

# Markdown to PDF

## We've converted 1 306 905 Markdown files to PDF and counting!

To convert your Markdown to PDF simply start by typing in the editor or pasting from your clipboard.

If your Markdown is in a file clear this content and drop your file into this editor.

**tip:** click on the pencil icon on the left to clear the editor)

## Classic Markdown to PDF1

If you have an account at classDokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat: Aplikacja Bankowa Autor: Jakub Bączalski Grupa: 20A Kierunek: informatyka Rok akademicki: 2023 Poziom i semestr: I/4 Tryb studiów: niestacjonarne

1	Spis treści	
2	Odnosniki do innych źródeł	4
3	Słownik pojęć	5
4	Wprowadzenie	6
4.1	Cel dokumentacji	6
4.2	Przeznaczenie dokumentacji	6
4.3	Opis organizacji lub analiza rynku	6
4.4	Analiza SWOT organizacji	6
5	Specyfikacja wymagań	7
5.1	Charakterystyka ogólna	7
5.2	Wymagania funkcjonalne	7
5.3	Wymagania niefunkcjonalne	8
6	Zarządzanie projektem	9
6.1	Zasoby ludzkie	9
6.2	Harmonogram prac	9
6.3	Etapy/kamienie milowe projektu	9
7	Zarządzanie ryzykiem	10
7.1	Lista czynników ryzyka	10
7.2	Ocena ryzyka	10
7.3	Plan reakcji na ryzyko	10
8	Zarządzanie jakością	11
8.1	Scenariusze i przypadki testowe	11
9	Projekt techniczny	12
9.1	Opis architektury systemu	12
9.2	Technologie implementacji systemu	12
9.3	Diagramy UML	12
9.4	Charakterystyka zastosowanych wzorców projektowych	12
9.5	Projekt bazy danych	12
9.6	Projekt interfejsu użytkownika	12
9.7	Procedura wdrożenia	13
10	Dokumentacja dla użytkownika	14
11	Podsumowanie	15
11.1	Szczegółowe nakłady projektowe członków zespołu	15
12	Inne informacje	16

### 2 Odnosniki do innych źródeł

- projekt GUI razem z plecami takimi jak logowanie, itd.: <https://github.com/Rotages/Laravel-project>
- API NBP do pobierania kursu walut, swojej budowy: <https://github.com/Rotages/ZadanieTestowe>

### 3 Słownik pojęć

Pojęcie	Wyjaśnienie
Tabela	W kontekście bazy danych, tabela jest strukturą, która składa się z kolumn i wierszy. Każda kolumna
Klucz główny	Klucz główny to unikalny identyfikator dla każdego rekordu w tabeli. Służy do jednoznacznego identyfikowania
Klucz obcy	Klucz obcy to pole w tabeli, które odwołuje się do klucza głównego innej tabeli. Jest to mechanizm używany
ENUM	ENUM to typ danych w bazie danych, który ogranicza wartości, jakie mogą być przechowywane w danej kolumnie
DECIMAL	DECIMAL to typ danych numerycznych w bazie danych, który służy do przechowywania liczb dziesiętnych o
DATETIME	DATETIME to typ danych w bazie danych, który służy do przechowywania daty i godziny. Reprezentuje on wartość

## 4 Wprowadzenie

### 4.1 Cel dokumentacji

Dokumentacja dla aplikacji bankowej jest kluczowa dla zapewnienia skutecznego i bezpiecznego jej działania. Głównym celem dokumentacji jest dostarczenie kompletnych informacji o funkcjonalności aplikacji, włącznie z opisem interfejsu użytkownika. Dokumentacja ma również na celu pomóc programistom i testerom w zrozumieniu kodu źródłowego, dzięki czemu mogą szybciej i skutecznie. Wreszcie, dokumentacja jest ważna dla użytkowników, którzy potrzebują jasnych i zrozumiałych instrukcji dotyczących korzystania z aplikacji.

### 4.2 Przeznaczenie dokumentacji

Dokumentacja aplikacji bankowej jest przeznaczona dla różnych grup odbiorców, w tym:

1. Programistów - którzy będą pracować nad dalszym rozwojem aplikacji lub wprowadzać zmiany i aktualizacje. Dokumentacja ta zawiera s
2. Testerów - którzy będą przeprowadzać testy funkcjonalne i нефункционалне w celu sprawdzenia jakości i wydajności aplikacji. Dokum
3. Analityków biznesowych - którzy będą analizować dane dotyczące użytkowania aplikacji, trendów i zachowań użytkowników. Dokumentacj
4. Użytkowników - którzy będą korzystać z aplikacji bankowej. Dokumentacja ta zawiera jasne instrukcje dotyczące korzystania z aplika

### 4.3 Opis organizacji lub analiza rynku

Organizacja, dla której aplikacja bankowa zostanie zrealizowana, to bank o ugruntowanej pozycji na rynku finansowym. Jest to inst Działanie banku opiera się na złożonych procesach biznesowych, które wymagają precyzyjnego planowania i realizacji. Istotną kwestią Wdrożenie aplikacji bankowej ma na celu usprawnienie procesów biznesowych banku oraz zwiększenie jakości świadczonych usług. Aplikac Warto również podkreślić, że bank, dla którego realizowana jest aplikacja, dąży do stałego rozwoju i innowacji w swojej działalności

