwości przeprowadzania audytów i kontroli wydatków były znacznie ograniczone. W celu rozwiązania problemu zlecono komórce rządowej o nazwie Central Computer and Telecommunications Agency (CCTA) opracowanie metody zarządzania pozwalającej na efektywne wykorzystanie zasobów IT.

Tak w roku 1988[[5]](#footnote-5) powstał GITIMM (ang. Government Information Technology Infrastructure Management Method) będący prekursorem ITIL. Myślą przewodnią nowej metodyki było ujednolicenie sposobu zarządzania IT. Dzięki wprowadzeniu wspólnego języka i stosowaniu podobnych praktyk możliwa stała się profesjonalizacja pracowników zarządzających infrastrukturą IT, co w konsekwencji pozwoliło podnieść efektywność i obniżyć koszty. Ponadto metodyka zawierała bardzo wiele cennych i sprawdzonych porad opartych na doświadczeniach wielu różnych firm, z którymi prowadzono konsultacje. W szczególności skorzystano z opracowanych przez IBM i wydanych w 1980 r. publikacji z serii „A Management System for the Information Business” opisujących koncepcję ITSM. GITIMM spełnił oczekiwania i został szybko przyjęty przez sektor prywatny oraz instytucje rządowe nie tylko Wielkiej Brytanii.

### Wersja pierwsza

Rok później, w sierpniu 1989, podczas krótkiej rozmowy dwóch twórców zmieniono nazwę na ITIL[[6]](#footnote-6). Zmiana była kluczowa, ponieważ z nazwy zniknęło słowo „government” oznaczające rząd, które odstraszało wielu przedsiębiorców. Ponadto zrezygnowano ze słowa „metoda”, ponieważ metodyka ta jest w rzeczywistości zbiorem wskazówek, a nie sztywnych wytycznych. W ciągu kilku lat wydano łącznie 42 tomy składające się na pierwszą wersję ITIL[[7]](#footnote-7). Do popularyzacji i dalszego rozwoju metodyki przyczyniła się utworzona w 1991 r. [[8]](#footnote-8) w internecie grupa użytkowników ITIMF (ang. IT Infrastructure Management Forum), obecnie funkcjonująca pod nazwą itSMF (ang. IT Service Management Forum) jako organizacja non-profit szerząca wiedzę o ITSM.

### Wersja druga

W 2000 roku podjęto próbę uporządkowania i ujednolicenia metodyki. Efektem był podział ITIL na dwa kluczowe obszary zawierające łącznie jedenaście dyscyplin. Pierwszy obszar — wspieranie usług (ang. Service Support) — obejmuje sześć dyscyplin związanych z technologią, niezbędnych do zapewnienia wysokiej jakości świadczonych usług:

|  |  |
| --- | --- |
| Nazwa dyscypliny | Opis |
| Service Desk | Zapewnia pojedynczy punktu kontaktu dla użytkowników końcowych potrzebujących wsparcia. Skupia się na komunikacji z klientem i kontroli incydentów. Jedyna dyscyplina będąca funkcją a nie procesem. |
| Zarządzanie incydentami  (Incident Management) | Głównym celem tej dyscypliny jest przywrócenie działania usługi po wystąpieniu incydentu w jak najkrótszym czasie i przy uwzględnieniu obowiązujących SLA[[9]](#footnote-9). Incydent to zdarzenie, które wpływa lub może wpływać na obniżenie jakości świadczonej usługi. |
| Zarządzanie problemami  (Problem Management) | Skupia się na znalezieniu problemów będących źródłem incydentów i podjęciu działań korygujących. W trakcie naprawy udostępnia informacje o znanych błędach i metodach ich obejścia. |
| Zarządzanie zmianami  (Change Management) | Zapewnia standardowe procedury zarządzania wszelkimi zmianami dotyczącymi usług. Minimalizuje ich negatywny wpływ poprzez odpowiednią ocenę, autoryzację i planowanie każdej zmiany. |
| Zarządzanie wersjami  (Release Management) | Grupuje zmiany w wersje i odpowiada za skoordynowanie ich implementacji oraz wdrożenia. |
| Zarządzanie konfiguracją  (Configuration Management) | Zajmuje się przechowywaniem i ciągłą aktualizacją informacji o wszystkich komponentach potrzebnych do świadczenia usług. Stanowi bazę wiedzy, z której korzystają inne procesy. |

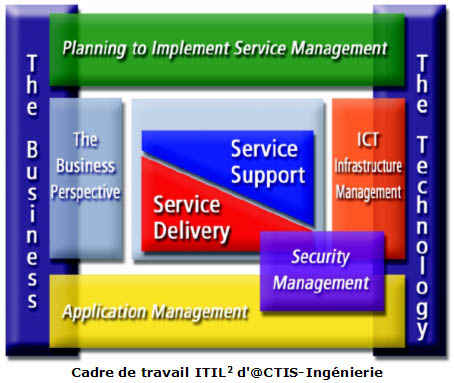
Tabela .. Dyscypliny wspierania usług ITIL v2

Drugi obszar — świadczenie usług (ang. Service Delivery) — jest bliższy perspektywie biznesowej i dotyczy przestrzegania gwarancji danych klientowi przez świadczącego usługi. Składa się z pięciu dyscyplin:

|  |  |
| --- | --- |
| Nazwa dyscypliny | Opis |
| Zarządzanie poziomem usług  (Service Level Management) | Pozwala na dopasowanie usług IT do wymagań biznesu. Skupia się na określeniu przejrzystych oczekiwań i wymagań klienta w stosunku do pobieranych przez niego usług. Dyscyplina ta zajmuje się tworzeniem katalogu usług, zbieraniem wymagań i negocjacją SLA. |
| Zarządzanie finansami  (Financial Management) | Zajmuje się obliczaniem kosztów świadczonych usług, ich księgowaniem oraz ustalaniem budżetu. Pozwala na funkcjonowanie wewnętrznych departamentów IT jako jednostek biznesowych. |
| Zarządzanie dostępnością  (Availability Management) | Celem tej dyscypliny jest zapewnienie dostępności usług zgodnie z SLA w efektywny kosztowo sposób. Wymaga to analizy, planowania, mierzenia i ewentualnej poprawy wszystkich czynników mających wpływ na dostępność. |
| Zarządzanie pojemnością  (Capacity Management) | Dyscyplina podobna do zarządzania dostępnością, zorientowana na monitorowanie i przestrzeganie gwarancji dotyczących pojemności świadczonych usług. |
| Zarządzanie ciągłością  usług informatycznych  (IT Service Continuity Management) | Znana także jako zarządzanie sytuacją kryzysową. Głównym celem jest przywrócenie działania systemów informatycznych firmy w uzgodnionym czasie. Składa się z zapobiegania wystąpieniu sytuacji kryzysowych i opracowaniu planów przeciwdziałania im. |

Tabela .. Dyscypliny świadczenia usług ITIL v2

Po wydaniu pierwszych dwóch tomów dotyczących wsparcia i świadczenia usług powstało jeszcze pięć następnych opisujących inne obszary i dwa dotyczące odpowiednio ITSM oraz stosowania ITIL w małych przedsiębiorstwach. W tym czasie doszło do restrukturyzacji instytucji rządowych Wielkiej Brytanii, na skutek której w 2001 r. OGC (ang. Office of Government Commerce) wchłonęło CCTA i tym samym stało się posiadaczem praw do ITIL. Druga wersja wprowadziła jasno wydzielone procesy i znacząco uporządkowała czynności opisane w wersji pierwszej. Skonsolidowano wiedzę zawartą w czterdziestu dwóch tomach do zaledwie dziewięciu.



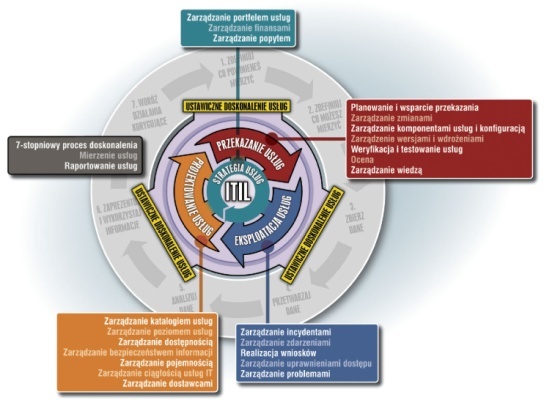
Ilustracja .. Obszary ITIL v2[[10]](#footnote-10)

### Wersja trzecia

W listopadzie 2004 r.[[11]](#footnote-11) OGC uruchomiło projekt ITIL Refresh, którego celem było stworzenie kolejnej wersji ITIL. Projekt w dużej mierze polegał na przeprowadzaniu konsultacji z tysiącami organizacji, które używały ITIL, i skorzystaniu z ich opinii oraz doświadczeń. W maju 2007 r. projekt został zakończony, a jego rezultatem było opublikowanie pięciu tomów ITIL w wersji trzeciej. Zmian było bardzo dużo, zarówno w strukturze metodyki, poszczególnych procesach, jak i samej filozofii zarządzania infrastrukturą IT. Zakres poszerzono blisko dwukrotnie, zwiększając liczbę procesów i funkcji prawie trzykrotnie. W lipcu 2010 r. zaprzestano prowadzenia egzaminów ITIL v2, a dokładnie rok później zaprzestano publikacji materiałów i oficjalnie wycofano wsparcie dla tej wersji.

