MedicaBARP – Dokumentacja

Spis treści

[1. Opis projektu 1](#_Toc154781686)

[2. Autorzy 1](#_Toc154781687)

[3. Założenia projektowe 1](#_Toc154781688)

[4. Ograniczenia i ryzyka projektowe 1](#_Toc154781689)

[5. Opis funkcjonalności 1](#_Toc154781690)

[6. Diagram związków encji 4](#_Toc154781691)

[7. Diagram przypadków użycia 4](#_Toc154781692)

[8. Diagramy życia encji 4](#_Toc154781693)

[9. Diagram przepływu danych 4](#_Toc154781694)

[10. Diagram sekwencji 4](#_Toc154781695)

# Opis projektu

# Autorzy

Alicja Kurczyk –

Bartosz Rabenda –

Piotr Ostrowski –

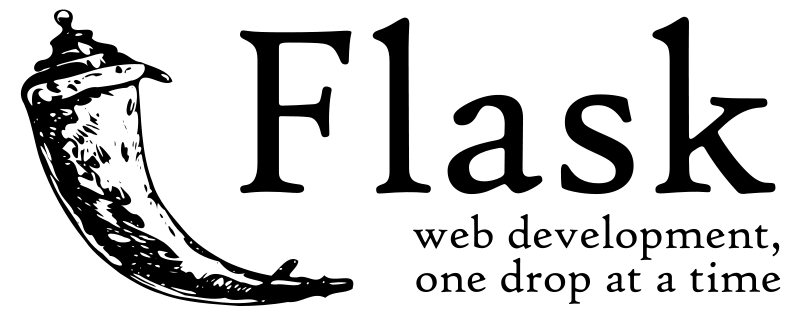
Remigiusz Zając – osoba odpowiedzialna za backend aplikacji napisany w języku Python. Rolą backendu jest połączenie wszystkich elementów i funkcjonalności w jedną współgrającą aplikację

# Założenia projektowe

# Ograniczenia i ryzyka projektowe

# Opis funkcjonalności

Placówka medyczna działa przy pomocy lokalnego serwera, biblioteki Flask używanej przez język Python. Oprócz serwera lokalnie działa również baza danych (o nazwie przychodnia) utworzona dzięki software’owi phpMyAdmin, do którego możemy uzyskać dostęp przy pomocy programu Xampp



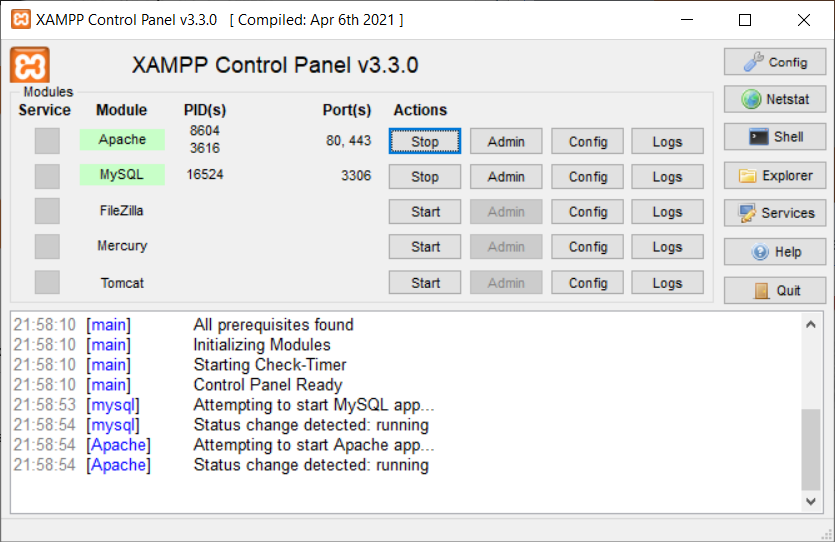
1. Logo Flask



1. Logo Xampp

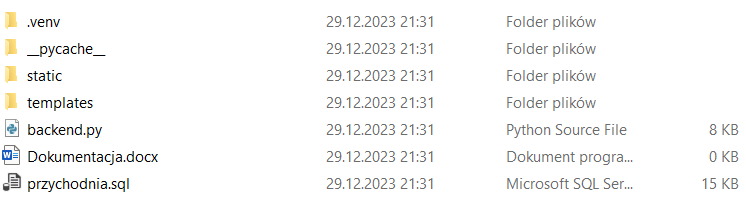
Proces uruchamiania aplikacji:

* Uruchamiamy program Xampp z wcześniejszym przygotowaniem bazy danych. Aby baza danych była dostępna, należy z dostępnych usług uruchomić moduły **Apache** oraz **MySQL**



1. XAMPP Control Panel

* W przypadku chęci zobaczenia bazy danych, możemy wejść do panelu phpMyAdmin poprzez kliknięcie przycisku Admin dla modułu MySQL lub w przeglądarce wpisać adres: [**localhost/phpmyadmin/**](http://localhost/phpmyadmin/)
* Kolejnym krokiem jest uruchomienie serwera Flask, dzięki któremu cała aplikacja będzie działać. Do tego potrzebne są pliki z aplikacji (github.com/RZajacUE/MedicaBARP/) oraz Wiersz Polecenia. W folderze MedicaBARP znajdziemy podfoldery oraz inne pliki



* .venv – wirtualne środowisko, na nim zainstalowane są wszystkie biblioteki dla języka Python, z których korzysta plik backend.py: Flask, mysql-connector-python, colorama oraz inne konieczne do poprawnego działania wyżej wymienionych
* \_\_pycache\_\_ - folder ze skompilowanymi plikami .py
* Static – pliki statyczne, arkusze stylów CSS oraz grafiki
* Templates – pliki HTML przedstawiające strony aplikacji
* Backend.py – backend aplikacji napisany w języku Python
* Dokumentacja.docx – obecny plik
* Przychodnia.sql – plik z bazą danych przychodnia
* Uruchamiając Wiersz Poleceń w folderze MedicaBARP należy wpisać polecenia: **.venv\Scripts\activate.bat** (aktywacja środowiska wirtualnego)  
  **python -m flask --app backend run** (uruchomienie serwera)
* W przeglądarce należy wpisać adres: **127.0.0.1:5000**

Aplikacja internetowa została podzielona na strony:

**Strona główna**Całość aplikacji, strona początkowa z odnośnikami do innych stron

**Logowanie**Logowanie polega na wpisaniu odpowiedniego numeru PESEL oraz wpisaniu odpowiadającego hasła. Występuje tutaj obsługa błędnego logowania, czyli w przypadku podania nieprawidłowego numeru PESEL lub nieprawidłowego hasła otrzymujemy odpowiedni komunikat.

**Rejestracja**Jeżeli użytkownik nie posiada swojego konta, może je utworzyć. Należy w formularzu wypełnić wszystkie dane. Również zastosowano obsługę błędnej rejestracji. Sprawdzane są pola (czy każde wymagane jest uzupełnione), poprawność hasła, numeru PESEL, adresu e-mail, numeru telefonu przy pomocy wyrażeń regularnych, poprawność powtórzonego hasła, czy któryś z użytkowników posiada ten sam numer PESEL oraz czy zostało zaakceptowane RODO

**Profil pacjenta**

**Wizyta**Miejsce, w którym można umówić wizytę z lekarzem. Każdy lekarz ma dodatkową informację o specjalizacji i numerze pokoju. Wizytę można ustalić o pełnej godzinie od 8:00 – 15:00

**Lekarze**

**Pakiety Medyczne**

**O nas**

**Kontakt i pomoc**

# Diagram związków encji

# Diagram przypadków użycia

# Diagramy życia encji

# Diagram przepływu danych

# Diagram sekwencji