



**Kolegium Nauk Przyrodniczych
Uniwersytet Rzeszowski**

Instrukcja obsługi

Asklepios

Wykonał:

Kamil Matusz, 122954

Spis treści

1. Opis aplikacji	3
2. Instalacja komponentów	3
2.1. Instalacja i uruchamianie komponentów Docker-a	3
2.2. Część backendowa .NET-a	4
2.3. Część frontendowa Vue.js	4
3. Wstępna konfiguracja	4
3.1. Konfiguracja bazy danych	4
3.2. Wysyłanie wiadomości email – SendGrid	4
3.3. Usługa Hangfire	5
3.4. Czyszczenie bazy danych	5
4. Role użytkowników	5
5. Moduły aplikacji	7

1. Opis aplikacji

Jest to system wspierający i oferujący wsparcie dla placówek medycznych. Jest to kompleksowe rozwiązanie umożliwiające zarządzanie i wspomaganie codziennych działań placówek medycznych, takich jak leczenie pacjentów, zarządzanie systemem czy przeprowadzanie badań. Aplikacja została zaprojektowana jako system full-stack, wykorzystujący .NET na backendzie oraz graficzny interfejs użytkownika (GUI) stworzony przy użyciu Vue.js i TypeScript.

2. Instalacja komponentów

2.1. Instalacja i uruchamianie komponentów Docker-a

Instalacja WSL2:

- Otwórz PowerShell jako administrator i uruchom:
wsl --install
- Polecenie automatycznie instaluje WSL2 i dystrybucję Linuxa (np. Ubuntu).

Ustawianie WSL2 jako domyślnej wersji:

- Wpisz w PowerShell:
wsl --set-default-version 2

Instalacja Dockera w WSL2:

- Wejdź na oficjalną stronę Docker i pobierz Docker Desktop for Windows.
- Uruchom pobrany instalator i postępuj zgodnie z instrukcjami.
- Upewnij się, że zaznaczysz opcję Enable WSL 2 Features podczas instalacji.

Konfiguracja integracji z WSL2:

- Otwórz Docker Desktop → Settings → Resources → WSL Integration.
- Włącz integrację z wybraną dystrybucją Linuxa, np. **Ubuntu**.
- Otwórz terminal WSL (np. Ubuntu) i wpisz:
docker --version
- Jeśli wszystko działa, zobaczysz zainstalowaną wersję Dockera.

Uruchamianie kontenerów projektu:

- W katalogu głównym projektu, w terminalu, należy wykonać następujące polecenie które zbuduje i uruchomi kontenery

docker-compose up -build

- Polecenie pobierze z repozytorium Dockera wszystkie wymagane obrazy oraz na ich podstawie stworzy wymagane kontenery wraz z zdefiniowanymi konfiguracjami.
- Aby sprawdzić poprawność działania kontenerów można sprawdzić to w GUI aplikacji Docker Desktop lub przy pomocy komendy:

docker ps

<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios</div>		Running (5/5)	0 seconds ago	<div><div></div><div>:</div><div></div></div>
<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios_web</div> <div>76af0227acb2 <div></div></div>	<div>asklepios-asklepios-web</div>	Running	<div>8080:80 <div></div></div>	<div>5 seconds ago</div> <div><div></div><div>:</div><div></div></div>
<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios_api</div> <div>cb06fa08317a <div></div></div>	<div>asklepios-asklepios-api</div>	Running	<div>5102:80 <div></div></div>	<div>0 seconds ago</div> <div><div></div><div>:</div><div></div></div>
<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios_postgres</div> <div>62a64f3b0116 <div></div></div>	<div>postgres</div>	Running	<div>5432:5432 <div></div></div>	<div>5 seconds ago</div> <div><div></div><div>:</div><div></div></div>
<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios_seq</div> <div>c62a129b103f <div></div></div>	<div>datalust/seq</div>	Running	<div>5341:80 <div></div></div>	<div>5 seconds ago</div> <div><div></div><div>:</div><div></div></div>
<input type="checkbox"/>	<div><div></div><div></div><div></div></div>	<div>asklepios_rabbitMQ</div> <div>768a262662be <div></div></div>	<div>rabbitmq:3-management</div>	Running	<div>15672:15672 <div></div></div> <div>Show all ports (3)</div>	<div>5 seconds ago</div> <div><div></div><div>:</div><div></div></div>

Rys. 2 – Uruchomione kontenery narzędzia Docker

- Aby wstrzymać uruchomione kontenery, należy użyć polecenia:

docker-compose stop

2.2. Część backendowa .NET-a

Część backendowa aplikacji została zkonteneryzowana przy użyciu Dockera, co umożliwia uruchomienie jej w izolowanym środowisku. Usługa jest dostępna lokalnie na porcie 5102 i może być wywoływana pod adresem **http://localhost:5102**

2.3. Część frontendowa Vue.js

Część frontendowa aplikacji została zkonteneryzowana przy użyciu Dockera, co zapewnia izolowane środowisko uruchomieniowe. Aplikacja jest dostępna lokalnie na porcie 8080 i można ją otworzyć pod adresem **http://localhost:8080**