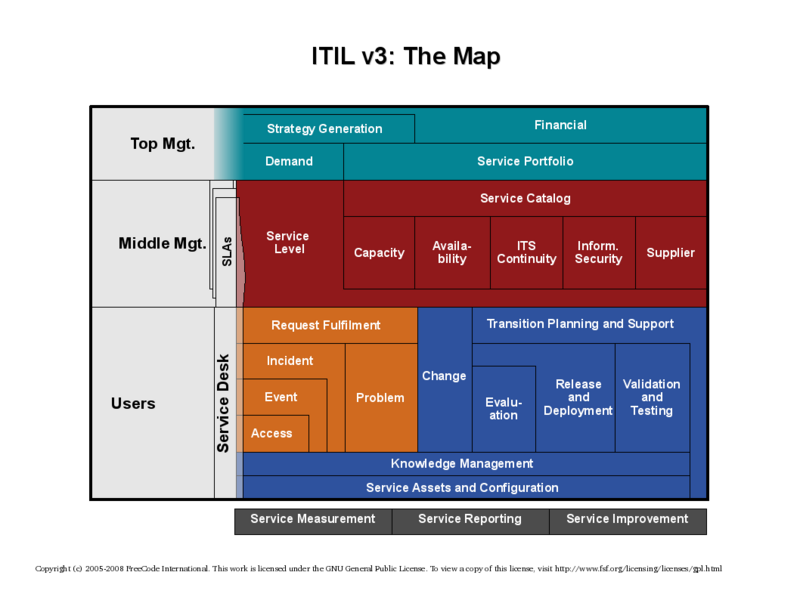
## Struktura

Kluczową zmianą w trzeciej wersji ITIL jest odejście od modelu w którym biznes i IT stanowią oddzielne ośrodki, które ze sobą współpracują. Podział ten zrodził się z przekonania, iż oba pełnią funkcje zasadniczo różniące się od siebie. Z tego powodu nacisk w zarządzaniu był kładziony na dopasowanie IT i biznesu (ang. Business IT Alignment). Niestety wraz z rozwojem technologii i wzrostem złożoności świadczonych usług współpraca między nimi stawała się coraz bardziej zażyła i skomplikowana. Obecnie oba ośrodki są wręcz nierozerwalne i sytuacja ta rodzi szereg problemów. Patrząc na usługę z perspektywy biznesu pomija się techniczne zawiłości, a w konsekwencji naraża na obniżenie jakości lub przerwy w działaniu usługi w związku z niedostateczną dostępnością, pojemnością lub błędami. Z drugiej strony spojrzenie IT nie jest w stanie uchwycić faktycznych potrzeb klienta i podejmować decyzji o charakterze strategicznym. Dlatego ITIL v3, zamiast starać się dopasować biznes i IT do siebie, mówi o ich integracji. Produktem tego innowacyjnego podejścia jest stworzenie modelu cyklu życia usługi IT (ang. IT service lifecycle), w którym obie domeny dopełniają się.



Ilustracja .. Cykl życia usługi w ITIL[[12]](#footnote-12)

Cykl życia usługi w ITIL v3 składa się z pięciu obszarów, którym odpowiada pięć dotychczas wydanych publikacji. Obszary te tworzą trzy koncentryczne, pierścienie, które obracając się wprawiają w ruch pozostałe (patrz: Ilustracja 2.2). Każda faza cyklu życia składa się z zestawu procesów i posiada jasno zdefiniowane wejścia oraz wyjścia, dzięki który komunikuje się z innymi. Pierwszym i zarazem centralnym obszarem jest strategia usług, odpowiedzialna za określenie portfela usług[[13]](#footnote-13) z uwzględnieniem dalekosiężnych celów przedsiębiorstwa. Wizja usługi stanowi początek, na podstawie którego projektowanie usług opracowuje specyfikację i architekturę rozwiązania, biorąc pod uwagę wymagania jakościowe, wydajnościowe, ryzyko i inne. W ten sposób powstaje pakiet projektu usługi (ang. Service Design Package), który jest przesyłany do obszaru przekazania usług odpowiedzialnego za jej implementację oraz wdrożenie. Działająca usługa znajduje się w obszarze eksploatacji usług. Zadaniem tego etapu jest zagwarantowanie dostarczania usługi zgodnie z uzgodnionymi warunkami. Gdy zajdzie konieczność poważnej modyfikacji, usługa jest ponownie przekazywana do etapu projektowania usług lub wycofywana. Ostatnim obszarem jest ustawiczne doskonalenie usług, które odpowiada za ciągłe podnoszenie jakości i wydajności usług.



Ilustracja .. Procesy ITIL pogrupowane wg hierarchii zarządzania[[14]](#footnote-14)

### Strategia usług (Service Strategy)

„People do not want quarter-inch drills. They want quarter-inch holes.”

(Ludzie nie chcą ćwierć-calowych wierteł. Chcą ćwierć-calowych otworów)

***Profesor Emeritus Theodore Levitt, Harvard Business School***[[15]](#footnote-15)

Źródłem sukcesu usługodawcy jest dogłębne rozumienie potrzeb rynku. Klienci chcą kupować zaspokojenie swoich potrzeb, a nie sprzedawane im produkty. Dlatego przed określeniem swojej oferty, usługowca musi zidentyfikować potencjalne rynki zbytu i dokładnie je zbadać. Dopiero na podstawie zebranych informacji jest w stanie określić jakie usługi są potrzebne i będą generować zyski.

Dokładnie tym zajmuje się etap strategii usług, obsługiwany przez najwyższe kierownictwo firmy. Ważnym procesem tego obszaru jest zarządzanie portfelem usług, który składa się z katalogu usług[[16]](#footnote-16) widocznych dla klienta oraz usług niewidocznych: będących w przygotowaniu lub wcześniej wycofanych. To właśnie w jego obrębie zapadają decyzje o tworzeniu, zmianach, zakończeniu lub wznowieniu świadczenia usług.

Aby móc efektywnie zarządzać portfelem usług, trzeba najpierw stworzyć strategię przedsiębiorstwa wyznaczającą cele i kierunek rozwoju w długiej perspektywie czasu. Jest to kluczowy proces pozwalający na specyfikację ram funkcjonowania firmy, określenie docelowych rynków oraz na stworzenie szeregu metryk, według których działanie przedsiębiorstwa będzie mogło być oceniane i w razie potrzeby korygowane. Definiując strategię otrzymuje się jasny obraz firmy, jej bieżącej i docelowej pozycji oraz wytyczne jak drogę te pokonać. Wiedza ta stanowi niezbędną bazę na podstawie której można oceniać usługi zarówno świadczone jak i dopiero tworzone.

Jednak informacje i plany same w sobie nie wystarczą, trzeba jeszcze odnaleźć złoty środek między konsumpcją świadczonych usług oraz kosztami przez nie generowanymi i cały czas go monitorować. Z pomocą przychodzą ostatnie dwa procesy tego obszaru: zarządzanie popytem i zarządzanie finansami. Pierwszy z nich skupia się na monitorowaniu zapotrzebowania oraz pozwala na dostosowywanie pojemności usług zgodnie z wahaniami rynku i długofalowym planem. Drugi kontroluje zyski i wydatki firmy, co pozwala na ustalanie odpowiedniego stosunku ceny do jakości świadczonych usług oraz funkcjonowanie przedsiębiorstwa w uwarunkowany ekonomicznie sposób.

### Projektowanie usług (Service Design)

Celem tej fazy cyklu życia usługi jest, zgodnie z nazwą, projektowanie usług w ramach wizji określonej przez strategię usług. Głównym artefaktem, powstającym w ramach współdziałania wszystkich siedmiu procesów tego etapu, jest katalog usług zawierający informacje o wszystkich świadczonych oraz obecnie implementowanych usługach. Stanowi on także podzbiór portfela usług, który dodatkowo przechowuje dane o usługach planowanych na przyszłość. Procesem zapewniającym przechowywanie informacji o katalogu i ich spójność jest zarządzanie katalogiem usług[[17]](#footnote-17).

W momencie gdy zajdzie potrzeba stworzenia nowej usługi lub zmiany istniejącej, procesem zaczynającym projektowanie usługi jest zarządzanie poziomem usług[[18]](#footnote-18). Jego zadaniem jest negocjacja i zebranie biznesowych wymagań dotyczących usługi oraz zapisanie ich w postaci formalnej jako wymagań dla poziomu usługi (SLR)[[19]](#footnote-19) oraz umów SLA. Następnie na ich podstawie tworzona jest specyfikacja usługi składająca się z: odpowiedniej architektury, procesów, zasad oraz dokumentacji; spełniających teraźniejsze i przyszłe potrzeby przedsiębiorstwa[[20]](#footnote-20). Pozostałe pięć procesów obszaru projektowania usług pomaga w tym kroku. Zarządzanie pojemnością i zarządzanie dostępnością odpowiadają za określenie infrastruktury potrzebnej do spełnienia odpowiadającym im wymagań zawartych w SLA. Polityka bezpieczeństwa i poufności jest egzekwowana przez zarządzanie bezpieczeństwem informacji[[21]](#footnote-21), natomiast zarządzanie ciągłością usług informatycznych[[22]](#footnote-22) odpowiada za zabezpieczenie usługi przed sytuacjami kryzysowymi i opracowanie planów przeciwdziałania. Ostatnim procesem jest zarządzanie dostawcami[[23]](#footnote-23), które definiuje wymagania względem kooperantów dostarczających komponenty usługi. Ponadto ta piątka procesów pełni rolę proaktywną i na bieżąco monitoruje, czy działające usługi spełniają nałożone na nie wymagania.