### 4.4 Analiza SWOT organizacji

**Silne strony:**

1. Ugruntowana pozycja na rynku finansowym
2. Sieć oddziałów w wielu miastach
3. Wieloletnie doświadczenie w branży finansowej
4. Wysoko wykwalifikowany personel
5. Różnorodność kanałów dystrybucji usług finansowych

**Szanse:**

1. Możliwość wdrożenia nowych produktów finansowych
2. Możliwość pozyskania nowych klientów
3. Wzrost zainteresowania klientów nowoczesnymi usługami bankowymi

**Zagrożenia:**

1. Konkurencja na rynku finansowym
2. Zmiany regulacyjne i prawne
3. Zmiany rynkowe, np. wzrost stóp procentowych lub spadek popytu na usługi finansowe

**Słabe strony:**

1. Wysoki koszt obsługi klienta
2. Złożone procesy biznesowe
3. Brak elastyczności w realizacji zadań
4. Niska innowacyjność

## 5 Specyfikacja wymagań

### 5.1 Charakterystyka ogólna

#### 5.1.1 Definicja produktu

System bankowy to kompleksowe narzędzie finansowe, obejmujące m.in. kalkulator walutowy, obsługę rachunków bankowych, kredytów or

#### 5.1.2 Podstawowe założenia

Celem systemu bankowego jest zapewnienie szybkiej, bezpiecznej i wygodnej obsługi finansowej dla klientów banku. System umożliwi

#### 5.1.3 Cel biznesowy

Głównym celem biznesowym organizacji jest zwiększenie konkurencyjności na rynku finansowym poprzez udostępnienie klientom nowocze

#### 5.1.4 Użytkownicy

System będzie przeznaczony dla szerokiego grona użytkowników, w tym dla klientów indywidualnych, firm oraz instytucji. Głównym

#### 5.1.5 Korzyści z systemu

Dla firm i instytucji:

1. Automatyzacja procesów finansowych, co pozwoli na zaoszczędzenie czasu oraz zmniejszenie kosztów związanych z ręcznym przetw
3. Przeglądanie i generowanie raportów finansowych w czasie rzeczywistym, co pozwoli na szybsze podejmowanie decyzji biznesowyc
4. Bezpieczeństwo transakcji finansowych, co zapewni klientom większe zaufanie do firmy i przyczyni się do wzrostu ich lojalność
5. Zarządzanie kontami i dostęp do informacji o transakcjach, co umożliwi firmom lepszą kontrolę nad swoimi finansami i szybsze
6. Integracja z innymi systemami firmowymi, co pozwoli na automatyzację całego procesu biznesowego oraz uniknięcie błędów ludzk
7. Procesu obsługi klienta, co przyczyni się do zwiększenia zadowolenia klientów i poprawy wizerunku firmy.
8. Śledzenie płatności i wystawiania faktur, co ułatwi prowadzenie działalności handlowej.
9. Optymalizacja kosztów związanych z obsługą transakcji finansowych, co przyczyni się do zwiększenia zysków firmy.

Dla klientów:

1. Dostęp do aplikacji bankowej 24/7, co pozwala na zarządzanie kontem z dowolnego miejsca i o każdej porze dnia i nocy.
2. Przeglądania salda i historii transakcji w czasie rzeczywistym.
3. Obsługa transakcji bez konieczności wizyty w oddziale banku.

Dla pracowników:

1. Przeprowadzanie operacji bankowych, takich jak weryfikacja klientów i przetwarzanie transakcji.
2. Dostęp do informacji o klientach i ich kontaktach.
3. Definicja szablonów kalkulacyjnych (wsparcie w pracy, dzięki automatyzacji niektórych zadań, co umożliwia skupienie się na z

Dla administratorów systemu:

1. Administracja systemu dzięki zastosowaniu jednolitej platformy.
2. Rozwiązywanie problemów technicznych dzięki rozbudowanym narzędziom monitorowania systemu. Przeglądanie logów aplikacji.
3. Dostęp do metryk systemu, na których możemy przeprowadzić analizę.

#### 5.1.6 Ograniczenia projektowe i wdrożeniowe

#### 1. Przepisy prawne:

Aplikacja bankowa musi spełniać wymogi prawne i regulacje związane z branżą bankową, takie jak GDPR, dyrektywy UE, Kodeks Cywilny. Przede wszystkim, aplikacja bankowa musi działać zgodnie z przepisami prawa bankowego oraz prawa ochrony danych osobowych, takim

#### 2. Narzędzia b.d.:

System ma korzystać z serwera bazy danych MySQL.

#### 3. Protokoły komunikacyjne:

Aplikacja bankowa musi korzystać z odpowiednich protokołów komunikacyjnych, takich jak HTTPS.