3. Wstępna konfiguracja

3.1. Konfiguracja bazy danych

Na podstawie zaimplementowanych migracji i seederów, podczas pierwszego uruchomienia aplikacji w bazie danych zostaną automatycznie utworzone wszystkie wymagane tabele oraz wypełnione przykładowymi danymi, jeśli w pliku `appsettings.json` opcja włączająca seedery zostanie ustawiona na `true`. Dzięki temu aplikacja będzie gotowa do działania bez potrzeby ręcznej konfiguracji struktury bazy danych. W tym celu w projekcie **Asklepios.Api** należy przejść w pliku `appsettings.json` zmienić konfigurację na:

```
"SeedData": {
  „Enable”: true
}
```

3.2. Wysyłanie wiadomości email – SendGrid

Funkcjonalność wysyłania wiadomości e-mail jest realizowana za pomocą usługi SendGrid, która wymaga podania klucza autoryzacyjnego. Klucz ten należy skonfigurować w ustawieniach aplikacji przed użyciem tej funkcji. W tym celu w projekcie **Asklepios.Api** należy przejść do folderu `Properties` a następnie w pliku `secret.json` w polu `keySensGrid` umieścić kod autoryzacyjny.

Kod autoryzacyjny:

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

3.3. Usługa Hangfire

Funkcjonalność wykonywania zadań w tle realizowana jest przy pomocy usługi Hangfire, która wymaga ręcznego włączenia w ustawieniach aplikacji. Aby włączyć usługę należy w pliku **appsettings.json** przejść do sekcji hangfire i zmienić wartość na **true**

```
„Hangfire”: {  
    „Enable”: true  
}
```

3.4. Czyszczenie bazy danych

Aby wyczyścić bazę danych i przywrócić ją do stanu początkowego, najlepiej usunąć cały kontener Dockera związany z bazą danych i ponownie go uruchomić. Dzięki temu, na podstawie zaimplementowanych migracji i seederów, baza danych zostanie automatycznie odtworzona ze wszystkimi wymaganymi tabelami i przykładowymi danymi, pod warunkiem że opcja włączająca seedery jest ustawiona na true w pliku appsettings.json.

4. Role użytkowników

Aby skorzystać z aplikacji niezbędne jest zalogowanie się na jedno z kont, stworzone przez administratora aplikacji lub samemu zarejestrować konto z uprawnieniami pacjenta przychodni. Po uruchomieniu aplikacji, otwiera się ekran logowania, gdzie użytkownik posiadający konto służbowe musi podać swój adres email oraz hasło.



Asklepios - System Zarządzania

Logowanie

E-mail użytkownika
admin@asklepios.com

Hasło
.....

ZALOGUJ SIĘ

PANEL REJESTRACJI

Rys. – Panel logowania

Asklepios - System Zarządzania

Rejestracja do przychodni

E-mail użytkownika

johndoe@wp.pl

Hasło

.....

Potwierdź hasło

.....

ZAREJESTRUJ SIĘ

PANEL LOGOWANIA

Rys. Panel rejestracji

Po zalogowaniu każdy użytkownik jest przenoszony na stronę główną, skąd może przejść do widoku, gdzie może sprawdzić swoje dane lub zmienić hasło.

Twój profil

Twoje dane

Poniziej znajdują się dane Twojego konta.

Email

kamilmatusz@asklepios.com

Rola

Lekarz

Konto stworzone

19.10.2024

Status konta

Aktywne

Zmiana hasła

Poniziej możesz zmienić hasło do swojego konta.

ZMIEN HASŁO

Rys. Panel z danymi konta użytkownika

Zdefiniowane konta wraz z danymi logowania:

Email	Hasło	Rola
admin@asklepios.com	password	Administrator
kamilmatusz@asklepios.com	password	Lekarz
miloszmichalski@asklepios.com	password	Lekarz
annapajak@asklepios.com	password	Pielęgniarka
andrzejkowalski@asklepios.com	password	Lekarz
martapiotrowska@asklepios.com	password	Pielęgniarka
lukasznawrocki@asklepios.com	password	Pacjent
grzegorzadamczyk@asklepios.com	password	Lekarz
barbaranowak@asklepios.com	password	Pielęgniarka

5. Moduły aplikacji

Asklepios definiuje role użytkowników, na bazie których definiowany jest dostęp do usług i modułów. Każda z ról posiada odrębne uprawnienia i dostęp do określonych modułów oraz widoków aplikacji:

Administrator

Administrator ma pełny dostęp do wszystkich modułów i widoków aplikacji. Może zarządzać użytkownikami systemu, konfiguracją aplikacji oraz danymi medycznymi.

Lekarze i pielęgniarki

Użytkownicy przypisani do ról lekarzy lub pielęgniarek mają dostęp do modułów związanych z zarządzaniem danymi medycznymi, w tym wizytami, kartotekami pacjentów oraz innymi elementami powiązanymi z ich specjalizacją. Dodatkowo mogą zarządzać danymi swoich kont użytkownika.

Pacjenci przychodni

Pacjenci przychodni mają dostęp wyłącznie do widoku zawierającego informacje o ich wizytach w przychodni, w tym daty, godziny i nazwiska przypisanych lekarzy, ponadto jak każdy z użytkowników posiadają możliwość zarządzania swoimi kontami.