W ten sposób powstaje pakiet projektu usługi[[24]](#footnote-24), który zawiera projekt oraz opis oczekiwanych rezultatów, potencjalnego ryzyka i wpływu na inne usługi. Jest to kompletna specyfikacja usługi, według której w kolejnej fazie zostanie zaimplementowana. Aby mieć pewność, że zbudowana usługa faktycznie spełnia wymagania funkcjonalne i jakościowe, równolegle powstaje drugi dokument zawierający kryteria odbioru usługi[[25]](#footnote-25).

### Przekazanie usług (Service Transition)

Po zaprojektowaniu usługi przychodzi czas na jej implementację i wdrożenie. Obie te czynności odbywają się w etapie przekazania usługi, który buduje systemy niezbędne do świadczenia usługi i przygotowuje przedsiębiorstwo do ich przyjęcia.

Centralnym procesem tej fazy cyklu życia usługi jest planowanie i wsparcie przekazania[[26]](#footnote-26). Jego celem jest określenie wszystkich etapów potrzebnych do przekazania usługi i koordynacja ich wykonania. Dzięki skupieniu się na implementacji całej usługi, a nie poszczególnych systemów, można znacznie skuteczniej identyfikować i przeciwdziałać ryzyku. Ponadto w zakresie obowiązków tego procesu leży zabezpieczenie wszystkich potrzebnych zasobów i zapewnienie wsparcia dla zespołów. Należy zwrócić uwagę, że ITIL nie narzuca sposobu prowadzenia poszczególnych projektów i może być używany razem z metodykami takimi jak PRINCE2[[27]](#footnote-27), czy agile. Wybór odpowiedniej, w zależności od rodzaju i rozmiaru projektu, leży w kompetencji firmy.

Gdy usługa zostanie zbudowana musi zostać dokładnie sprawdzona. Zajmuje się tym proces weryfikacji i testowania usług[[28]](#footnote-28) i proces oceny[[29]](#footnote-29). Pierwszy z nich bada zgodność usługi ze specyfikacją oraz potrzebami biznesowymi. Wybudowana usługa jest analizowana zarówno pod kątem użyteczności, jak i gwarancji opisanych w pakiecie projektu usługi. Dokładne testowanie bezpośrednio przekłada się na podniesienie jakości i pozwala uniknąć konsekwencji związanych z ewentualnymi awariami. Drugi sprawdza wydajność usługi względem kryteriów zdefiniowanych w SAC.

Gdy weryfikacja zakończy się pomyślnie, usługa zostaje uruchomiona z pomocą procesu zarządzania wersjami i wdrożeniami[[30]](#footnote-30). Krok ten wymaga zgromadzenia wszystkich potrzebnych zasobów i artefaktów w jeden spójny pakiet wersji[[31]](#footnote-31) i zaplanowania kolejnych etapów wdrożenia zgodnie z ustaloną polityką. Pozwala to skoordynować modyfikacje środowiska operacyjnego spowodowane jednoczesnym wdrażaniem i zmianą wielu usług, co minimalizuje ryzyko zakłóceń w dostawie działających usług. Proces ten może także zapewniać wczesne wsparcie[[32]](#footnote-32) dla wdrożonych usług, szybko rozwiązując problemy powstałe w początkowej fazie życia usługi.

Etap przekazania usług zajmuje się także małymi zmianami w działających usługach. Wszystkie potrzeby modyfikacji są zgłaszane jako wnioski o zmianę (RFC)[[33]](#footnote-33), które trafiają do procesu zarządzania zmianami[[34]](#footnote-34). Następnie są rejestrowane, oceniane i autoryzowane przez kompetentny organ. Umożliwia to podejmowanie rozważnych decyzji, minimalizację ryzyka i szybkie przywracanie usług do stanu sprawności po wykryciu problemów.

Pozostałe dwa procesy, mimo że znajdują się w obszarze przekazania usług, dotyczą całego cyklu życia usługi. Zarządzanie komponentami usług i konfiguracją[[35]](#footnote-35) odpowiada za przechowywanie w bazie zarządzania konfiguracją (CMDB)[[36]](#footnote-36) i uaktualnianie informacji na temat elementów konfiguracji[[37]](#footnote-37), czyli wszystkich obiektów związanych z usługami. Są to między innymi: usługi, składowe usług, zasoby strategiczne, dokumenty firmowe, dokumenty generowane przez inne procesy (SLA, RFC itp.), systemy, elementy infrastruktury IT, oprogramowanie, elementy infrastruktury budowlanej, obiekty potrzebne do świadczenia usług związane ze specyfiką branży i inne. Ponadto w CMDB są trzymane informacje na temat pracowników, kooperantów i klientów związanych z danymi elementami. Dane o wszystkich ważnych dla przedsiębiorstwa elementach możemy wykorzystać do pozyskiwania różnego rodzaju interesujących faktów. Do tego celu służy proces zarządzania wiedzą[[38]](#footnote-38), którego celem jest udostępnianie informacji wspomagających podejmowanie decyzji odpowiednim ludziom w odpowiednim czasie.

### Eksploatacja usług (Service Operation)

Obszar ten zajmuje się dostarczaniem usług klientom zgodnie z ustalonymi w SLA kryteriami. W jego obrębie znajdują się wszystkie wdrożone i działające usługi, które są pielęgnowane i utrzymywane w stanie operacyjnym. Wymaga to obsługi interakcji z użytkownikami, monitorowania stanu infrastruktury i poziomu świadczenia usług oraz ewentualnego podjęcia działań naprawczych.

Kluczowym procesem etapu eksploatacji usług jest zarządzanie zdarzeniami[[39]](#footnote-39). Zdarzenie[[40]](#footnote-40) jest wykrywalnym i rozpoznawalnym zjawiskiem, które może mieć bezpośredni lub pośredni wpływ na świadczenie usługi[[41]](#footnote-41). Rejestrowanie wszystkich zdarzeń i ocenianie ich wpływu umożliwia podjęcie w odpowiednim czasie, odpowiednich decyzji i kroków minimalizujących ryzyko zakłóceń w dostawie usługi. Większość zdarzeń jest generowana automatycznie przez odpowiednie narzędzia monitorujące.

Jeśli jednak ryzyko awarii zrealizuje się i nastąpi przerwa w świadczeniu lub obniżenie jakości usługi, mamy do czynienia z incydentem[[42]](#footnote-42). Zgłoszenia tego typu są gromadzone przez proces zarządzania incydentami[[43]](#footnote-43), który stara się jak najszybciej przywrócić, określone w SLA, oczekiwane działanie usługi. Incydenty są priorytetyzowane i kategoryzowane, co pozwala na znalezienie powtarzających się zgłoszeń. Źródła incydentów nazywane są problemami[[44]](#footnote-44). Za ich znalezienie, a następnie opracowanie planu naprawczego i jego implementację odpowiada proces zarządzania problemami[[45]](#footnote-45). Gdy problem zostanie zidentyfikowany, tworzony jest rekord[[46]](#footnote-46) w bazie znanych błędów[[47]](#footnote-47). Pomaga to podjąć szybkie działania, w sytuacji jeżeli problem powtarza się co jakiś czas. Ponadto dla znanych błędów może zostać sformułowane obejście problemu, pozwalające użytkownikom w pewnym stopniu zmniejszyć skutki awarii. Efektem końcowym tej procedury jest zgłoszenie wniosku o zmianę.

Innym ważnym procesem tej grupy jest zarządzanie uprawnieniami dostępu[[48]](#footnote-48), które stoi na straży polityki poufności i bezpieczeństwa. Proces ten odpowiada za udostępnienie właściwym użytkownikom, odpowiednich usług, zasobów lub informacji. Wymaga to mechanizmów weryfikacji tożsamości, kontroli uprawnień i śledzenia dostępu.

Wewnątrz obszaru eksplantacji usług mieści się funkcja Service Desk, która — tak jak w wersji drugiej ITIL — odpowiada za kontakt z użytkownikami usług. Może posiadać różne formy organizacji, jednak zdecydowanie najczęściej przyjmuje kształt centrum telefonicznego. Pracownicy działu obsługi klienta, oprócz wprowadzają do systemu zasygnalizowanych zgłoszeń incydentów, mogą spełniać drobne żądania — wnioski o usługę[[49]](#footnote-49). Są to małe zmiany, o niewielkim koszcie i ryzyku, z góry zatwierdzone np.: prośba o zmianę hasła, zainstalowanie aplikacji itp.; a także zapytania, skargi i zażalenia. Wykonaniem czynności związanych z ich obsługą i rozwiązaniem zajmuje się proces realizacji wniosków[[50]](#footnote-50). Pozwala to na utworzenie szybkiego kanału komunikacji i dostosowanie świadczonych usług zgodnie z wolą klienta.

