|  |
| --- |
| **Wyższa Szkoła Technologii Informatycznych w Katowicach** |
| Wydział Informatyki Kierunek Informatyka |
| **Adrian Leś**  Nr albumu: 34150  Studia stacjonarne |
| **Temat pracy**  System wspomagający proces komunikacji między studentami a pracownikami dziekanatu |
| Praca dyplomowa inżynierska  Napisana pod kierunkiem  dr inż. Romana Simińskiego  w roku akademickim 2023/2024 |
| Katowice 2024 |

Ta strona powinna być pusta. Uwaga, ten tekst jest ukryty i nie będzie widoczny na wydruku.

Spis treści

[1 Wstęp 3](#_Toc157682486)

[2 Analiza problemu 4](#_Toc157682487)

[3 Analiza istniejących rozwiązań 9](#_Toc157682488)

[3.1 Kryteria analizy porównawczej 9](#_Toc157682489)

[3.2 Moduł podań w systemie USOS UŚ 9](#_Toc157682490)

[3.3 System obiegu dokumentów EZD RP 11](#_Toc157682491)

[3.4 Platforma Zendesk z licencją dla sektora edukacji 12](#_Toc157682492)

[3.5 Platforma Slack dostosowana do środowiska edukacyjnego 13](#_Toc157682493)

[3.6 Podsumowanie 14](#_Toc157682494)

[4 Koncepcja własnego rozwiązania 15](#_Toc157682495)

[4.1 Koncepcja rozwiązania użytkowego 16](#_Toc157682496)

[4.2 Koncepcja rozwiązania technologicznego 16](#_Toc157682497)

[5 Projekt ogólny 18](#_Toc157682498)

[5.1 Specyfikacja wymagań funkcjonalnych i niefunkcjonalnych 18](#_Toc157682499)

[5.2 Architektura systemu 18](#_Toc157682500)

[5.3 Metody i narzędzia realizacji 19](#_Toc157682501)

[5.4 Koncepcja przechowywania danych 19](#_Toc157682502)

[5.5 Projekt interfejsu użytkownika 20](#_Toc157682503)

[6 Dokumentacja techniczna 20](#_Toc157682504)

[7 Testy i weryfikacja systemu 21](#_Toc157682505)

[8 Przykładowy scenariusz wykorzystania systemu 21](#_Toc157682506)

[9 Zakończenie 21](#_Toc157682507)

[10 Bibliografia 21](#_Toc157682508)

[11 Spis rysunków 22](#_Toc157682509)

[12 Spis tabel 22](#_Toc157682510)

Ta strona powinna być pusta. Uwaga, ten tekst jest ukryty i nie będzie widoczny na wydruku.

# Wstęp

Popularyzacja usług teleinformatycznych przyniosła znaczne udogodnienia w sferze życia zawodowego, umożliwiając pracę zdalną oraz elektroniczne składanie wniosków. Niemniej jednak, pomimo rosnącej popularności tych innowacji, wiele instytucji nadal utrzymuje się w cyfrowej stagnacji. Doskonałym przykładem takiej opieszałości jest dziekanat studencki, który wciąż polega głównie na tradycyjnych dokumentach papierowych, dostarczanych osobiście do siedziby instytucji. W obliczu dzisiejszych możliwości realizacji większości spraw drogą internetową, taki sposób wymiany informacji, komunikatów, dokumentów i wniosków staje się nie tylko niewygodny, lecz również czasochłonny zarówno dla studentów, jak i personelu uczelni. Przy obecnej dostępności nowoczesnych narzędzi, takie podejście wydaje się anachroniczne i ogranicza efektywność działań administracyjnych.

Celem pracy jest zaprojektowanie i realizacja systemu wspomagania procesu komunikacji pomiędzy studentem a dziekanatem umożliwiającego wymianę informacji, wiadomości, wniosków w sposób bezkontaktowy. System ten umożliwi studentom składanie i odbieranie komunikatów, wniosków oraz dokumentów w formie elektronicznej poprzez zgłoszenia. Planuje się, że system umożliwi również efektywne zarządzanie dokumentami przez personel uczelni, z dostępem do nadesłanych, rozpatrzonych i zarchiwizowanych dokumentów. Dzięki odpowiedniemu grupowaniu i filtrowaniu dokumentów, proces rozpatrzenia wniosków stanie się bardziej efektywny. Zastosowanie tego systemu przyczyni się do skrócenia czasu całej procedury składania wniosków zarówno dla pracowników uczelni, jak i studentów. System ten zostanie zrealizowany w formie aplikacji internetowej, korzystając z nowoczesnych technologii, takich jak języki programowania Java i TypeScript, szablony Spring Boot, Angular, oraz silnik baz danych MSSQL.

