Dokumentacja projektu

Skarbonka

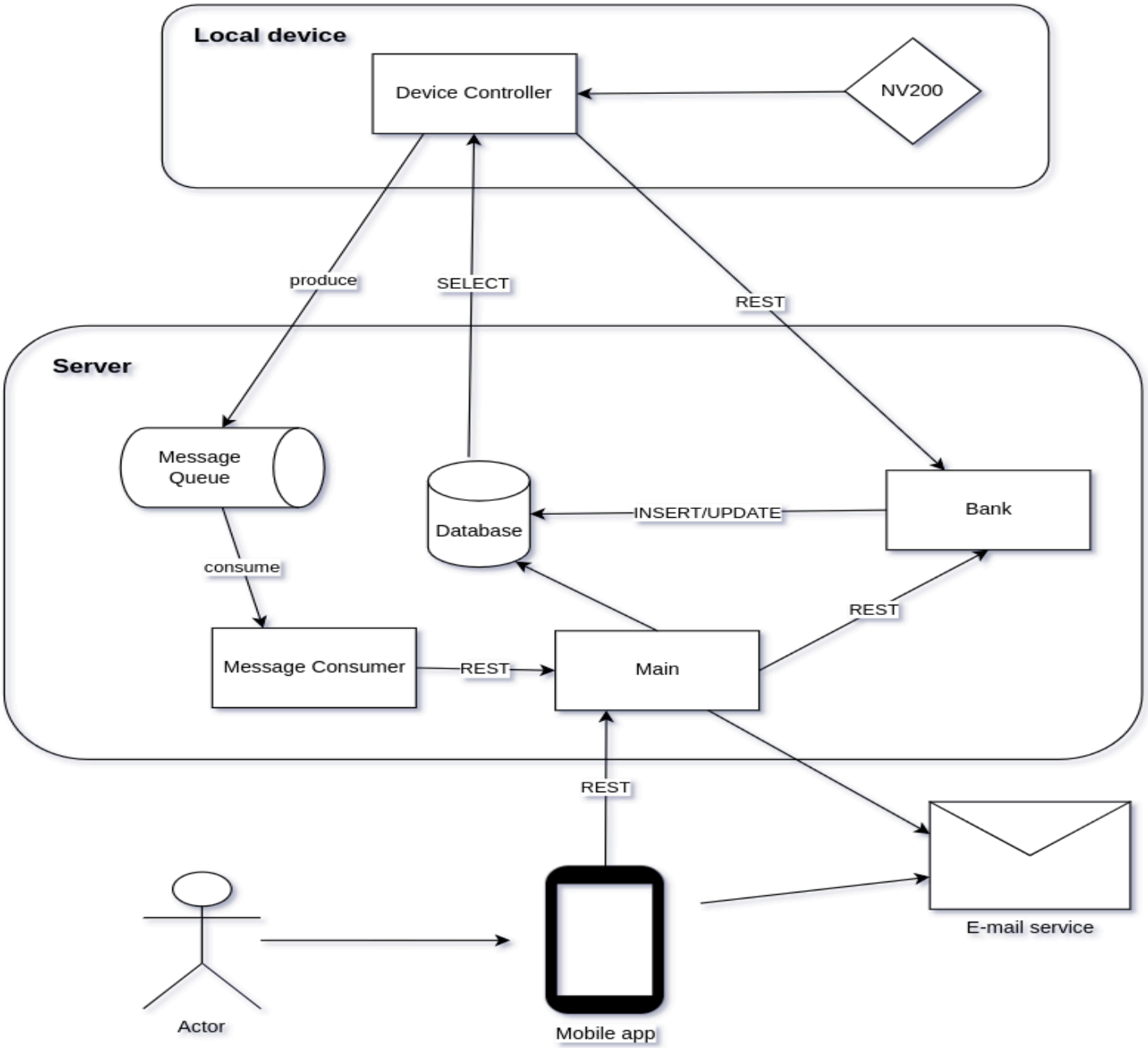
Dokumentację sporządzili:

Stefan Reszel 75696
Szymon Mikołajek 75691
Sebastian Krause 45647

Cel projektu

Celem aplikacji jest ułatwienie użytkownikom oszczędzania pieniędzy oraz zarządzania wpłatami i wypłatami za pomocą urządzenia NV200 firmy Innovative Technology oraz platformy mobilnej. System zapewnia możliwość monitorowania stanu środków, realizacji transakcji oraz przejrzystą komunikację z użytkownikiem.