### Ustawiczne doskonalenie usług (Continual Service Improvement)

Trzecim pierścieniem cyklu życia usługi, który wprawia w ruch wszystkie pozostałe, jest ustawiczne doskonalenie usług. Obszar ten jest odpowiedzialny za ciągłe dostosowywanie się firmy do zmieniających się potrzeb biznesu i pozwala sterować nią zgodnie z wyznaczonymi kierunkami. Aby było to możliwe należy zadać sobie sześć kolejnych pytań[[51]](#footnote-51):

1. Jaka jest wizja przedsiębiorstwa?
2. Gdzie jesteśmy teraz?
3. Gdzie chcemy być?
4. Jak tam się dostać?
5. Czy już się tam jesteśmy?
6. Jak utrzymać prędkość?

Odpowiedź wymaga uprzednio jasnego zdefiniowania celów i metryk. Podstawowym elementem obszaru ustawicznego doskonalenia usług są właśnie pomiary. Jedynie na podstawie wiarygodnych i miarodajnych informacji można kontrolować i zarządzać. Z pomocą przychodzi proces mierzenia usług[[52]](#footnote-52), który definiuje zasady wyznaczania metryk, zbiera dane z pomiarów, odpowiednio je filtruje i konsoliduje, a w końcu analizuje zgodnie z wyznaczonymi celami. Na podstawie tych badań proces raportowania usług[[53]](#footnote-53) tworzy odpowiednie zestawienia, które są wysyłane do kompetentnych osób podejmujących decyzje o ewentualnych usprawnieniach. Całością zarządza proces ustawicznego doskonalenia usług[[54]](#footnote-54) pracujący według siedmiokrokowego planu poprawy[[55]](#footnote-55):

1. Zdefiniuj co powinieneś mierzyć.
2. Zdefiniuj co możesz mierzyć.
3. Zbierz dane.
4. Przetwórz dane.
5. Analizuj dane.
6. Pokaż i wykorzystaj informacje.
7. Zaimplementuj akcję korygującą.

## Wyszczególnione procesy

Podczas omówienia ITIL, szczególnie interesują nas dwa procesy należące do obszaru przekazania usług. Są to: zarządzanie komponentami usług i konfiguracją oraz zarządzanie zmianą. Wsparcie dla obu zostało zaimplementowane w stworzonym systemie Service Desk. Ponadto system obsługuje procesy zarządzania incydentami i problemami, które zostały opisane w pracy Piotra Kalańskiego[[56]](#footnote-56).

### Zarządzanie komponentami usług i konfiguracją

Cel procesu

Zakres obowiązków tego procesu można streścić bardzo krótko, jest nim administrowanie bazą zarządzania konfiguracją (CMDB)[[57]](#footnote-57). Baza ta zawiera informacje na temat wszystkich ważnych z punktu widzenia przedsiębiorstwa elementów konfiguracji:

“Configuration Item (CI): Any Component that needs to be managed in order to deliver an IT Service. […] CIs typically include IT Services, hardware, software, buildings, people and formal documentation such as Process documentation and SLAs.”

(Element konfiguracji (CI): Dowolny komponent, który musi być zarządzany w celu dostarczania usługi IT. […] Elementy konfiguracji zazwyczaj obejmują usługi IT, sprzęt, oprogramowanie, budynki, ludzi i formalną dokumentację, taką jak dokumentacja procesu i SLA.)

Definicja elementu konfiguracji, źródło: (Knowledge Transfer)

Posiadanie danych o wszystkich potrzebnych komponentach jest bardzo ważne z wielu powodów. Po pierwsze utrzymanie usługi w stanie operacyjnym wymaga identyfikacji i śledzenia stanu wszystkich wykorzystywanych elementów. Niesprawność jednego z nich może równać się zakłóceniom w dostawie całej usługi. Defekt może zostać wywołany przez fizyczną awarię, ale może wystąpić także z powodu nieświadomych lub nieskoordynowanych zmian. Dysponując informacjami możemy temu zapobiec poprzez wyszukanie zależności niosących duże prawdopodobieństwo awarii oraz koordynując zmiany dotyczące elementów używanych przez wiele usług.

Po drugie zbieranie danych znacząco ułatwia odnalezienie komponentów nie spełniających norm jakości: przeciążonych, wadliwych, zużytych; oraz nie spełniających norm prawnych, w szczególności oprogramowania bez ważnej licencji. Przeciwdziałanie tym zjawiskom redukuje koszty awarii i ewentualnych procesów sądowych oraz pośrednio wpływa na podniesienie jakości świadczonych usług.

Po trzecie wszystkie procesy decyzyjne występujące podczas całego cyklu życia usługi — od fazy strategii; przez etap projektowania, wdrażania, konsumpcji; do jej wycofania — nie są w stanie funkcjonować bez informacji o faktycznym stanie elementów konfiguracji. Wyborów świadomych i efektywnych można dokonywać jedynie posiadając obiektywne i miarodajne dane oraz odpowiednią wiedzę opartą o nie. Baza zarządzania konfiguracją stanowi pojedyncze i spójne źródło, które dostarcza te informacje.

Z tych powodów zarządzanie komponentami usług i konfiguracją odgrywa podstawową i kluczową rolę w całym ITIL. Bez bazy zarządzania konfiguracją w praktyce niemożliwe jest udane wprowadzenie innych procesów.

Kluczowe pojęcia

Model konfiguracji[[58]](#footnote-58)

Baza zarządzania konfiguracją przechowuje model, będący odbiciem faktycznego stanu elementów konfiguracji. Przechowywane informacje mogą mieć dowolną szczegółowość, zależną od potrzeb przedsiębiorstwa. Jednak czasami wartość danych przewyższają koszty związane z utrzymaniem ich aktualnego stanu. Dlatego ważnym wyborem, podejmowanym przy wprowadzaniu procesu zarządzania komponentami usług i konfiguracją jest określenie śledzonych elementów. Na przykład komponenty takie jak myszki i klawiatury, ze względu na małą wartość i niskie ryzyko związane z ich awarią, mogą być pominięte w bazie zarządzania konfiguracją. Model konfiguracji ustala kryteria, według których elementy konfiguracji są identyfikowane, klasyfikowane i opisywane. Pozwala to zachować balans między zyskiem i kosztem pozyskiwania informacji.

Element konfiguracji

Zgodnie z przytoczoną na wstępie definicją element konfiguracji to obiekt ważny dla funkcjonowania usługi. Każdy jest opisywany w bazie zarządzania konfiguracją przez zbiór atrybutów zgodny z ustalonym modelem konfiguracji. Ponadto posiada fizycznie dołączony unikalny identyfikator, którego format zależy od konwencji firmy, pozwalający go jednoznacznie odróżnić od pozostałych i odszukać w systemie. Elementami konfiguracji mogą być:

* Świadczone usługi, ich komponenty i wykorzystywane przez nie zasoby;
* Produkty cyklu życia usług takie jak projekty, plany przeciwdziałania awarii, wersje, zgłoszenia incydentów, żądania zmian, problemy, umowy SLA, SAC i inne;
* Dokumenty formalne określające np. strategię i politykę przedsiębiorstwa, wymagania wewnętrzne lub prawne, harmonogramy prac itp.;
* Ludzie: pracownicy, kooperanci i klienci;
* Elementy wirtualne używane do grupowania i kategoryzacji;
* Komponenty zewnętrzne, nie należące do firmy, w tym konsumowane usługi.

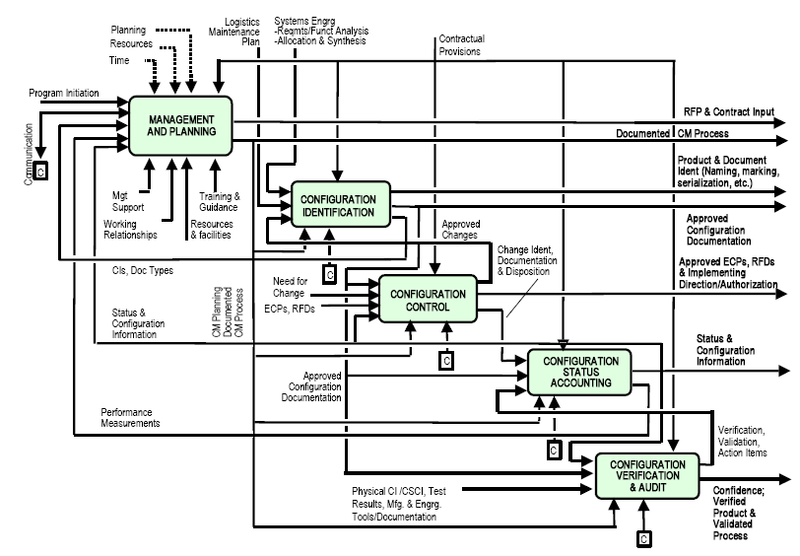
Istotną informację związaną z elementami konfiguracji niosą różnego typu relacje w jakie komponenty wchodzą ze sobą. Na przykład usługi mogą korzystać z systemów informatycznych, te z kolei mogą być złożone z ustalonych komponentów i administrowane przez konkretną grupę ludzi. Powiązane elementy konfiguracji tworzą sieć, w której znajduje się niepoliczalna liczba interesujących faktów i wiedzy. Dzięki niej można szacować wpływ zmian, obliczyć prawdopodobieństwo awarii i doszukiwać się wielu innych zjawisk.

System zarządzania konfiguracją[[59]](#footnote-59)

Jest to system informatyczny fizycznie przechowujący dane elementów konfiguracji. Pozwala na zarządzanie tymi danymi odpowiednio uprawnionym zarządzającym konfiguracją[[60]](#footnote-60). Może korzystać z wielu źródeł takich jak bazy zarządzania konfiguracją, odpowiednio sformatowane dokumenty, narzędzia do automatycznego wykrywania konfiguracji, systemy działające w firmie (np. przechowujące dane pracowników lub klientów) oraz inne.