#### 5. Aspekty zabezpieczeń:

Bezpieczeństwo aplikacji bankowej jest kluczowe, dlatego musi ona spełniać wysokie wymagania dotyczące ochrony danych, autentykac

#### 6. Zgodność ze standardami:

Aplikacja bankowa musi spełniać normy i standardy branżowe, takie jak ISO 27001.

#### 7. Powiązania z innymi aplikacjami:

Aplikacja bankowa musi być zintegrowana z innymi systemami i aplikacjami, takimi jak systemy płatnicze, aplikacje mobilne, system

#### 9. System operacyjny: Aplikacja bankowa musi działać na odpowiednim systemie operacyjnym: Windows, Linux, MacOS itp.



### 5.2 Wymagania funkcjonalne:

#### 5.2.1 Lista wymagań:

- Kalkulator z kursami walut pobieranych z API NBP
- System logowania użytkowników
- Przeglądanie salda konta
- Wykonywanie przelewów
- Zarządzanie kontami bankowymi
- Obsługa kart płatniczych

#### 5.2.2 Diagramy przypadków użycia:

Diagram przypadków użycia systemu logowania użytkowników:

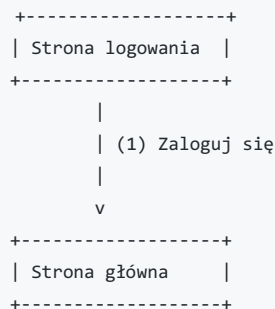


Diagram przypadków użycia przeglądania salda konta:

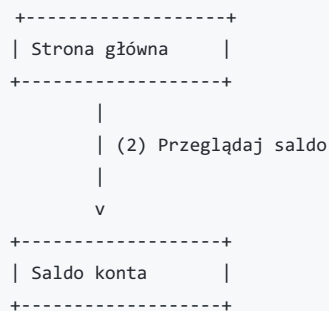


Diagram przypadków użycia wykonywania przelewów:

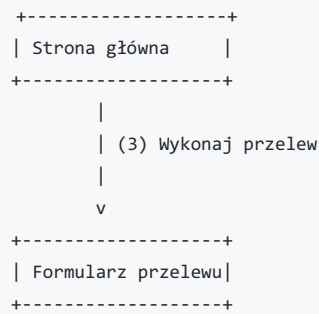
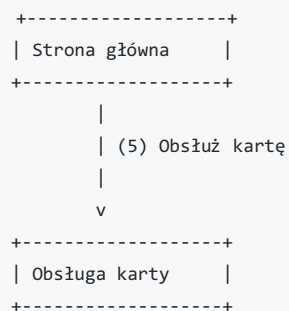


Diagram przypadków użycia zarządzania kontami bankowymi:



Diagram przypadków użycia obsługi kart płatniczych:



5.2.3 Szczegółowy opis wymagań dla 5-7 wybranych najważniejszych przypadków użycia:

1. Przykładowo dla przypadku użycia "Wykonywanie przelewów":

Nazwa: Wykonywanie przelewów

Uzasadnienie biznesowe: Umożliwienie klientom dokonywania przelewów z ich konta bankowego.

Użytkownicy: Klienci banku

Scenariusze:

Nazwa scenariusza: Wykonanie standardowego przelewu

Warunki początkowe: Klient jest zalogowany do systemu i posiada wystarczającą ilość środków na koncie.

Przebieg działań:

Klient wybiera opcję wykonania przelewu.

System prosi o podanie danych odbiorcy (numer konta, nazwa banku, nazwa odbiorcy).

Klient wprowadza dane odbiorcy.

System prosi o podanie kwoty przelewu.

Klient wprowadza kwotę przelewu.

System prosi o potwierdzenie danych przelewu.

Klient potwierdza dane przelewu.

Efekty: Środki zostają przelane na konto odbiorcy.

Wymagania niefunkcjonalne:

Częstotliwość: 5 (często wykorzystywany przez klientów)

Istotność: 5 (kluczowa funkcjonalność systemu)

5.3 Wymagania niefunkcjonalne:

- Wydajność:
  1. Przepustowość systemu powinna wynosić co najmniej 100 transakcji na sekundę.
- Bezpieczeństwo:
  1. System powinien być odporny na ataki typu SQL injection.
  2. Hasła użytkowników powinny być przechowywane w bezpieczny sposób, np. zaszyfrowane w bazie danych. Ponadto, system powinien po
- Zabezpieczenia:
  1. System powinien posiadać mechanizmy zapobiegające utracie danych, np. poprzez regularne tworzenie kopii zapasowych. Ponadto, p
- Inne cechy jakości:
  1. System powinien być łatwy w obsłudze dla użytkowników, zapewniać szybki czas odpowiedzi oraz łatwość konserwacji. Ponadto, pow

## 6 Zarządzanie projektem

### 6.1 Zasoby ludzkie

W ramach projektu aplikacji bankowej, zasoby ludzkie obejmują różne role. Zespół projektowy składa się z następujących osób:

Kierownik projektu: Osoba odpowiedzialna za zarządzanie całym projektem, koordynację zespołu oraz nadzór nad postępem prac.