Rozdział drugi stanowi analizę głównego problemu, oferując zdefiniowanie jego kluczowych aspektów oraz ukazując kontekst i istotę wyzwania. W trzecim rozdziale dokonano pełnego przeglądu istniejących rozwiązań, łącząc go z krytycznym porównaniem do naszego projektowanego systemu. Koncepcje rozwiązania problemu, autorstwa inżyniera, zostały przedstawione w rozdziale czwartym, podczas gdy piąty rozdział poświęcony jest omówieniu ogólnej koncepcji organizacji systemu. W szóstym rozdziale odnajdziemy obszerną dokumentację techniczną projektu, zawierającą szczegółowe informacje na temat zastosowanych technologii, narzędzi, struktury kodu, sposobów testowania oraz innych aspektów technicznych. Proces testowania i weryfikacji systemu pod kątem funkcjonalności został precyzyjnie opisany w siódmym rozdziale. Rozdział ósmy skupia się na prezentacji realnych scenariuszy i przypadków wykorzystania systemu, szczegółowo opisując kroki, interakcje i korzyści wynikające z implementacji opracowanego rozwiązania. Ostatni rozdział stanowi syntezę osiągnięć projektu, jednocześnie proponując perspektywy dalszego rozwoju.

# Analiza problemu

Podczas analizy potencjalnych problemów związanych z obecną formą wymiany informacji między studentami a pracownikami dziekanatu, autor zidentyfikował dwa potencjalne punkty inicjacji komunikacji. Student może zainicjować komunikację, gdy istnieje potrzeba uzyskania odpowiedzi na kwestie administracyjne, złożenia wniosku lub dostarczenia dokumentów. Założono, że jest to rodzaj komunikacji jeden do jednego, w której uczestniczą student i pracownik dziekanatu. Poza tym scenariuszem autor rozważył również możliwość, w której inicjatorem komunikacji będzie pracownik dziekanatu. Analizując ten przypadek, podzielono go na dwie potencjalne kategorie komunikacji. Pierwszą kategorią jest komunikacja jeden do jednego, podobnie jak w przypadku komunikacji inicjowanej przez studenta. Przykładem może być sytuacja, gdy od konkretnego studenta wymagane jest dostarczenie brakującego wniosku. Poza tym rodzajem komunikacji pojawić się może również komunikacja jeden do wielu, na przykład w formie rozsyłania informacji o konieczności wypełnienia obowiązkowych badań ankietowych przez określoną grupę studentów. Analizując obie możliwe ścieżki komunikacji - zarówno inicjowaną przez studenta, jak i pracownika dziekanatu - zauważamy wspólne problemy, ale także te, które występują tylko w jednym z tych przypadków.

W trakcie analizy problemów wynikających z komunikacji między studentem a pracownikiem dziekanatu, szczególnie zauważalnym aspektem jest przepływ dokumentów papierowych, który przedstawia Rysunek 1. Kwestia ta staje się problematyczna w sytuacjach, gdy od studenta wymaga się wypełnienia i dostarczenia określonych dokumentów w celu rozpoczęcia konkretnego procesu, na przykład wniosku o przeniesienie z formy studiów stacjonarnych na niestacjonarne, wzór wniosku przedstawiaRysunek2. Proces ten nakłada szereg trudności na studenta, zaczynając od ograniczonego czasu, szczególnie dla tych, którzy mieszkają poza obszarem uczelni lub są zatrudnieni. Konieczność osobistego dostarczenia dokumentów może prowadzić do konieczności wzięcia urlopu lub przerwania innych zobowiązań, co stwarza dodatkowe utrudnienia. Ponadto, koszty podróży związane z koniecznością odwiedzenia dziekanatu mogą być istotnym obciążeniem dla studentów, zwłaszcza tych, którzy mieszkają daleko od uczelni. Dodatkowe wydatki związane z transportem publicznym, paliwem czy parkowaniem tylko potęgują trudności. To nie tylko sprawia trudności dla studentów, ale także generuje dodatkowe obciążenia dla pracowników instytucji. Po dostarczeniu dokumentów, proces weryfikacji i udzielania odpowiedzi jest obecnie nieustrukturyzowany. Brak standaryzacji w przekazywaniu informacji zwrotnej, często opierającej się na rozmowach telefonicznych lub wiadomościach e-mail, może prowadzić do niejednoznaczności i kolejnych iteracji komunikacji między studentem a pracownikiem. Kolejnym aspektem jest problem archiwizacji i przechowywania dokumentów papierowych, co może wprowadzić nieefektywności zarówno w organizacji, jak i wydajności procesów. Zajmowanie przestrzeni fizycznej przez papierowe dokumenty oraz długotrwałe wyszukiwanie konkretnych dokumentów w archiwum stwarzają znaczące wyzwania. Podsumowując, istnieje potrzeba przemyślenia i ulepszenia procesu komunikacji oraz przepływu dokumentów, aby zminimalizować trudności zarówno dla studentów, jak i pracowników instytucji. Wprowadzenie bardziej efektywnych i zautomatyzowanych rozwiązań może znacząco poprawić tę sytuację.

Analiza zagadnień związanych z inicjacją komunikacji przez pracowników dziekanatu uwzględnia różne aspekty, szczególnie ilość odbiorców komunikatu w kontekście pojedynczego studenta lub grupy studentów. W przypadku komunikacji jednostronnej, gdzie pracownik instytucji pełni rolę nadawcy, a student jest odbiorcą, identyfikuje się szereg barier podobnych do tych obserwowanych w odwrotnym przepływie informacji, tj. od studenta do pracownika. Przykładowe trudności obejmują brak standardowego systemu przekazywania informacji dotyczących komunikatów, wydłużony czas niezbędnyo archiwizacji oraz przeszukiwania dokumentów.