Czynności

Proces zarządzania konfiguracją zaczyna działanie podczas projektowania usługi w etapie przekazania. Pierwszym jego krokiem jest zarządzanie i planowanie, które polega na opracowaniu planu zarządzania konfiguracją. Zawiera on szczegółowy opis procedur potrzebnych do identyfikacji, kontroli, weryfikacji i audytu elementów konfiguracji danej usługi. Następnie, podczas identyfikacji konfiguracji, powstaje model konfiguracji i zgodnie z nim do systemu zostają wprowadzone dane wszystkich komponentów. Po tym kroku żaden element nie może być modyfikowany bez odpowiednich zezwoleń, a każda zmiana musi zostać odnotowana w bazie zarządzania konfiguracją. Te ograniczenia są niezbędne do zachowania ważności i spójności zapisanych informacji. Za autoryzację wprowadzanych zmian odpowiada kontrola konfiguracji, natomiast za uaktualnienie zapisanego stanu — raportowanie statusu konfiguracji. Ostatnią grupą czynności związaną z tym procesem jest weryfikacja i audyt konfiguracji. Jej celem jest sprawdzenie czy dane przechowywane w bazie zarządzania konfiguracją odzwierciedlają stan faktyczny. Audyty konfiguracji powinny być przeprowadzane w regularnych odstępach czasu oraz dodatkowo przed wydaniem głównych wersji.

Ilustracja . Model czynności zarządzania konfiguracją[[61]](#footnote-61)

Powiązane procesy

Zarządzanie komponentami usług i konfiguracją stanowi źródło danych, z którego intensywnie korzysta proces zarządzania wiedzą. Wszystkie opracowania i raporty wytworzone przez ten proces bazują na informacjach zawartych w systemie zarządzania konfiguracją. SACM jest także blisko powiązany z procesami zarządzania zmianą oraz zarządzania wersjami i wdrożeniami. Oba kontrolują i autoryzują zmiany elementów konfiguracji. Ponadto każdy inny proces w pewnym stopniu korzysta z systemu zarządzania konfiguracją wykorzystując jego dane lub wprowadzając do niego swoje produkty.

### Zarządzanie zmianą

Cel procesu

Zmiany są nieodłącznym elementem związanym z świadczeniem usług i systemami IT. Powstają zarówno reaktywnie na skutek wykrycia błędów, jak i proaktywnie w celu redukcji kosztów lub poprawienia jakości. Każda niesie ze sobą ryzyko niepowodzenia i w konsekwencji obniżenia jakości świadczenia usługi.

“The addition, modification or removal of anything that could have an effect on IT Services. The Scope should include all IT Services, Configuration Items, Process, Documentation, etc.”

(Dodanie, modyfikacja albo usunięcie czegokolwiek, co może mieć efekt na usługi IT. Zakres powinien obejmować wszystkie usługi IT, elementy konfiguracji, procesy, dokumentację itp.”

Definicja zmiany, źródło: (Knowledge Transfer)

Proces zarządzania zmianą wprowadza jednolite procedury obsługi wszystkich żądań zmian, opierające się na rejestracji, ocenie, autoryzacji i ewaluacji. Dzięki wprowadzeniu standardowych metod, wnioski te mogą być obsługiwane i implementowane znacznie szybciej. Pozwala to lepiej dostosować się do potrzeb klienta i biznesu. Ponadto systematyczne podejście umożliwia minimalizację ryzyka wprowadzenia zmian nieautoryzowanych, nieprzemyślanych, szkodliwych oraz ryzyka awarii wynikającego z braku koordynacji. Bezpośrednim skutkiem jest zmniejszenie liczby incydentów oraz przerw w świadczeniu usługi.

Kluczowe pojęcia

Wniosek o zmianę (RFC) [[62]](#footnote-62)

Formalny dokument zawierający propozycję zmiany i jej dokładny opis. RFC posiada odnotowaną datę złożenia, autora, identyfikator oraz opcjonalnie kategorię i wstępny priorytet. Duże zmiany mające znaczący wpływ na przedsiębiorstwo zawierają ponadto uzasadnienie podpisane przez odpowiednie kierownictwo. Wszystkie modyfikacje elementów konfiguracji muszą być zgłoszone w tej postaci. Wniosek o zmianę stanowi podstawę, a jego złożenie punkt początkowy procesu zarządzania zmianą.

Oprócz zwykłych wniosków o zmianę istnieją jeszcze dwa inne rodzaje: pilne[[63]](#footnote-63) oraz standardowe. Wnioski z pierwszej grupy stanowią zmiany, które trzeba wykonać natychmiast. Z racji na wysokie ryzyko i duży negatywny wpływ na usługę są one przetwarzane znacznie szybciej, z pominięciem wielu kroków kontrolnych procedury standardowej. Istnienie tego skrótu pozwala na szybszą reakcję na poważne i nieoczekiwane zmiany, w tym awarie. Natomiast zmiany standardowe to małe, częste i powtarzające się modyfikacje, niosące ze sobą niskie ryzyko. Do tego typu zmian kwalifikują się żądania zmiany hasła, wymiany klawiatury i temu podobne. Ich każdorazowa autoryzacja była by niepotrzebnym i nadmiernym obciążeniem dla całego procesu, dlatego odpowiedzialne organy poświadczają je z góry. Obsługą tego typu żądań zajmuje się Service Desk i proces realizacji wniosków.

Należy zwrócić uwagę na to, że często dochodzi do nadużyć językowych poprzez utożsamianie rekordu zmiany lub samej zmiany z RFC.

Rekord zmiany[[64]](#footnote-64)

Wpis w systemie informatycznym wspomagającym proces zarządzania zmianą, tworzony na podstawie zgłoszonego wniosku o zmianę. Odzwierciedla bieżący stan zmiany oraz dokumentuje jej cykl życia, wraz z wszystkimi podjętymi decyzjami. Dodatkowo zawiera wskazania do elementów konfiguracji, na które proponowana zmiana ma wpływ.

Zarządzający zmianami[[65]](#footnote-65)

Rola, którą pełnią pracownicy procesu zarządzania zmianą. W jej kompetencji leży przeprowadzanie rekordów zmian przez cały proces, w tym: filtrowanie niepoprawnych wniosków, przydzielanie kategorii i priorytetu, szacowanie wpływu, koordynacja implementacji oraz recenzja i zamknięcie zmiany. Zarządzający zmianą mogą posiadać prawo do autoryzacji niektórych zmian.

Rada ds. Zmian (CAB)[[66]](#footnote-66)

Organ decyzyjny złożony z przedstawicieli IT, biznesu, kooperantów i użytkowników — jednym zdaniem wszystkich zainteresowanych, na których zmiany mogą mieć wpływ. Dokładny skład CAB zależy głównie od rodzaju rozważanych zmian. Każda kategoria zmian — np. dotycząca usługi A, B lub jej komponentu — posiada własną radę, do której trafiają wszystkie rekordy zmian z tej kategorii. Rolą tego organu jest ich ostateczna autoryzacja, nadawanie priorytetów, harmonogramowanie oraz pomoc zarządzającym zmianą w ocenie i podejmowaniu decyzji. Zebrania rady mogą być organizowane elektronicznie, co znacząco oszczędza czas. Jednak w wypadku skomplikowany i dużych zmian zalecane są bezpośrednie spotkania.

Nadzwyczajna Rada ds. Zmian (ECAB)[[67]](#footnote-67)

Czasami, gdy potrzeba szybkich decyzji, zorganizowanie zebrania Rady ds. Zmian może być niemożliwe. W tym celu istnieje znacznie mniejszy organ, złożony z jak najmniejszej możliwej ilości osób. Organem tym jest Nadzwyczajna Rada ds. Zmian. Posiada ona taką samą charakterystykę, co zwykłe CAB, a jej obowiązkiem jest zarządzanie zmianami pilnymi.

Czynności

Proces zarządzania zmianą zaczyna się od stworzenia i złożenia wniosku o zmianę. Autorem żądania może być każda zainteresowana osoba. Forma zgłaszania wniosków jest dowolna i zależy od przedsiębiorstwa. Może obejmować wnioski pisemne, nadsyłane pocztą elektroniczną oraz wprowadzane przez formularze internetowe.

Następnie wniosek jest odbierany przez zrządzającego zmianą i zapisywany w systemie w postaci rekordu zmiany. Rejestracja wszystkich wniosków jest bardzo ważna. Z jednej strony pozwala wykorzystać wszystkie możliwości ulepszenia usług, a z drugiej pozwala bardzo szybko wychwycić żądania nadmiarowe, które pojawiły się już wcześniej.

Kolejnym krokiem jest przegląd żądań. W tej fazie zarządzający zmianą musi zapoznać się z treścią wniosków i odrzucić wszystkie te, które nie spełniają wymagań formalnych, duplikują się, są niekompletne albo w oczywisty sposób niestosowne. Podjęcie decyzji o oddaleniu wniosku wymaga podania przyczyny, a autorowi żądania przysługuje procedura apelacyjna. Zarządzający konfiguracją może także przekazać wniosek do innej kategorii lub zmienić jego priorytet.