Architektura



Lokalne Urządzenie

- Device Controller
    - Obsługuje komunikację z urządzeniem NV200 przy użyciu protokołu ccTalk.
    - Technologia: Python.
    - Przetwarza dane o wpłatach i wypłatach w urządzeniu NV200.
- NV200

    - Urządzenie płatnicze odpowiedzialne za przyjmowanie i wydawanie gotówki.

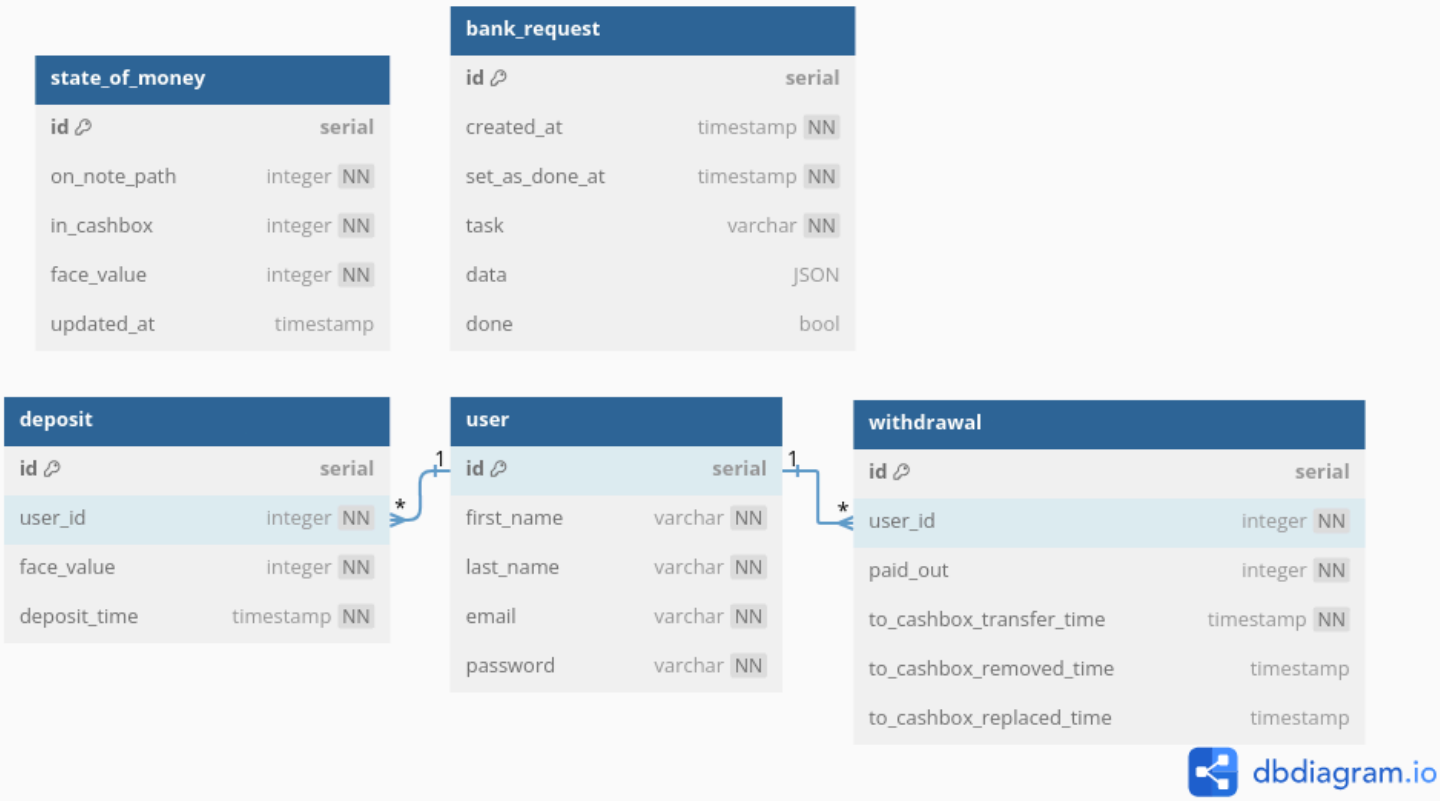
Serwer

- Message Queue
    - Narzędzie: RabbitMQ.
    - Obsługuje asynchroniczną komunikację między komponentami.
- Database
    - Narzędzie: PostgreSQL.
    - Przechowuje dane o użytkownikach, transakcjach, stanie gotówki i zadań dla urządzenia płatniczego.
- Main
    - Technologia: Python, FastAPI.
    - Główny moduł zarządzający aplikacją.
    - Obsługuje logikę biznesową, REST API oraz komunikację z bazą danych, bankiem i aplikacją mobilną.
- Bank
    - Technologia: Python, FastAPI.
    - Moduł realizujący transakcje pieniężne (np. wpłaty, wypłaty) oraz wysyła żądania do urządzenia płatniczego.