Analitik biznesowy: Osoba odpowiedzialna za analizę wymagań klienta oraz opracowanie specyfikacji funkcjonalnej aplikacji.

Projektant interfejsu użytkownika: Osoba odpowiedzialna za zaprojektowanie intuicyjnego i atrakcyjnego interfejsu użytkownika aplika

Programiści: Zespół programistów odpowiedzialnych za implementację aplikacji bankowej.

Testerzy: Zespół testerów, którzy przeprowadzą testy jednostkowe, integracyjne oraz testy akceptacyjne, aby upewnić się o jakości ap

Administratorzy systemu: Osoby odpowiedzialne za konfigurację, utrzymanie i monitorowanie infrastruktury, na której będzie działać a

### 6.2 Harmonogram prac

Harmonogram prac dla projektu aplikacji bankowej obejmuje następujące etapy:

Etap 1: Analiza wymagań (trwałość: 2 tygodnie)

Etap 2: Projektowanie interfejsu użytkownika (trwałość: 1 tydzień)

Etap 3: Implementacja aplikacji (trwałość: 4 miesiące)

Etap 4: Testowanie i rozwiązywanie usterek (trwałość: 2 tygodnie)

Etap 5: Wdrożenie aplikacji (trwałość: 1 tydzień)

Etap 6: Szkolenie personelu (trwałość: 2 dni)

Etap 7: Serwis i konserwacja (trwałość: bezterminowo, jako stała część utrzymania aplikacji)

### 6.3 Etapy/kamienie milowe projektu

Główne etapy projektu aplikacji bankowej:

Etap 1: Analiza wymagań: Przeprowadzenie badań rynku, analiza potrzeb klienta i określenie funkcjonalności aplikacji.

Etap 2: Projektowanie interfejsu użytkownika: Opracowanie prototypów interfejsu, dostosowanie do potrzeb użytkowników i zatwie

Etap 3: Implementacja aplikacji: Tworzenie kodu źródłowego, implementacja funkcjonalności, integracja z systemami bankowymi i

Etap 4: Testowanie i rozwiązywanie usterek: Przeprowadzenie testów integracyjnych, testów akceptacyjnych i identyfikowanie ora

Etap 5: Wdrożenie aplikacji: Instalacja aplikacji na serwerze produkcyjnym, konfiguracja środowiska produkcyjnego oraz migracj

Etap 6: Szkolenie personelu: Przeprowadzenie szkoleń dla pracowników banku, aby zapewnić im umiejętność korzystania z aplikacj

Etap 7: Serwis i konserwacja: Zapewnienie stałej opieki technicznej, utrzymanie aplikacji, monitorowanie jej działania i wpro

## 7 Zarządzanie ryzykiem

### 7.1 Lista czynników ryzyka

- 1.Brak wystarczającej analizy wymagań klienta
- 2.Opóźnienia w dostawie zewnętrznych usług lub komponentów
- 3.Konflikty harmonogramowe z innymi projektami w organizacji
- 4.Nieodpowiednie zarządzanie zasobami ludzkimi
- 5.Brak umiejętności technicznych lub doświadczenia w zespole projektowym
- 6.Awaria infrastruktury technologicznej
- 7.Problemy z bezpieczeństwem danych i cyberbezpieczeństwem
- 8.Zmiany w regulacjach lub przepisach dotyczących sektora bankowego
- 9.Niski poziom zaangażowania klientów lub niewłaściwa akceptacja aplikacji
- 10.Konflikty interesów wewnątrz organizacji klienta

### 7.2 Ocena ryzyka

Czynnik ryzyka    Prawdopodobieństwo    Wpływ

1	3	4
2	2	3
3	3	2
4	4	3
5	2	4
6	3	5
7	4	5
8	2	3
9	3	2
10	2	3

### 7.3 Plan reakcji na ryzyko

Czynnik ryzyka

Plan reakcji

1	Przeprowadzenie szczegółowej analizy wymagań klienta, uwzględniając regularne spotkania i konsultacje.
3	Utworzenie harmonogramu projektów w organizacji, aby uniknąć konfliktów harmonogramowych.
6	Regularne monitorowanie stanu infrastruktury, utrzymanie kopii zapasowych i planowanie awaryjne w razie
7	Wdrożenie odpowiednich środków bezpieczeństwa danych, takich jak szyfrowanie, uwierzytelnianie wieloskład
10	Ustanowienie jasnych polityk i procedur w organizacji klienta, aby zapobiegać konfliktom interesów.