Gdy wniosek przejdzie kryterium selekcji zostaje oceniony. Krok ten polega na oszacowaniu potencjalnych zysków, strat i ryzyka oraz wnikliwym przestudiowaniu wpływu zmiany na usługi i elementy konfiguracji. Dokładne jego oszacowanie pozwala uniknąć wielu incydentów i zakłóceń w dostawie usług. Dodatkowo należy wskazać osoby odpowiedzialne za ewentualne wdrożenie zmiany. Istnieje zasada siedmiu R[[68]](#footnote-68), która pomaga oceniać wnioski. Stawia ona następujące pytania:

1. Kto zgłosił (Raised) wniosek?
2. Jaka jest przyczyna (Reason) zmiany?
3. Jaki zwrot (Return) jest wymagany od zmiany?
4. Jakie są ryzyka (Risks) związane ze zmianą?
5. Jakie zasoby (Resources) są wymagane do dostarczenia zmiany?
6. Kto jest odpowiedzialny (Responsible) za budowanie, testowanie i implementację zmiany?
7. Jaki jest związek (Relationship) pomiędzy tą zmianą i innymi zmianami?

Odpowiedzi na te pytania stanowią ogólną bazę do oceny wniosku, którą można uzupełnić o informacje związane ze specyfiką przedsiębiorstwa, usługi lub samej zmiany. Precyzyjne oszacowanie wpływu jest trudne i często wymaga uwzględnienia stanowiska zainteresowanych grup, dlatego zarządzający zmianą może w każdej chwili poprosić Radę ds. Zmian o pomoc.

Posiadając komplet informacji dotyczący zmiany i wszystkich związanych z nią zjawisk nadchodzi moment, w którym należy podjąć decyzję o jej implementacji. W zależności od czynników takich jak typ, rozmiar, wpływ, czy ryzyko; zmiana jest kierowana do odpowiedniego organu autoryzującego. Najczęściej jest nim Rada ds. Zmian. Jednak przy małych zmianach organem tym może być sam zarządzający zmianą, natomiast przy zmianach o charakterze strategicznym może być to kierownictwo firmy. Organizacja zależy tutaj jedynie od zasad funkcjonowania przedsiębiorstwa.

Wszystkie zatwierdzone zmiany trafiają do Rady ds. Zmian, która jest odpowiedzialna za koordynowanie ich implementacji. Harmonogramowanie z wyprzedzeniem minimalizuje ryzyko awarii związane z jednoczesnym wprowadzaniem wielu zmian oraz, w przypadku potrzeby wyłączenia niektórych komponentów, pozwala na wcześniejsze poinformowanie użytkowników. Implementacją zmiany zajmuje się wskazany budowniczy. Przed przystąpieniem do prac musi on opracować plan pozwalający na ewentualne wycofanie zmiany. Procedura ta jest niezbędna biorąc pod uwagę ryzyko napotkania nieoczekiwanych problemów uniemożliwiających pełne wdrożenie. Po zaimplementowaniu zmiana jest sprawdzana przez testera, a następnie przekazywana do środowiska operacyjnego.

Ostatnim krokiem procesu zarządzania zmianą jest podsumowanie i zamknięcie zmiany. Zarządzający zmianą musi spisać przegląd powdrożeniowy[[69]](#footnote-69), w którym podaje ostateczny status zmiany oraz sprawdza, czy zmiana spełnia stawiane wymagania i pozwala osiągnąć oczekiwane rezultaty. Ponadto dokumentuje wszystkie powstałe efekty uboczne.

Powiązane procesy

Zarządzanie zmianą jest bardzo blisko powiązane z zarządzaniem komponentami usług i konfiguracją. Wszystkie autoryzowane zmiany stanowią podstawę do modyfikowania elementów konfiguracji i ocena zmiany wymaga dostępu do danych udostępnianych przez

Zarządzanie wdrożeniami i wersjami

Natomiast proces zarządzania problemami jest źródłem wielu wniosków o zmianę.

## Inne metodyki

1

# Firma RedHost

Co i dlaczego opis firmy. 0.5 + 4 = 4.5

## Charakterystyka

0.5 + 0.5 = 1

### Świadczone usługi

0.5

## Wprowadzenie zarządzania konfiguracją

0.5 + 1 = 1.5

### Przykładowe elementy konfiguracji

1

## Wprowadzenie zarządzania zmianą

0.5 + 1 = 1.5

### Przykładowe żądania zmian

1

# Wymagania

Przedstawiony zbiór wymagań systemowych stanowi podstawę do wykonania oprogramowania i określa jego zakres. Wszystkie zebrane wymagania zostały podzielone na grupy. Pierwsze dwie — technologia i architektura — zawierają ogólne wymogi dotyczące wykonania systemu, natomiast następne opisują poszczególne elementy systemu. Zarządzanie danymi należy rozumieć jako możliwość wykonywania operacji CRUD[[70]](#footnote-70).

## Technologia (TECH)

Ograniczenia technologii sprowadzają się do wyboru języka programowania i narzędzi. Celem jest stworzenie niezawodnego systemu wysokiej jakości, łatwego w obsłudze i taniego w utrzymaniu.

Spis wymagań niefunkcjonalnych

1. Zastosowanie sprawdzonych i powszechnie znanych technik programowania, aby zmniejszyć koszt i ułatwić utrzymanie systemu: paradygmatu obiektowego wraz ze wzorcami programowania opisanymi w (Gamma, Helm i Johnson) i (Fowler).
2. Umożliwienie dostosowania do specyficznych potrzeb klienta poprzez użycie paradygmatu programowania komponentowego, który pozwala na wymianę nawet najmniejszych elementów systemu.
3. Nacisk na enkapsulację poprzez wyodrębnienie funkcjonalności pojawiających się w różnych warstwach systemu (zarządzania transakcjami, autoryzacji i innych), dzięki wykorzystaniu paradygmatu programowania aspektowego.
4. System musi być napisany w języku o statycznym typowaniu, co pozwoli na wczesne wykrycie wielu błędów już na etapie kompilacji.
5. Ilość błędów podczas pracy systemu musi być minimalizowana, poprzez wykorzystanie testów jednostkowych.
6. System do przechowywania informacji musi wykorzystywać relacyjne bazy danych.

## Architektura (ARCH)

Spis wymagań niefunkcjonalnych

1. Wykonanie systemu w postaci trójwarstwowej aplikacji sieciowej z dostępem zdalnym.
2. Wsparcie wielu różnych warstw prezentacji.
   1. Podstawową warstwę prezentacji będzie przeglądarka www.
   2. Drugą stanowią usługi sieciowe[[71]](#footnote-71).
   3. Musi istnieć możliwość późniejszego zaimplementowania innych warstw prezentacji takich jak np. interfejs okienkowy.
3. System musi być zbudowany modularnie i być rozszerzalny.
   1. Rdzeń systemu złożony z wielu modułów będzie udostępniał usługi nie związane bezpośrednio z funkcjonalnością systemu, lecz wymagane przez inne moduły. Przykład: autoryzacja, dostarczanie danych o pracownikach firmy, wysyłanie komunikatów do użytkowników itp.
   2. Drugą grupę będą stanowić moduły odpowiadające procesom ITIL.
   3. Zaimplementowane zostaną procesy zarządzania konfiguracją i zmianą
      1. Procesy te muszą współpracować z zaimplementowanymi przez Piotra Kalańskiego: zarządzaniem incydentami i zmianą.
   4. Musi istnieć możliwość późniejszej implementacji kolejnych procesów ITIL.
   5. Pożądana jest możliwość rozszerzenia systemu o moduły zapewniające funkcjonalność SCM[[72]](#footnote-72), CRM[[73]](#footnote-73) i ERP[[74]](#footnote-74).

## Moduł zarządzania konfiguracją (KONF)

Zadaniem modułu jest zarządzanie CMDB, czyli przechowywanie danych konfiguracji firmy zgodnie z procesem zarządzania konfiguracją ITIL w wersji 3. Zawartość CMDB stanowią informacje o elementach konfiguracji, ich atrybutach i relacjach między nimi. Elementy konfiguracji dzielą się na dwie zasadnicze grupy: elementy wbudowane i uniwersalne.

Do pierwszej należą elementy konfiguracji zarządzanie przez inne moduły takie jak zgłoszenia incydentów, problemów, żądania zmian, ale także pracownicy i klienci. Elementy te są wyświetlane w module konfiguracji w trybie do odczytu. Za edycję elementów wbudowanych odpowiedzialne są ich macierzyste moduły, co pozwala na egzekwowanie odpowiednich uprawnień i ograniczeń wewnątrz tych modułów.

Drugą grupę stanowią elementy uniwersalne posiadające dowolny zestaw atrybutów, określany przez użytkownika. Ich zadaniem jest przechowywanie informacji o elementach konfiguracji specyficznych dla danej branży. Przykładami są: monitory, serwery, elementy infrastruktury, komponenty niezbędne do funkcjonowania formy.