Rysunek 1Aktualny obieg dokumentów w dziekanacie studenckim



Rysunek 2Wniosek o przeniesienie z formy studiów stacjonarnych na niestacjonarne

Źródło: Przeniesienie (studia stacjonarne i niestacjonarne), <https://us.edu.pl/wydzial/wnst/studia/student/regulaminy/>, 11.10.21.

W kontekście omawianej ścieżki procesu pojawia się problem związanego z niepewnością pracownika co do faktycznego otrzymania i zapoznania się przez studenta z przekazanym komunikatem. W przypadku indywidualnej korespondencji, gdzie pracownik instytucji pełni rolę nadawcy, a student jest odbiorcą, jedynym efektywnym środkiem eliminującym nieścisłości jest bezpośrednia rozmowa telefoniczna. W takim przypadku pracownik ma pewność, że przekazał informacje klarownie, a także ma możliwość udzielenia dodatkowych wyjaśnień oraz odpowiedzi na ewentualne pytania odbiorcy. Natomiast w sytuacji, gdy komunikacja odbywa się za pośrednictwem formy pisemnej, takiej jak wiadomość e-mail czy korespondencja, pracownik nie ma pewności, czy informacje zostały poprawnie przekazane, dopóki odbiorca nie udzieli informacji zwrotnej potwierdzającej zapoznanie się z treścią komunikatu.

W kontekście omówionych wyzwań, planuje się wdrożenie systemu wspomagającego komunikację w postaci aplikacji internetowej. Wybór tego rozwiązania wynika z jego zdolności do obsługi zgłoszeń z dowolnego miejsca i o każdej porze dnia. Aplikacja internetowa stanowi efektywne narzędzie, które pozwoli studentom inicjować i monitorować procesy administracyjne bez konieczności fizycznego odwiedzania dziekanatu. Dzięki temu rozwiązaniu, ograniczenia czasowe związane z dostarczaniem dokumentów czy uzyskiwaniem odpowiedzi zostaną zredukowane, co wpłynie pozytywnie na wygodę studentów. Ponadto, aplikacja internetowa eliminuje konieczność ponoszenia kosztów związanych z podróżami do dziekanatu, co stanowi kolejny korzyść dla studentów. Dzięki dostępowi do zgłoszeń i informacji zwrotnych, proces komunikacji staje się bardziej przejrzysty i dostępny.

System ten umożliwi także pracownikom instytucji bardziej efektywne zarządzanie zgłoszeniami, co przyczyni się do usprawnienia procesów administracyjnych. Standaryzacja przekazywania informacji zwrotnych poprzez aplikację internetową pozwoli uniknąć niejednoznaczności i błędów wynikających z tradycyjnych form komunikacji.

Wprowadzenie aplikacji internetowej stanowi krok ku usprawnieniu całego procesu komunikacji między studentami a pracownikami dziekanatu, zwiększając jednocześnie dostępność i efektywność obsługi administracyjnej.

# Analiza istniejących rozwiązań

W rozdziale tym przedstawione zostaną rozwiązania, takie jak platformy, systemy czy moduły, które są już dostępne na rynku i mogą przyczynić się do rozwiązania problemów opisanych w rozdziale 2. Informacje na temat tych rozwiązań zostały pozyskane ze stron internetowych producentów lub dystrybutorów oprogramowania. Analiza będzie oparta na dostępnych danych dotyczących poszczególnych rozwiązań, umożliwiając pełniejsze zrozumienie ich potencjalnych korzyści i zastosowań.

## Kryteria analizy porównawczej

Analiza obejmować będzie następujące elementy, uwzględniające różnorodne kryteria w celu dokładnej oceny wad i zalet dostępnych rozwiązań. Kluczowe kryteria analizy porównawczej obejmują obecną dostępność funkcji komunikacyjnych, takich jak czaty i wiadomości prywatne. Ponadto, skupimy się na intuicyjności interfejsu użytkownika oraz ogólnej łatwości użycia, zwłaszcza dla różnych grup docelowych, takich jak studenci i pracownicy dziekanatu. Analizowane będą również możliwości związane z funkcjonalnością współpracy, w tym współdzieleniem plików i udzielaniem komentarzy. Wszystkie te kryteria umożliwią kompleksową ocenę potencjalnych wad i zalet każdego z rozwiązań.