Aplikacja mobilna

- Technologia: React Native.
- Cechy:
    - Wieloplatformowość (Android, iOS).
    - Interfejs umożliwiający monitorowanie stanu środków, historię transakcji oraz inicjację wpłat/wypłat.
    - Autoryzacja i szyfrowanie danych.

Baza Danych



Usługa: **PostgreSQL**.

Schemat:

- **user**: Przechowuje dane użytkowników.
- **deposit**: Dane o wpłatach.
- **withdrawal**: Dane o wypłatach.
- **state_of_money**: Aktualny stan środków.
- **bank_request**: Zlecenia do urządzenia NV200.

Plan Projektu

Etap 1: Analiza Wymagań

1. Określenie wymagań funkcjonalnych i niefunkcjonalnych.

Etap 2: Projektowanie Systemu

1. Backend:
    - Projekt API dla aplikacji mobilnej i urządzenia.
    - Wybór struktury bazy danych (normalizacja, relacje).
    - Przygotowanie RabbitMQ do obsługi kolejek wiadomości.
2. Frontend:
    - Projekt interfejsu aplikacji mobilnej (prototypy, UX/UI).
3. Bezpieczeństwo:
    - Implementacja OAuth2 do autoryzacji użytkowników.
    - Szyfrowanie danych w bazie i transmisji danych (TLS/SSL).

Etap 3: Implementacja

1. Device Controller:
    - Implementacja obsługi NV200 w Pythonie.
2. Serwer:
    - Tworzenie REST API w FastAPI.
    - Obsługa RabbitMQ do komunikacji asynchronicznej.
    - Logika zarządzania transakcjami.
3. Baza Danych:
    - Migracje i implementacja schematu w PostgreSQL.
4. Aplikacja Mobilna:
    - Implementacja interfejsu w React Native.
    - Komunikacja z API serwera.

Etap 4: Testowanie

1. Testy jednostkowe (backend, frontend).
2. Testy integracyjne (np. połączenie NV200, RabbitMQ).
3. Testy użytkownika (UX, akceptacja).

Etap 5: Wdrożenie

1. Konfiguracja serwera produkcyjnego.
2. Wdrożenie aplikacji mobilnej do App Store i Google Play.
3. Monitorowanie i utrzymanie (logowanie, alerty).

Bezpieczeństwo

1. Autoryzacja i uwierzytelnianie:
    - Wykorzystanie OAuth2 do zarządzania sesjami.
    - Tokeny dostępu i odświeżania.
2. Szyfrowanie danych:
    - TLS/SSL dla komunikacji klient-serwer.
    - Szyfrowanie wrażliwych danych w bazie (np. hasła, dane osobowe).
3. Monitorowanie:
    - Wykorzystanie narzędzi takich jak Prometheus/Grafana.
    - Monitorowanie RabbitMQ i serwera FastAPI.

Technologie i Narzędzia

1. Backend: Python, FastAPI, RabbitMQ.
2. Frontend: React Native.
3. Baza Danych: PostgreSQL.
4. Narzędzia CI/CD: GitHub Actions, Docker.
5. Monitorowanie: Prometheus, Grafana.
6. Kontrola wersji: Git, GitHub.

Harmonogram

1. Tydzień 1-2: Analiza wymagań.
2. Tydzień 3-5: Projektowanie systemu.
3. Tydzień 6-12: Implementacja backendu, bazy danych i aplikacji mobilnej.
4. Tydzień 13-14: Testowanie systemu.
5. Tydzień 15: Wdrożenie aplikacji.

Wnioski

Projekt umożliwia stworzenie zaawansowanego systemu do zarządzania oszczędzaniem, łącząc nowoczesne technologie z intuicyjnym interfejsem mobilnym. Uwzględnienie bezpieczeństwa i skalowalności zapewnia długoterminowe funkcjonowanie aplikacji.