MVP dla systemu zarządzania przychodnią

1. Moduł użytkowników:
   * Możliwość tworzenia kont dla lekarzy i pracowników przez administratora.
   * Podstawowe zarządzanie profilem użytkownika np. zmiana hasła
   * Możliwość rejestracji pacjenta
2. Moduł terminarza wizyt:
   * Wgląd dla lekarzy w ich harmonogram wizyt, oraz dodawanie dostępnych terminów
   * Możliwość dodawania, edycji i anulowanie wizyt przez lekarzy i recepcjonistów
3. Moduł placówek:
   * Możliwość dodawania i zarządzania placówkami
   * Możliwość przypisania pracowników do placówki
4. Moduł danych pacjenta:
   * Prowadzenie i wgląd w historie chorób pacjenta
   * Wgląd w historię recept i zwolnień pacjenta

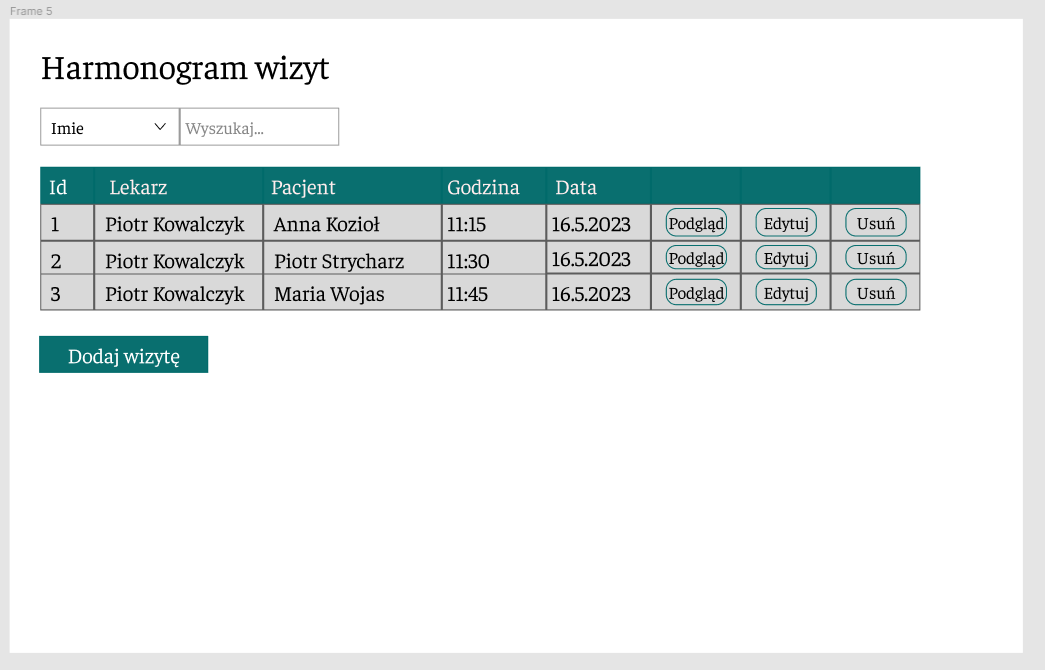
Mockupy

1. Cały layout (headder, sidebar, maincontent) z listą pracowników

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

1. Harmonogram wizyt (maincontent)



1. Formularz (maincontent)

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, design

Opis wygenerowany automatycznie

1. Logowanie (maincontet)

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

Lista zadań

1. Specyfikacja techniczna

* Backend: REST API w ASP.NET Core
* Frontend: React, dla mobilki React Native

1. Epic / stories:
   1. Epic: Konta użytwkoników

* Story: API – entites, models, controlers, services dla klinik, pracowników
* Story: testowanie w Postaman czy API komunikuje się z bazą i przekazuje odpowiednio nowo dodane operacje na danych
* Story: Frontend – komponent header
* Story: Frontend – komponent sidemenu
* Story: Frontend – komponenty main content list, formularzy dla placówek, pracowników
* Story: API/Frontend – Logowanie dla pracowników
  1. Epic: Harmonogram wizyt
* Story: API – entites, models, controlers, services dla harmonogramu i wizyt
* Story: testowanie w Postaman czy API komunikuje się z bazą i przekazuje odpowiednio nowo dodane operacje na danych
* Story: Frontend – komponent wizyty do harmonogramu
* Story: Frontend – komponenty main content dla harmonogramu, dodawania i edycji wizyty
  1. Epic: Historia leczenia
     + - Story: API – entites, models, controlers, services dla histori leczenia, recept i zwolnień
       - Story: testowanie w Postaman czy API komunikuje się z bazą i przekazuje odpowiednio nowo dodane operacje na danych
       - Story: Frontend – komponent main content dla historii leczenia pacjenta
  2. Epic: Rejestracja pacjenta (apk webowa)
     + - Strory: API – entites, models, controlers, services dla pracowników
       - Story: testowanie w Postaman czy API komunikuje się z bazą i przekazuje odpowiednio nowo dodane operacje na danych
       - Story: Frontend – komponent pacjenta dla listy
       - Story: Frontend – komponent main content listy pacjentów
       - Story: Frontend – komponent main content formularz dodania, edycji pacjanta
  3. Epic: Rejestracja pacjenta (apk mobilina)
     + - Story: Frontend – komponent pacjenta dla listy
       - Story: Frontend – komponent main content listy pacjentów
       - Story: Frontend – komponent main content formularz dodania, edycji pacjanta