## Moduł podań w systemie USOS UŚ

Moduł podań stanowi integralną część systemu USOS UŚ, umożliwiając składanie określonych rodzajów podań w formie zdalnej. Składa się on z różnych elementów, z których głównym jest katalog, przedstawiony na rysunku 3, zawierający sprawy rozstrzygnięte lub aktualnie rozstrzygane, a także umożliwiający złożenie nowego podania. Proces składania nowego wniosku z katalogiem dostępnych dokumentów, jak przedstawiono na rysunku 4, staje się widoczny po naciśnięciu przycisku "Złóż nowe podanie" z okna głównego modułu. Po wyborze wniosku i naciśnięciu przycisku "Wypełnij podanie" użytkownik zostaje przeniesiony do okna wypełniania podania z instrukcją dotyczącą jego kompleksowego uzupełnienia.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie

Rysunek 3 Moduł podań systemu USOS UŚ

Obraz zawierający tekst, elektronika, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 4 Składanie nowego podania w systemie USOS UŚ z listy określonych podań

Moduł podań w ramach systemu USOS UŚ nie obejmuje funkcji komunikacyjnych i współpracy; brak możliwości wymiany wiadomości prywatnych, uruchomienia czatu na określony temat, wymiany plików czy dodania komentarzy. Moduł ten skupia się jedynie na umożliwieniu wypełnienia podania i prezentuje ostateczną odpowiedź na zgłoszenie. Mimo braku zaawansowanych funkcji komunikacyjnych, moduł charakteryzuje się spójnym interfejsem, klarowną nawigacją i czytelnym przedstawianiem informacji. Jego głównym celem jest efektywne umożliwienie składania podań i prezentacja końcowego stanu odpowiedzi.

## System obiegu dokumentów EZD RP

System elektronicznego zarządzania dokumentacją EZD RP umożliwia szereg funkcji, w tym prowadzenie korespondencji w formie elektronicznej, wspierając tym samym efektywną komunikację wewnętrzną jednostki. Zgodnie z dostępnym opisem na stronie producenta, system ten został specjalnie opracowany, aby sprostać potrzebom biznesowym polskiej administracji publicznej. EZD RP nie tylko umożliwia elektroniczne rozstrzyganie spraw, ale również wspiera obsługę spraw w formie papierowej. Oprogramowanie odzwierciedla kluczowe procesy wykonywane przez urzędników podczas załatwiania spraw. EZD RP wyróżnia się także wspieraniem efektywnego obiegu dokumentów za pomocą systemu zadań. Dzięki tej funkcji, czas obiegu dokumentów jest redukowany, co z kolei przekłada się na zmniejszenie liczby koniecznej korespondencji. To wszystko wspólnie sprawia, że EZD RP stanowi kompleksowe narzędzie wspomagające zarządzanie dokumentacją oraz procesami administracyjnymi w kontekście polskiej administracji publicznej.

EZD RP zapewnia możliwość prowadzenia korespondencji w formie elektronicznej, w pełni spełniając wymagania związane z funkcjami komunikacyjnymi. System doskonale wspiera obieg elektronicznych dokumentów, umożliwiając dostęp do nich dla osób uczestniczących w procesie obiegu. W ten sposób efektywnie spełnia również wymagania dotyczące współpracy nad dokumentami. Jednolity i intuicyjny interfejs systemu, przedstawiony na rysunku 5, znacznie ułatwia korzystanie z jego funkcji, co przyczynia się do wygodnego i efektywnego zarządzania dokumentacją elektroniczną.



Rysunek 5 Ekran startowy systemu ERP RP

źródło: Podręcznik użytkownika systemu EZD RP, podstawy użytkowania, ekran startowy, <https://podrecznik.ezdrp.gov.pl/ekran-startowy/>, 22.08.2023

## Platforma Zendesk z licencją dla sektora edukacji

Platforma Zendesk oferuje dedykowane licencje dostosowane specjalnie do sektora edukacyjnego, dostarczając kompleksowe narzędzia do efektywnego zarządzania w instytucjach edukacyjnych. Dzięki temu specjalnemu rozwiązaniu, szkoły, uczelnie i inne instytucje edukacyjne mogą skorzystać z zaawansowanych funkcji Zendesk, dostosowanych do ich unikalnych potrzeb. Zendesk wspiera efektywną komunikację między pracownikami dziekanatu, a studentami poprzez system biletów, czaty online i możliwość śledzenia zgłoszeń. Dodatkowo, platforma umożliwia zautomatyzowane przypisywanie i rozwiązywanie zadań, co skutkuje szybszą reakcją na pytania oraz problemy studentów.

Możliwość dostosowania do specyfiki instytucji czyni Zendesk dedykowanym rozwiązaniem wspierającym efektywną komunikację w sektorze edukacyjnym. Platforma oferuje funkcję dodawania załączników do biletów, co umożliwia przesyłanie plików w ramach komunikacji między pracownikami dziekanatu, a studentami. Interfejs Zendesk został zaprojektowany w sposób umożliwiający łatwe nawigowanie i korzystanie z różnych funkcji platformy. Dzięki temu użytkownicy mogą efektywnie korzystać z dostępnych narzędzi, co przyczynia się do płynności i skuteczności procesu komunikacji.