## 8 Zarządzanie jakością

### 8.1 Scenariusze i przypadki testowe

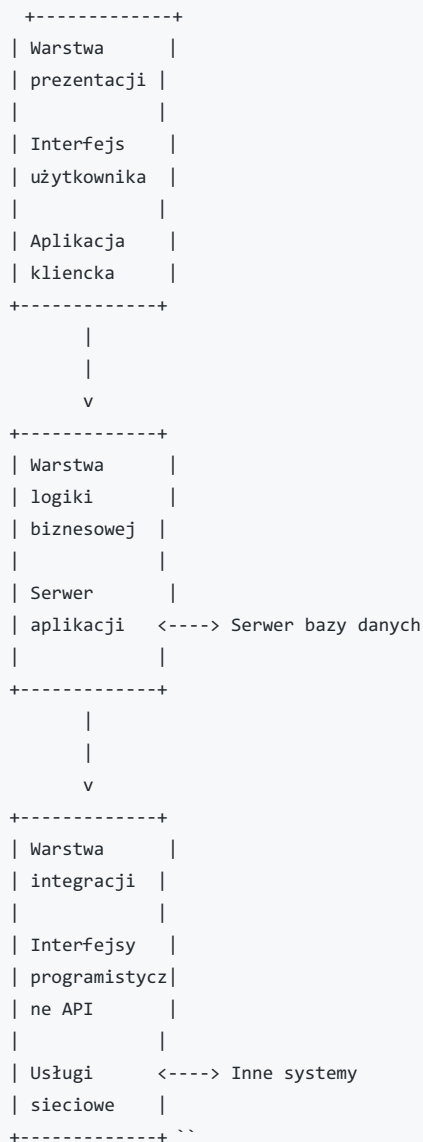
Numer	Nazwa scenariusza	Kategoria	Opis	Tester	Termin	Narzędzia wspomagające
1	Rejestracja użytkownika	Testy funkcjonalne	Testowanie procesu rejestracji nowego użytkownika w aplikacji	Tester1	01.07.2023	Brak
2	Logowanie do systemu	Testy funkcjonalne	Testowanie procesu logowania użytkownika do aplikacji	Tester2	05.07.2023	Brak
3	Wykonanie przelewu	Testy funkcjonalne	Testowanie procesu wykonania przelewu środków z jednego konta na inne	Tester3	10.07.2023	Brak
4	Sprawdzenie historii transakcji	Testy funkcjonalne	Testowanie funkcji wyświetlania historii transakcji dla danego konta	Tester4	15.07.2023	Brak

## 9 Projekt techniczny

### 9.1 Opis architektury systemu Architektura systemu aplikacji bankowej składa się z kilku elementów, które łącznie tworzą spójną całość.

- Warstwa prezentacji - jest to część systemu, która odpowiada za interakcję użytkownika z aplikacją. Składa się ona z interfejsu użytkownika i aplikacji klienckiej.
- Warstwa logiki biznesowej - jest to część systemu, która odpowiada za logikę biznesową aplikacji. Składa się ona z serwera aplikacji i serwera bazy danych.
- Warstwa integracji - jest to część systemu, która odpowiada za integrację aplikacji bankowej z innymi systemami. Składa się ona z interfejsu programistycznego i usług sieciowych.

Poniżej przedstawiony jest schematyczny rysunek architektury systemu aplikacji bankowej:



## 9.2 Technologie implementacji systemu

PHP - Wybrano język programowania PHP jako główny język implementacji, ponieważ jest to popularny język skryptowy wykorzystywany do

Blade - Do implementacji warstwy prezentacji wykorzystano framework Blade, który jest częścią frameworka Laravel. Użyto go ze względu

MySQL - Jako system zarządzania bazą danych wykorzystano MySQL, ponieważ jest to popularny, bezpieczny i wydajny system. MySQL jest

API - Do pobierania kursów walut dla kalkulatora wykorzystano API. Wybrano tę technologię, ponieważ umożliwia ona łatwe i szybkie po

## 9.3 Diagramy UML

### 9.3.1 Diagram(-y) klas:



Użytkownik	Konto	Przelew
-id: int	-numer: string	-id: int
-imie: string	-saldo: float	-kwota: float
-nazwisko: string	-użytkownik_id: int	-odbiorca: string
-email: string   1-----1..*	1-----1..*	-tytuł: string
-hasło: string		-data: datetime

### 9.3.2 Diagram(-y) czynności:

#### 1. Logowanie użytkownika:

```

sequenceDiagram
    participant U as Użytkownik
    U->>S: (1) Wykonaj logowanie
    activate S
    S->>U: 
    deactivate S
    participant S as Strona logowania
  
```

#### 2. Wykonanie przelewu

```

sequenceDiagram
    participant S as Strona główna
    S->>F: (2) Wykonaj przelew
    activate F
    F->>S: 
    deactivate F
    participant F as Formularz przelewu
  
```

#### 3. Przeglądanie historii transakcji

```

sequenceDiagram
    participant S as Strona główna
    S->>H: (3) Przeglądaj historię
    activate H
    H->>S: 
    deactivate H
    participant H as Historia transakcji
  
```

### 9.3.3 Diagramy sekwencji:

#### 1. Zarządzanie kontami bankowymi

```

+-----+
| Strona główna |
+-----+
| -użytkownik_id: int |
+-----+
|
| (7) Zarządzaj kontami
|
v
+-----+
| Lista kont |
+-----+