Spis wymagań funkcjonalnych

1. Przechowywanie, wyszukiwanie i prezentacja danych wszystkich elementów konfiguracji (zarówno wbudowanych jak i uniwersalnych).
   1. Element konfiguracji posiada przypisany identyfikator oraz zbiór wartości atrybutów.
   2. Format identyfikatora jest ustalany przez firmę (np. dla monitora LCD zakupionego w roku 2010 roku: HW/MON/LCD/4/30/2010/8).
   3. Dodatkowo element posiada krótkie podsumowanie złożone z wybranych wartości atrybutów, dzięki któremu znacznie łatwiej odnaleźć szukany element na liście (np. dla tego samego monitora: Samsung SME1920N 19" 1360x768 TFT/TN).
2. Zarządzanie uniwersalnymi elementami konfiguracji.
   1. Uniwersalny element konfiguracji posiada dodatkowo typ. Typ podawany jest podczas tworzenia elementu konfiguracji i nie może być zmieniony.
3. Zarządzanie typami elementów uniwersalnych.
   1. Typ elementu uniwersalnego posiada unikalną nazwę, opis oraz zestaw atrybutów.
   2. Typ może być abstrakcyjny, co oznacza, że nie można stworzyć elementu konfiguracji tego typu.
   3. Typy tworzą hierarchię. Podtypy dziedziczą atrybuty typu nadrzędnego.
   4. Kluczowym wymaganiem jest, aby można było dodawać nowe typy z poziomu interfejsu użytkownika, bez potrzeby modyfikacji systemu.
4. Zarządzanie relacjami między elementami konfiguracji.
   1. Relacja łączy dwa elementy konfiguracji (wbudowane lub uniwersalne). Ponadto tak jak w przypadku uniwersalnych elementów konfiguracji posiada typ oraz zbiór wartości atrybutów.
5. Zarządzanie typami relacji. Wszystkie wymagania dotyczące typów uniwersalnych elementów konfiguracji odnoszą się także do typów relacji. A ponadto klasa relacji:
   1. określa czy relacja jest skierowana, czy nie;
   2. wyznacza podpis relacji w obu kierunkach (np. obsługuje, jest obsługiwany przez);
   3. nakłada ograniczenia na typy elementów obu stron relacji.
6. Zarządzanie atrybutami typów elementów uniwersalnych i relacji.
   1. Atrybut posiada nazwę, opis oraz swój format.
      1. W obecnej wersji jedynym dostępnym formatem atrybutu jest napis. W kolejnych wersjach będą dodawane nowe formaty.
7. Zarządzanie wartościami atrybutów elementów uniwersalnych i relacji.
   1. Wartości atrybutów to pary typu atrybut-wartość. Pierwszy element pary musi być atrybutem należącym do typu elementu lub relacji, natomiast drugi musi posiadać format określony przez dany atrybut.

Spis wymagań niefunkcjonalnych

1. Zarządzać elementami konfiguracji i relacjami oraz wartościami ich atrybutów mogą jedynie uprawnieni zarządzający konfiguracją
2. Zarządzać typami elementów konfiguracji i relacji oraz ich atrybutami mogą jedynie uprawnieni administratorzy konfiguracji.
3. System musi zapewnić miejsce do przechowywania stu tysięcy elementów konfiguracji. Elementy testowe będą posiadać dwadzieścia wartości atrybutów o średniej długości dwustu bajtów.
4. System musi zapewnić miejsce do przechowywania stu typów elementów uniwersalnych i stu typów relacji. Typy testowe będą posiadać dwadzieścia atrybutów o średniej długości wynoszącej jeden kilobajt, ponadto ich nazwa i opis nie przekroczą dwóch kilobajtów.
5. Przy bazie danych zapełnionej w ilości wskazanej w punktach 3. i 4. system powinien odpowiadać na wszystkie zapytania w czasie poniżej pięciu sekund.

## Moduł zarządzania zmianą (ZMIA)

Moduł służy do przechowywania i przetwarzania żądań zmian (RFC) zgodnie z ITIL. Ocenione i autoryzowane zmiany są podstawą do modyfikacji CMDB przy użyciu modułu konfiguracji.

Spis wymagań funkcjonalnych

1. Przyjmowanie nowych żądań zmian.
   1. Zgłaszane nowe żądania zmian posiadają unikalny identyfikator, krótki tytuł, dokładny opis, wstępny priorytet, wstępną kategorię oraz odnotowane: autora i czas złożenia.
   2. Żądanie stanowi element konfiguracji i może być powiązane różnymi relacjami z innymi elementami.
   3. Ponadto żądanie może posiadać komentarze.
   4. Użytkownik może usuwać swoje nowe żądania zmian.
   5. Zarządzający konfiguracją może przejąć nowe żądanie zmiany, zmieniając jego stan na nieautoryzowane żądanie zmiany.
2. Zarządzanie nieautoryzowanymi żądaniami zmian.
   1. Nieautoryzowane żądania zmian mogą być filtrowane (odrzucane) przez kierującego nimi zarządzającego zmianą po podaniu przyczyny.
   2. Zarządzający zmianą może zmienić priorytet i kategorię ustawioną przez autora żądania.
   3. Zarządzający określa typ zmiany (np. rutynowa, mała, ważna, duża).
   4. Następnie żądanie jest wysyłane do organu autoryzującego zmiany. Organ może różnić się w zależności od kategorii, priorytetu (wraz z pilnością) i typu.
   5. Organ autoryzujący żądanie autoryzuje bądź odrzuca podając przyczynę.
      1. Przy pozytywnym rozpatrzeniu może opcjonalnie podać proponowany czas implementacji zmiany.
3. Zarządzanie autoryzowanymi zmianami.
   1. Wszystkie autoryzowane zmiany trafiają do powiązanej z ich kategorią rady ds. zmian lub w przypadku zmian pilnych — nadzwyczajnej rady ds. zmian.
   2. Rada może zmiany odrzucić lub zmienić ich priorytet i typ.
   3. Rada wpisuje zmiany do harmonogramu zmian oraz określa osobę odpowiedzialną za budowanie i testowanie zmiany. Stan żądania zmiany zmienia się na implementowane.
4. Zarządzanie implementowanym żądaniami zmianami.
   1. Budowniczy po wykonaniu prac zmienia status żądania na gotowe do wdrożenia.
   2. Zarządzający zmianą koordynuje wdrożenie i odnotowuje jego wynik w żądaniu zmiany.
   3. W przypadku porażki zmiana jest wycofywana i przekazywana z powrotem do rady ds. zmian.
   4. Natomiast w przypadku sukcesu zarządzający zmianą dołącza do żądania sprawozdanie z implementacji i zamyka żądanie.
5. Zarządzanie odrzuconymi żądaniami zmian.
   1. Dowolny użytkownik może zgłosić apelację od odrzucenia żądania po podaniu odpowiedniego uzasadnienia.
   2. Apelacja może zostać uznana przez zarządzającego zmianą.
   3. Zmiany, których apelacja została uznana, stają się zmianami nieautoryzowanymi.
6. Wyświetlanie harmonogramu zmian.
7. Zarządzanie poziomami priorytetów zmian.
   1. Priorytet ma unikalną nazwę.
   2. Niektóre poziomy mogą być oznaczone jako pilne. Zmiany o priorytecie oznaczonym jako pilny trafiają do nadzwyczajnej rady ds. zmian.
8. Zarządzanie kategoriami żądań zmian.
   1. Kategoria posiada unikalną nazwę.
   2. Kategorie tworzą strukturę hierarchiczną.
   3. Kategoria może posiadać przypisaną zwykłą i nadzwyczajną radę ds. zmian, do której kierowane są wszystkie autoryzowane zmiany danej kategorii. W przeciwnym przypadku zmiany te są wysyłane do rady związanej z kategorią nadrzędną.
9. Zarządzanie typami zmian.
   1. Typ zmiany posiada unikalną nazwę.
10. Zarządzanie organami autoryzującymi.
    1. Organ autoryzujący posiada unikalną nazwę oraz może składać się z jednego lub więcej użytkowników systemu.
    2. Organ otrzymuje zmiany wymagające autoryzacji określonego typu z określonej kategorii.
    3. Szczególnymi rodzajami organu autoryzującego jest Rada ds. Zmian (CAB) i nadzwyczajna Rada ds. Zmian (CAB/EC).

Spis wymagań niefunkcjonalnych

1. Żądania zmian mogą być składane tylko przez uprawnionych użytkowników.
2. Żądania zmian mogą być zarządzane tylko przez uprawnionych zarządzających zmianą.
3. Poziomy priorytetów, kategorie zmian i zakresy wpływu mogą być zarządzane tylko przez uprawnionych administratorów zmian.
4. System musi przechowywać nie mniej niż sto tysięcy wniosków zmian.
5. Czas odpowiedzi systemu na zapytania przy takim obciążeniu nie powinien przekroczyć trzech sekund.

## Rdzeń systemu (CORE)

Zbiór modułów świadczących szeroki wachlarz usług, nie związany bezpośrednio z procesami ITIL, ale uzupełniający je o niezbędną z punktu widzenia użytkownika funkcjonalność.

Spis wymagań funkcjonalnych

1. Uwierzytelnienie tożsamości i logowanie użytkowników do systemu.
   1. Użytkownicy mogą być pracownikami, klientami lub kooperantami.
   2. Uwierzytelnianie bazuje na nazwie użytkownika i haśle.
   3. Każdy użytkownik posiada swój zestaw uprawnień.
   4. Bez odpowiedniej autoryzacji nie można korzystać z systemu.
   5. Po zakończonej pracy użytkownicy mogą się wylogować.
2. Udostępnienie narzędzi komunikacji między użytkownikami
   1. Wysyłanie ogłoszeń do wszystkich użytkowników.
   2. Wysyłanie wiadomości do wskazanych użytkowników.
   3. Przypisywanie użytkownikom zleceń na wykonanie zadania.
      1. Powiadamianie o przebiegu wykonania tych zleceń.
3. Zapisywanie i przechowywanie dowolnych plików w systemie.
   1. Możliwość dodawania plików do wiadomości w formie załączników.
4. Udostępnianie informacji na temat firmy.
   1. Danych pracowników.
   2. Danych klientów.
   3. Danych kooperantów.