Rysunek 6 Ekran główny platformy Zendesk

źródło: https://www.zendesk.com/register/

## Platforma Slack dostosowana do środowiska edukacyjnego

Platforma Slack, dostosowana do kontekstu edukacyjnego, pełni znaczącą rolę jako skuteczne narzędzie ułatwiające komunikację między pracownikami uczelni a studentami. Wykorzystanie kanałów tematycznych pozwala na precyzyjne uporządkowanie dyskusji związanych z różnymi przedmiotami, projektami badawczymi czy wydarzeniami kampusowymi. Pracownicy dziekanatu korzystają z platformy do szybkiego przekazywania informacji dotyczących aktualnych wydarzeń na kampusie, udostępniania materiałów akademickich oraz przekazywania ważnych ogłoszeń.

Dzięki funkcji czatu w czasie rzeczywistym, studenci mogą aktywnie uczestniczyć w procesie komunikacji, zadając pytania i otrzymując natychmiastowe odpowiedzi od pracowników uczelni. Ta interaktywna forma komunikacji istotnie przyczynia się do poprawy jakości dialogu, umożliwiając szybką i skuteczną wymianę informacji. Dodatkowo, możliwość współdzielenia plików w ramach platformy Slack ułatwia przesyłanie dokumentów, co sprzyja efektywnej wymianie informacji. Te rozbudowane funkcje sprawiają, że Slack, dostosowany do środowiska edukacyjnego, staje się platformą sprzyjającą efektywnej i dynamicznej komunikacji między pracownikami uczelni a studentami. Interfejs Slacka został zaprojektowany w taki sposób, aby umożliwić użytkownikom łatwe poruszanie się po platformie i korzystanie z różnych funkcji bez zbędnych trudności. To z kolei przyczynia się do płynnego i intuicyjnego doświadczenia użytkownika, co jest istotne dla efektywnego wykorzystania potencjału komunikacyjnego platformy w kontekście edukacyjnym.



Rysunek 7 Interfejs platformy Slack

źródło: https://slack.com/solutions/distance-learning

## Podsumowanie

Podsumowując analizę porównawczą różnych rozwiązań, można zauważyć, że każde z badanych modułów, systemów czy platform posiada swoje unikalne zalety oraz oferuje różnorodne funkcje, które mogą być dostosowane do specyfiki działania dziekanatu.

W przypadku modułu podań systemu USOS UŚ, pomimo braku pewnych funkcjonalności związanych z komunikacją i współpracą, wartością dodaną jest spójny i intuicyjny interfejs oraz klarowna nawigacja, co ułatwia obsługę systemu.

System elektronicznego zarządzania dokumentacją EZD RP w pełni spełnia przyjęte kryteria funkcjonalne, umożliwiając efektywną komunikację i wspierając przejrzystość procesów dzięki możliwości dołączania załączników i komentowania.

Platforma Zendesk, dzięki dedykowanej licencji edukacyjnej, dostosowuje się do specyfiki instytucji, efektywnie wspierając komunikację i wymianę informacji. System biletowy, możliwość dołączania załączników oraz intuicyjny interfejs sprawiają, że jest kompleksowym narzędziem w obszarze obsługi klienta.

Platforma Slack, dostosowana do środowiska edukacyjnego, wyróżnia się funkcją czatu w czasie rzeczywistym, umożliwiając aktywną komunikację. Możliwość współdzielenia plików oraz przyjazny interfejs stanowią dodatkowe atuty wspierające efektywną komunikację wśród użytkowników.

# Koncepcja własnego rozwiązania

W rozdziale tym przedstawiona zostanie koncepcja innowacyjnego systemu wspierającego wymianę wiadomości między studentem a dziekanatem, opartego na zaawansowanym systemie biletowym. Zaproponowane rozwiązanie skoncentruje się wyłącznie na efektywnej komunikacji, eliminując funkcje administracyjne. System oparty na biletach umożliwi studentom zgłaszanie różnorodnych spraw, takich jak pytania dotyczące przedmiotów, prośby o informacje czy sytuacje wymagające wsparcia ze strony dziekanatu. Każde zgłoszenie zostanie przypisane do unikatowego numeru biletu, co ułatwi śledzenie postępu i zapewni transparentność procesu. Prostota funkcji systemu ma na celu skoncentrowanie się na usprawnieniu komunikacji, bez zbędnych elementów administracyjnych. Interfejs użytkownika zostanie zaprojektowany w sposób przyjazny i intuicyjny, aby ułatwić studentom korzystanie z systemu, a tym samym zminimalizować potencjalne bariery w komunikacji z dziekanatem. Wprowadzenie tego systemu ma na celu efektywne zarządzanie wymianą informacji, poprawę komunikacji oraz zwiększenie satysfakcji studentów z obsługi studenckiej.