```

## 2. Obsługa kart płatniczych

```

+-----+
| Strona główna |
+-----+
| -użytkownik_id: int |
+-----+
|
| (8) Obsłuż kartę
|
v
+-----+
| Obsługa karty |
+-----+

```

## 3. Zmiana ustawień użytkownika

```

+-----+
| Strona ustawień |
+-----+
| -użytkownik_id: int |
+-----+
|
| (10) Zmień ustawienia
|
v
+-----+
| Formularz ustawień |
+-----+

```

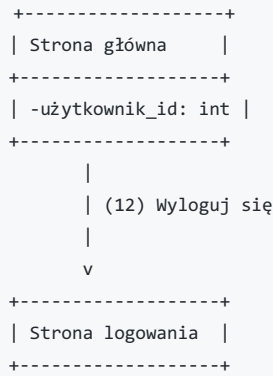
## 4. Pobieranie kursów walut z API NBP

```

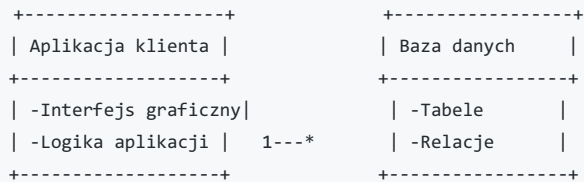
+-----+
| Strona kalkulatora |
+-----+
|
|
+-----+
|
| (11) Pobierz kursy walut
|
v
+-----+
| API NBP |
+-----+

```

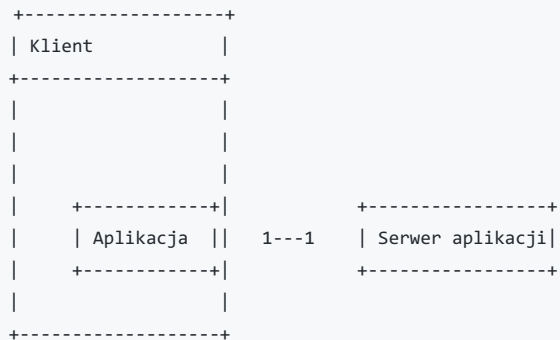
## 5. Wylogowanie użytkownika



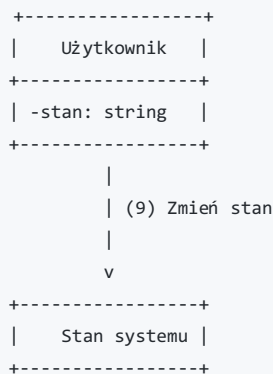
#### 9.3.4 Inne diagramy Komponentów:



Rozmieszczenia:



Maszyny stanowej:



#### 9.4 Charakterystyka zastosowanych wzorców projektowych

W projekcie zastosowano kilka wzorców projektowych, które pomogły w zaprojektowaniu i implementacji systemu. Poniżej przedstawione są dwa zastosowane wzorce projektowe wraz z diagramami UML.

1. Wzorzec projektowy Model-Widok-Kontroler (MVC) - wzorzec ten został wykorzystany do oddzielenia warstwy prezentacji, logiki
2. Wzorzec projektowy Fabryka - wzorzec ten został wykorzystany do tworzenia obiektów różnych typów na podstawie podanego para



## 9.5Projekt bazy danych

### 9.5.1 Schemat

Poniżej przedstawiony jest schemat bazy danych aplikacji bankowej w trzeciej postaci normalnej (3NF):



```

| username (VARCHAR) |
| password (VARCHAR) |
+-----+

+-----+
|      accounts      |
+-----+
| id (INT)            |
| user_id (INT)       |
| account_number (VARCHAR) |
| balance (DECIMAL)   |
+-----+

+-----+
|      transactions   |
+-----+
| id (INT)            |
| account_id (INT)    |
| transaction_type (ENUM) |
| amount (DECIMAL)    |
| date_time (DATETIME)|
+-----+

+-----+
|      cards          |
+-----+
| id (INT)            |
| user_id (INT)       |
| card_number (VARCHAR) |
| expiry_date (DATE)   |
+-----+

+-----+
|      currencies     |
+-----+
| id (INT)            |
| code (VARCHAR)      |
| name (VARCHAR)      |
+-----+

+-----+
|      exchange_rates  |
+-----+
| id (INT)            |
| currency_id (INT)   |
| rate (DECIMAL)      |
| date (DATE)         |
+-----+

+-----+
|      payments       |
+-----+
| id (INT)            |
| user_id (INT)       |
| payment_type (ENUM) |
| amount (DECIMAL)    |
| date_time (DATETIME)|
+-----+

+-----+
|      beneficiaries   |
+-----+
| id (INT)            |
| user_id (INT)       |
| name (VARCHAR)      |
| account_number (VARCHAR) |

```

```

| account_number (VARCHAR) |
+-----+

+-----+
| notifications |
+-----+
| id (INT)      |
| user_id (INT) |
| message (VARCHAR) |
| is_read (BOOLEAN) |
+-----+

+-----+
| loans |
+-----+
| id (INT) |
| user_id (INT) |
| amount (DECIMAL) |
| interest_rate (DECIMAL) |
| start_date (DATE) |
| end_date (DATE) |
+-----+

```

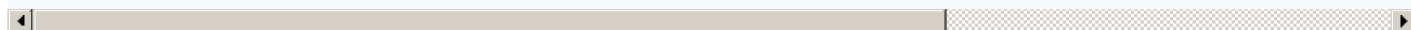
## 9.6 Projekt interfejsu użytkownika

Tutaj wkleić zdjęcia na githuba.