Spis wymagań niefunkcjonalnych

1. System musi zapewnić miejsce do przechowywania danych o co najmniej dziesięciu tysiącach użytkowników.
2. System musi mieć możliwość przechowywania łącznie co najmniej miliona ogłoszeń, wiadomości i zleceń.
3. Ilość przechowywanych plików powinna być ograniczona jedynie ilością wolnego miejsca na dyskach twardych serwera.

# Projekt

# Implementacja

# Użytkowanie systemu

# Testy i ocena wydajności

# Podsumowanie

Bibliografia

Grudzień 2010. Global Journal of Computer Science and Technology. 18 Lipiec 2011 <http://computerresearch.org/stpr/index.php/gjcst/article/download/465/424>.

brak daty. Lipiec 2010 <http://www.actis-ingenierie.com>.

An Introductory Overview of ITIL® V3. The UK Chapter of the itSMF, 2007.

Erich Gamma i inni. Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku. Wydawnictwa Naukowo Techniczne, 2005.

Henry Fowler. Architektura systemów zarządzania przedsiębiorstwem. Wzorce projektowe. Helion, 2005.

IT frameworks wiki - Hasło ITIL. brak daty. 7 Lipiec 2011 <http://www.itframeworks.org/wiki/Information\_Technology\_Infrastructure\_Library>.

IT Life - Co to jest ITIL? brak daty. 7 Lipiec 2011 <http://itsm.itlife.pl/>.

„IT Life - Korzenie ITIL.” brak daty. IT Life. 18 Lipiec 2011 <http://itsm.itlife.pl/content/view/10012/57/>.

ITIL Service Management - A Brief History of ITIL. 18 Lipca 2011. 18 Lipca 2011 <http://itservicemngmt.blogspot.com/2007/09/brief-history-of-itil.html>.

„ITILv3 Glossary Polish.” 2008. ITIL Official Site. 10 Grudnia 2010 <http://www.itil-officialsite.com/nmsruntime/saveasdialog.aspx?lID=921&sID=242>.

Noel Scott. Case Study: Using ITIL® and PRINCE2™ Together. The Stationery Office, 2010.

OGC. Service Operation Book (ITIL). The Stationery Office, 2007.

—. Service Strategy Book (ITIL). The Stationery Office, 2007.

Pink Elephant - IT Service Management 2007. brak daty. 18 Lipiec 2011 <http://www.pinkelephant.com/articles/MicrosoftPowerPointDavidRatcliffePinkPerspectiveBM.pdf>.

Piotr Kalański. System Service Desk zgodny z zaleceniami ITIL. 2010.

„Projekt ITIL Refresh.” brak daty. ITLife.pl. 19 Lipiec 2011 <http://itsm.itlife.pl/content/view/10071/138/>.

Tom Sante i Ermers Jeroen. TOGAF™ 9 and ITIL® V3 Two Frameworks Whitepaper. 2009.

Zawartość płyty CD

Wykaz skrótów

Spis Tabel

Spis ilustracji

Ilustracja 2.1. Cykl życia usługi w ITIL 19

Ilustracja 2.2 Procesy ITIL wg hierarchii zarządzania (źródło: http://www.itframeworks.org) 20

1. Office of Government Commerce — komórka rządu Wielkiej Brytanii. W momencie stworzenia ITIL działająca pod nazwą Central Computer and Telecommunications Agency (CCTA). [↑](#footnote-ref-1)
2. (Kalański) [↑](#footnote-ref-2)
3. (ITILv3 Glossary Polish) [↑](#footnote-ref-3)
4. (Global Journal of Computer Science and Technology) [↑](#footnote-ref-4)
5. (ITIL Service Management - A Brief History of ITIL) [↑](#footnote-ref-5)
6. (IT Life - Korzenie ITIL) [↑](#footnote-ref-6)
7. (Sante i Jeroen) [↑](#footnote-ref-7)
8. (Pink Elephant - IT Service Management 2007). [↑](#footnote-ref-8)
9. ang. Service Level Agreement — umowa określająca warunki świadczenia usługi, w szczególności mierzalne parametry dotyczące wydajności i awaryjności. [↑](#footnote-ref-9)
10. źródło: (actis-ingenierie) [↑](#footnote-ref-10)
11. (Projekt ITIL Refresh) [↑](#footnote-ref-11)
12. źródło: (IT Life - Co to jest ITIL?) [↑](#footnote-ref-12)
13. ang. Service Portfolio [↑](#footnote-ref-13)
14. źródło: (IT frameworks wiki - Hasło ITIL) [↑](#footnote-ref-14)
15. (OGC 31) [↑](#footnote-ref-15)
16. ang. Service Catalogue [↑](#footnote-ref-16)
17. ang. Service Catalogue Management [↑](#footnote-ref-17)
18. ang. Service Level Management [↑](#footnote-ref-18)
19. ang. Service Level Requirement — dokument zawierający biznesowe wymagania dotyczące usługi [↑](#footnote-ref-19)
20. (An Introductory Overview of ITIL® V3) [↑](#footnote-ref-20)
21. ang. Information Security Management [↑](#footnote-ref-21)
22. ang. IT Service Continuity Management (ITSCM) [↑](#footnote-ref-22)
23. ang. Supplier Managment [↑](#footnote-ref-23)
24. ang. Service Design Package (SDP) [↑](#footnote-ref-24)
25. ang. Service Acceptance Criteria (SAC) [↑](#footnote-ref-25)
26. ang. Transition Planning and Support [↑](#footnote-ref-26)
27. ang. PRojects IN Controlled Environments 2, (Scott, 2010) [↑](#footnote-ref-27)
28. ang. Service Validation and Testing [↑](#footnote-ref-28)
29. ang. Evaluation [↑](#footnote-ref-29)
30. ang. Release and Deployment Management [↑](#footnote-ref-30)
31. ang. Release Package [↑](#footnote-ref-31)
32. ang. Early Life Support (ELS) [↑](#footnote-ref-32)
33. ang. Request for Change [↑](#footnote-ref-33)
34. ang. Change Management [↑](#footnote-ref-34)
35. ang. Zarządzanie komponentami usług i konfiguracją (SACM) [↑](#footnote-ref-35)
36. ang. Configuration Management Database [↑](#footnote-ref-36)
37. ang. Configuration Item (CI) [↑](#footnote-ref-37)
38. ang. Knowledge Management [↑](#footnote-ref-38)
39. ang. Event Management [↑](#footnote-ref-39)
40. ang. Event [↑](#footnote-ref-40)
41. (OGC) [↑](#footnote-ref-41)
42. ang. Incident [↑](#footnote-ref-42)
43. ang. Incident Management [↑](#footnote-ref-43)
44. ang. Problem [↑](#footnote-ref-44)
45. ang. Problem Management [↑](#footnote-ref-45)
46. ang. Known Error Record (KER) [↑](#footnote-ref-46)
47. ang. Known Error Database (KEDB) [↑](#footnote-ref-47)
48. ang. Access Management [↑](#footnote-ref-48)
49. ang. Service Request [↑](#footnote-ref-49)
50. ang. Request Fulfilment [↑](#footnote-ref-50)
51. źródło: (OGC) [↑](#footnote-ref-51)
52. ang. Service Measurement [↑](#footnote-ref-52)
53. ang. Service Reporting [↑](#footnote-ref-53)
54. mimo takich samych nazw należy rozróżnić proces CSI od obszaru CSI [↑](#footnote-ref-54)
55. źródło: (OGC) [↑](#footnote-ref-55)
56. (Kalański) [↑](#footnote-ref-56)
57. ang. Configuration Management Database [↑](#footnote-ref-57)
58. ang. Configuration Model [↑](#footnote-ref-58)
59. ang. Configuration Management System (CMS) [↑](#footnote-ref-59)
60. ang. Configuration Manager [↑](#footnote-ref-60)
61. źródło: (Wikipedia) [↑](#footnote-ref-61)
62. ang. Request For Change [↑](#footnote-ref-62)
63. ang. urgent [↑](#footnote-ref-63)
64. ang. Change Record [↑](#footnote-ref-64)
65. ang. Change Manager [↑](#footnote-ref-65)
66. ang. Change Advisory Board [↑](#footnote-ref-66)
67. ang. Emergency Advisory Board [↑](#footnote-ref-67)
68. (OGC) [↑](#footnote-ref-68)
69. ang. Post Implementation Review (PIR) [↑](#footnote-ref-69)
70. ang. Create, Read, Update, Delete. Operacje pozwalające dodawać, odczytywać, uaktualniać i usuwać dane. [↑](#footnote-ref-70)
71. ang. web services [↑](#footnote-ref-71)
72. ang. Supply chain management — Zarządzanie łańcuchem dostaw [↑](#footnote-ref-72)
73. ang. Customer relationship management — Zarządzanie relacjami z klientami [↑](#footnote-ref-73)
74. ang. Enterprise resource planning — Planowanie zasobów przedsiębiorstwa [↑](#footnote-ref-74)