## Koncepcja rozwiązania użytkowego

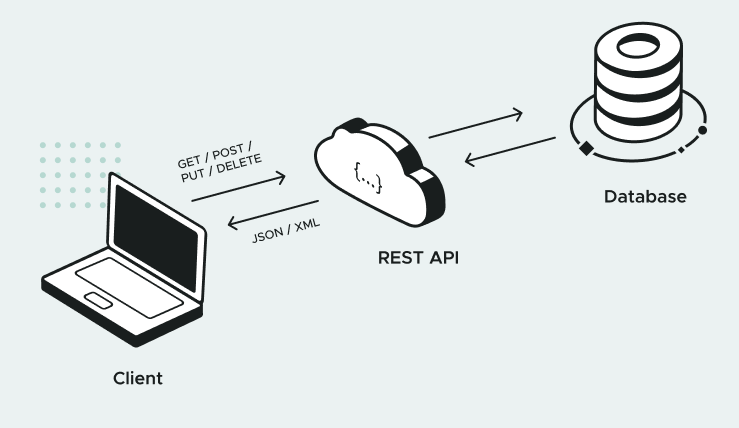
W ramach proponowanej koncepcji praktycznego systemu wymiany informacji między studentami a dziekanatem, opartego na systemie biletowym, priorytetem będzie prostota obsługi i szybkość reakcji. Interfejs tego narzędzia zostanie starannie zaprojektowany, aby każdy student mógł łatwo zrozumieć, jak korzystać z systemu, nie wymagając specjalistycznej wiedzy technicznej. W celu zgłoszenia sprawy, studenci będą mieli dostęp do intuicyjnego formularza, który umożliwi im precyzyjne opisanie problemu. Generowanie unikalnych numerów biletów pozwoli na szybkie i precyzyjne identyfikowanie zgłoszeń, a także umożliwi studentom śledzenie postępu w ich rozpatrywaniu.

System ten będzie również uwzględniał możliwość natychmiastowych powiadomień, informujących studentów o statusie ich zgłoszeń oraz ewentualnych dodatkowych krokach do podjęcia. Dziekanat będzie mógł efektywnie zarządzać zgłoszeniami, przypisując odpowiednich pracowników do ich rozpatrzenia. Dodatkowo, wbudowane funkcje raportowania i analizy pozwolą dziekanatowi na monitorowanie trendów i identyfikację obszarów wymagających dodatkowego wsparcia lub usprawnień.

Wprowadzenie tego systemu ma na celu nie tylko zwiększenie efektywności komunikacji między studentami a dziekanatem, ale także poprawę jakości obsługi studenckiej, redukcję czasu oczekiwania na odpowiedzi oraz zwiększenie satysfakcji studentów z procesów administracyjnych na uczelni. To narzędzie nie tylko ułatwi wymianę informacji, ale również przyczyni się do bardziej efektywnego i zintegrowanego zarządzania sprawami studenckimi.

## Koncepcja rozwiązania technologicznego

Koncepcja rozwiązania technologicznego systemu wymiany wiadomości pomiędzy studentem a pracownikami dziekanatu oparta jest na architekturze REST, stanowiącej kluczowy element projektu. Aplikacja internetowa zostanie opracowana przy użyciu języków Java i TypeScript, wykorzystując szablony Spring i Angular oraz silnika bazy danych MSSQL. Architektura REST, przedstawiona na rysunku 8, umożliwi spójną i efektywną komunikację pomiędzy klientem a serwerem, co jest kluczowe dla dynamicznego obiegu informacji w systemie. W kontekście operacji CRUD dla zgłoszeń, REST pozwoli na zaimplementowanie prostych i jednolitych interfejsów, umożliwiając studentom zgłaszanie, przeglądanie, aktualizację i usuwanie swoich zgłoszeń. To z kolei ułatwi śledzenie postępu w rozpatrywaniu spraw, zgodnie z zasadami architektury REST. Do transportu danych pomiędzy klientem a REST API użyty zostanie format JSON. Dodatkowo, w ramach tej koncepcji, wykorzystany zostanie silnik bazy danych MSSQL w celu skutecznego przechowywania i zarządzania danymi systemowymi. MSSQL oferuje nie tylko wysoką wydajność, ale także solidne mechanizmy zabezpieczające integralność danych. Dzięki temu, informacje dotyczące zgłoszeń, komunikatów i historii interakcji między studentami a pracownikami dziekanatu będą przechowywane w sposób niezawodny i dostępne dla systemu w czasie rzeczywistym. Implementacja architektury REST w połączeniu z silnikiem bazy danych MSSQL ma na celu zapewnienie skalowalności, elastyczności oraz wysokiej wydajności systemu. To kompleksowe podejście pozwala skoncentrować się na efektywnej wymianie informacji między użytkownikami, jednocześnie gwarantując solidne zarządzanie danymi oraz zgodność z nowoczesnymi standardami technologicznymi.



Rysunek 8 Schemat architektury REST API

źródło: https://spring.academy/courses/building-a-rest-api-with-spring-boot/lessons/implementing-get

# Projekt ogólny

Rozdział ten prezentuje ogólny projekt systemu wspomagającego komunikację między studentami a pracownikami dziekanatu, skupiający się na efektywnym przepływie informacji i optymalizacji procesów komunikacyjnych.