### 9.6.1 Lista głównych elementów interfejsu

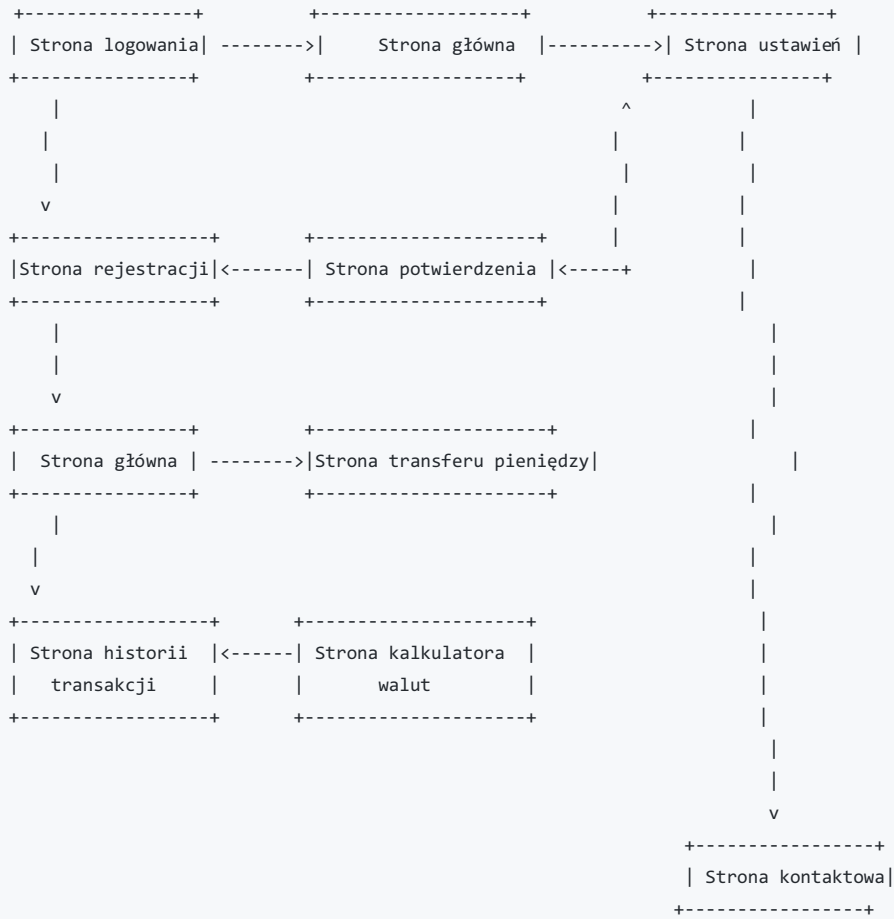
Poniżej przedstawiona jest lista głównych elementów interfejsu aplikacji bankowej:

- 1.Strona logowania - na tej stronie użytkownik może wprowadzić swoje dane logowania, takie jak nazwa użytkownika i hasło, aby uzyskać dostęp do systemu.
- 2.Strona główna - na stronie głównej użytkownik może zobaczyć podsumowanie swojego konta bankowego, takie jak saldo, ostatnie transakcje i informacje o kontach.
- 3.Strona transferu pieniędzy - na tej stronie użytkownik może dokonać transferu pieniędzy na inne konto bankowe poprzez wprowadzenie numeru konta i kwoty.
- 4.Strona historii transakcji - na tej stronie użytkownik może zobaczyć historię swoich transakcji, takich jak daty, kwoty i opisy.
- 5.Strona kalkulatora walut - na tej stronie użytkownik może obliczyć wartość jednej waluty w innej walucie, wykorzystując aktualne kursy.
- 6.Strona ustawień - na tej stronie użytkownik może zmienić swoje dane osobowe, takie jak adres, numer telefonu i adres e-mail, oraz konfigurację powiadomień.
- 7.Strona kontaktowa - na tej stronie użytkownik może znaleźć informacje kontaktowe do banku, takie jak numer telefonu, adres e-mail i adres strony internetowej.
- 8.Okno powiadomienia - na tym oknie użytkownik otrzymuje powiadomienia o transakcjach, zmianach w swoim koncie bankowym i innych ważnych informacjach.
- 9.Okno potwierdzenia - na tym oknie użytkownik otrzymuje potwierdzenie dokonania transakcji i może zaakceptować lub anulować transakcję.