## Specyfikacja wymagań funkcjonalnych i niefunkcjonalnych

Wymagania funkcjonalne:

* Dodawanie użytkowników, administracja użytkownikami
* Logowanie, Zmiana hasła,
* Zakładanie zgłoszeń
* Wgląd do archiwalnych zgłoszeń, historia zgłoszeń
* Kategoryzowanie zgłoszeń
* Priorytetyzacja zgłoszeń
* Monitorowanie statusu zgłoszenia
* Przypisywanie zgłoszeń do pracowników dziekanatu
* Komentarze w zgłoszeniach
* Powiadomienia e-mail oraz powiadomienia push
* Wyszukiwanie zgłoszeń
* Zamknięcie zgłoszenia
* Archiwizacja danych – dokumenty/załączniki

Wymagania niefunkcjonalne:

1. System powinien zostać zaprojektowany w architekturze klient – serwer.
2. Dostępność systemu 24 godziny na dobę przez 7 dni w tygodniu, z minimalnymi przerwami na konserwację.
3. System powinien jednocześnie obsługiwać co najmniej 500 studentów bez utraty wydajności.
4. Dane przesyłane powinny być szyfrowane w celu zabezpieczenia prywatności.
5. System powinien zawierać autoryzację i uwierzytelnianie, aby zapewnić dostęp jedynie upoważnionym osobą.
6. Interfejs użytkownika powinien być intuicyjny i łatwy w użyciu.
7. Obiekty w systemie powinny charakteryzować się spójną kolorystyką.

Podrozdział w Projekt ogólny. Specyfikacja wymagań jest fragmentem SRS — Software Requirements Specification. Szczegółowe informacje na temat dokumentu SRS dostępne są tutaj (PDF). Specyfikacja wymagań funkcjonalnych zawiera opis funkcji udostępnianych przez system. Np. Otwarcie istniejącego raportu, Utworzenie nowego raportu, Edycja ustawień, itp. Funkcje systemu odpowiadają zazwyczaj przypadkom użycia występującym na diagramie Use Case (PDF), taki diagram jest mile widziany jako element specyfikacji wymagań funkcjonalnych. Poszczególne wymagania mogą być prezentowane w postaci listy numerowanej lub bardziej szczegółowo w postaci tabeli. Wymagania niefunkcjonalne dotyczą aspektów systemu nie przekładających się bezpośrednio na akcje wykonywane przez system, dotyczą takich aspektów jak architektura, bezpieczeństwo, wydajność, ergonomia interfejsu użytkownika, kolorystyka. Wystarczy prezentacja wymagań niefunkcjonalnych w postaci listy numerowanej.

W przypadku osób przyzwyczajonych do pracy z wykorzystaniem metodyk zwinnych, można przedstawić specyfikację funkcji systemu w postaci historii użytkownika.

## Architektura systemu

Podrozdział w Projekt ogólny. Określamy jakiego rodzaju będzie system (system klasy desktop, mobilny, internetowy). W zależności od danego rodzaju systemu określamy czy będzie podzielony na podsystemy, elementy, warstwy, moduły, określamy gdzie będą one ulokowane i jak będą ze sobą współpracować. Przykład: “System będzie trójwarstwową aplikacją internetową. Warstwa kliencka działać będzie w środowisku przeglądarki, będzie ona się komunikować z warstwą usług serwerowych za pośrednictwem API REST. API będzie korzystało z bazy danych umieszczonej na osobnym, dedykowanym serwerze”. Dalej następują rysunki ideowe ilustrujące architekturę aplikacji, ogólny opis przeznaczenia oraz roli poszczególnych elementów aplikacji, metod komunikacji pomiędzy nimi.

## Metody i narzędzia realizacji

Podrozdział w Projekt ogólny. Opierając się na znanych już wymaganiach oraz znanej architekturze systemu dokonujemy krótkiego przeglądu możliwych metod oraz narzędzi jego realizacji, wybieramy jedną wersję oraz uzasadniamy wybór (o ile nie dokonano tego w rozdziale dotyczącym koncepcji rozwiązania technologicznego). Przedstawiamy krótką charakterystykę narzędzi z uszczegółowieniem wersji którą planujemy wykorzystać. Uwaga: nie wstawiamy tutaj rozbudowanych opisów narzędzi, nie opisujemy języków programowania i wszystkich ich mechanizmów, nie robimy tutorialu z serwerów baz danych, innymi słowy nie robimy “lania wody”, ma być krótko, konkretnie i kompetentnie, a zatem jeżeli wybieramy PHP, to uzasadniamy dlaczego, jeżeli wybieramy konkretny framework dla PHP to piszemy dlaczego ten właśnie, i której wersji planujemy użyć. Przechodzimy do szczegółów jedynie wtedy, gdy są one istotne dla projektu w fazie projektu ogólnego, np. jeżeli istotnym elementem wpływającym na wybór biblioteki Qt jest mechanizm slotów i sygnałów, to można napisać parę zdań na ten temat, co ułatwi wskazanie uzasadnienia wybory tejże biblioteki.

## Koncepcja przechowywania danych

Podrozdział w Projekt ogólny. Zazwyczaj systemy informatyczne potrzebują danych przechowywanych trwale. Wykorzystuje się do tego najczęściej bazy danych, zwykle trzeba je zaprojektować. W tym podrozdziale należy przedstawić jak będziemy zapisywać dane trwałe, jeżeli stosujemy bazę danych to należy określić jej model oraz przedstawić szczegóły modelu. W przypadku baz relacyjnych podajmy model konceptualny, logiczny i fizyczny, w przypadku gdy merytorycznie one się istotnie nie różnią, podajemy model najbardziej pełny. Ilustracją wybranego modelu są najczęściej diagramy, np. diagramy ERD opisane tutaj (PDF).

Uwaga, zamieszczenie samych diagramów nie wystarcza, należy przedstawić również opis przeznaczenia poszczególnych tabel, wraz z informacjami o kluczach głównych i obcych. Dla baz nierelacyjnych podajemy adekwatne dla tych baz formy ilustracji metody organizacji danych. W przypadku, gdy bazy danych nie są wykorzystywane w projektowanym systemie, przedstawiamy przyjętą koncepcję przechowywania informacji (pliki serializacyjne, pliki tekstowe, pliki CSV, pliki XML, pliki binarne), oraz szczegóły ich reprezentacji, przyjęte formaty.

## Projekt interfejsu użytkownika

Podrozdział w Projekt ogólny. Prezentacja koncepcji organizacji interfejsu użytkownika, przyjmująca postać rysunków wraz ze stosownym komentarzem. rysunki mogą zawierać szkice interfejsu (również odręczne, skany), makiety z odpowiednich programów do modelowania interfejsu użytkownika. Uwaga — nie umieszczamy tutaj finalnych elementów gotowego systemu, przecież go jeszcze nie ma! Jeżeli aplikacja mam być responsywna i w różny sposób “wyglądać” na różnych urządzeniach, należy przedstawić szkice interfejsu dla każdego z przewidywanych trybów pracy.

# Dokumentacja techniczna

Struktura tego rozdziału mocno zależy od architektury danego rozwiązania, zastosowanych metod i narzędzi realizacji. Należy rozpocząć od schematu rzeczywistej struktury systemu, prezentującego poszczególne składowe: pakiety, moduły, biblioteki, klasy. Każda składowa powinna zostać omówiona, opis powinien zawierać nazwę składowej, jej rolę w projekcie, przeznaczenie, strukturę wewnętrzną.

Ten rozdział to dobre miejsce na umieszczenie różnorodnych diagramów projektowych, takich jak diagram hierarchii klas, diagramy sekwencji, kolaboracji, itp., itd.. Tutaj można przedstawić opisy metod i/lub stosowane algorytmy.

Przy opisie składowych systemu warto przedstawić fragmenty wybranych kodów źródłowych, prezentujące szczególnie istotne, ciekawe czy nietypowe rozwiązania oraz algorytmy. Nie należy przedstawiać fragmentów kodu operacji typowych lub trywialnych. Proszę nie zamieszczać długich, ciągnących się przez wiele stron listingów kodu. Opis kodu zapisujemy w sekwencji: kilka linijek a potem fragment opisujący istotne operacje realizowane w kodzie, kilka linijek a potem fragment opisujący istotne operacje realizowane w kodzie… itd.

Fragmenty kodu proszę zamieszczać jako odpowiednio sformatowane teksty, zwykle pisane nieproporcjonalnym krojem pisma. Proszę zadbać o to, by zamieszczane kody były poprawne z punktu widzenia inżynierii programowania, poprawne merytorycznie oraz właściwie sformatowane. Proszę nie umieszczać w tekście fragmentów kodu w postaci rysunków zawierających zrzuty ekranów.

Proszę pamiętać, że praca inżynierska ma być wymiernym dowodem wysokich kompetencji jej autora w zakresie informatyki, niechlujny kod takich kompetencji nie potwierdza.

# Testy i weryfikacja systemu

# Przykładowy scenariusz wykorzystania systemu

# Zakończenie

# Bibliografia

1. Autor anonimowy, Tworzenie bibliografii pracy dyplomowej [on-line], Polski Portal Edukacyjny, http://www.e\_nauka.pl, [dostępne: 4 kwietnia 2016].
2. Kowalski J., Jak cytować materiały źródłowe, Oficyna Wydawnicza Example, Katowice, 2014.
3. Autor anonimowy, Przeniesienie (studia stacjonarne i niestacjonarne), <https://us.edu.pl/wydzial/wnst/studia/student/regulaminy/>, [dostępne: 11 października 2021].
4. O systemie, <https://www.gov.pl/web/ezd-rp/o-ezd-rp>

# Spis rysunków

[Rysunek 1Aktualny obieg dokumentów w dziekanacie studenckim 6](#_Toc157682511)

[Rysunek 2Wniosek o przeniesienie z formy studiów stacjonarnych na niestacjonarne 7](#_Toc157682512)

[Rysunek 3 Moduł podań systemu USOS UŚ 10](#_Toc157682513)

[Rysunek 4 Składanie nowego podania w systemie USOS UŚ z listy określonych podań 10](#_Toc157682514)

[Rysunek 5 Ekran startowy systemu ERP RP 12](#_Toc157682515)

[Rysunek 6 Ekran główny platformy Zendesk 13](#_Toc157682516)

[Rysunek 7 Interfejs platformy Slack 14](#_Toc157682517)

[Rysunek 8 Schemat architektury REST API 17](#_Toc157682518)

# Spis tabel

[Tabela 1 Przykładowa tabela 15](#_Toc155265347)