### 9.6.2 Przejścia między głównymi elementami

Poniżej przedstawiony jest schemat blokowy przedstawiający przejścia między głównymi elementami interfejsu aplikacji bankowej:



Schemat przedstawia, że użytkownik zaczyna od strony logowania, skąd może przejść do strony rejestracji lub po wpisaniu poprawnych danych logowania do strony głównej. Z głównej strony użytkownik może przejść do innych głównych elementów interfejsu, takich jak strona transferu pieniędzy, strona historii transakcji, strona kalkulatora walut i strona kontaktowa. Z poziomu strony głównej użytkownik może także przejść do strony ustawień. Przejścia między elementami interfejsu mogą być wywołane przez kliknięcie przycisków lub opcji w menu.

### 9.6.3 Projekty szczegółowe poszczególnych elementów

Przykładowe projekty szczegółowe dla 4 głównych elementów interfejsu:

Formularz rejestracji użytkownika

ID: 1

Nazwa: Formularz rejestracji

Projekt graficzny:

Formularz rejestracji

Opis: Formularz służący do rejestracji użytkownika w systemie. Wymagane dane to: imię, nazwisko, adres e-mail, hasło.

Wykorzystane dane: dane wprowadzone przez użytkownika zostają zapisane w bazie danych.

Opis działania: po wprowadzeniu wszystkich wymaganych danych użytkownik klikając przycisk "Zarejestruj się" zostaje zapisany w bazie

Panel użytkownika

ID: 2

Nazwa: Panel użytkownika

Projekt graficzny:

Panel użytkownika

Opis: Panel służący do zarządzania kontem użytkownika. Użytkownik może tutaj zmienić swoje dane, hasło oraz zobaczyć historię transa

Wykorzystane dane: dane użytkownika pobierane są z bazy danych i wyświetlane w formularzach. Historia transakcji pobierana jest równ

Opis działania: po zalogowaniu użytkownik zostaje przekierowany na panel użytkownika. Klikając w odpowiednie opcje użytkownik może z

Formularz przelewu

ID: 3

Nazwa: Formularz przelewu

Projekt graficzny:

Formularz przelewu

Opis: Formularz służący do wykonania przelewu. Użytkownik wprowadza tutaj dane odbiorcy oraz kwotę przelewu.

Wykorzystane dane: dane wprowadzone przez użytkownika zostają zapisane w bazie danych. Przy wykonaniu przelewu system pobiera aktual

Opis działania: po wprowadzeniu danych użytkownik klikając przycisk "Wykonaj przelew" wykonuje przelew, a dane zostają zapisane w ba

Strona główna

ID: 4

Nazwa: Strona główna

Projekt graficzny:

Strona główna

Opis: Strona główna systemu bankowego. Zawiera informacje o saldzie konta oraz listę ostatnich transakcji.

Wykorzystane dane: dane o saldzie konta oraz listę ostatnich transakcji pobierane są z bazy danych.

Opis działania: po zalogowaniu użytkownik zostaje przekierowany na stronę główną.

## 9.7 Procedura wdrożenia

1. Testowanie aplikacji: Przed rozpoczęciem wdrożenia aplikacji bankowej należy przeprowadzić testy. Testy powinny obejmować weryfikację
2. Uruchomienie środowiska produkcyjnego: Po pomyślnym zakończeniu testów należy przygotować środowisko produkcyjne. Powinno ono zape
3. Przygotowanie bazy danych: Należy zapewnić, że baza danych jest przygotowana do działania z aplikacją bankową. Wszystkie potrzebne
4. Instalacja aplikacji bankowej: Po przygotowaniu środowiska produkcyjnego i bazy danych, należy zainstalować aplikację bankową. Ins
5. Testowanie aplikacji w środowisku produkcyjnym: Po zainstalowaniu aplikacji bankowej na serwerze produkcyjnym, należy przeprowadzi
6. Wdrożenie aplikacji: Po pomyślnym zakończeniu testów aplikacji w środowisku produkcyjnym, należy udostępnić aplikację użytkownikom
7. Szkolenie użytkowników: Należy zapewnić szkolenie dla użytkowników aplikacji bankowej. Szkolenie powinno obejmować instrukcje doty
8. Monitorowanie i utrzymanie aplikacji: Po wdrożeniu aplikacji bankowej, należy monitorować jej działanie i zapewnić regularne utrzy

## 10 Podsumowanie

### 10.1 Szczegółowe nakłady projektowe członków zespołu

Wszystko ja

ic Markdown to PDF don't worry your files are safe and you can login at [classic.markdowntopdf.com](https://classic.markdowntopdf.com).

## Privacy First

Unlike version 1 of Markdown to PDF your content is no longer stored in a file before being read/converted and subsequently removed from our servers. Instead your content is now sent securely via our API before returning the converted file so your data is never stored.

## GitHub flavoured styling by default



---

We now use GitHub flavoured styling by default.

## Image Support

---

Images are base64 encoded into the PDF document so images do not embed the remote source that could go offline rendering your image broken and of course do not require an internet connection.

## Next...

---

Up next we're working on Emoji support 🍌