**Technologie Sieciowe**

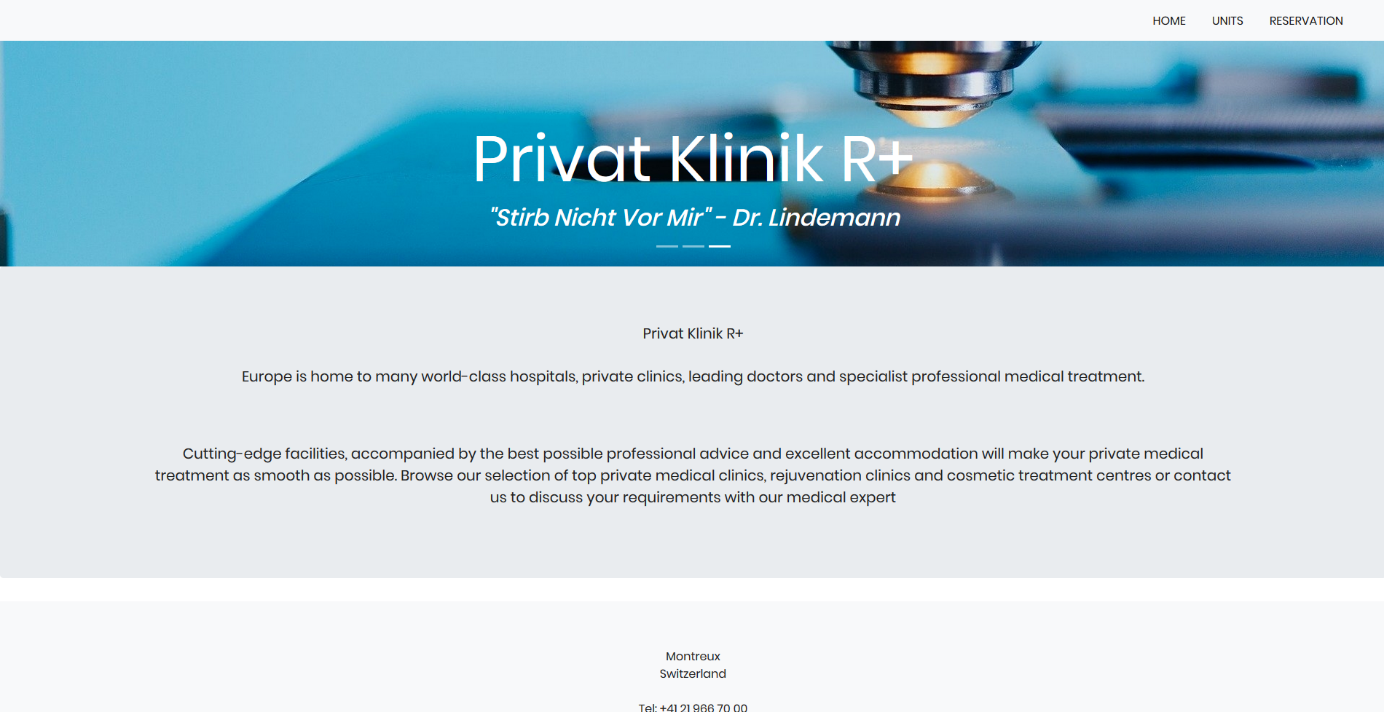
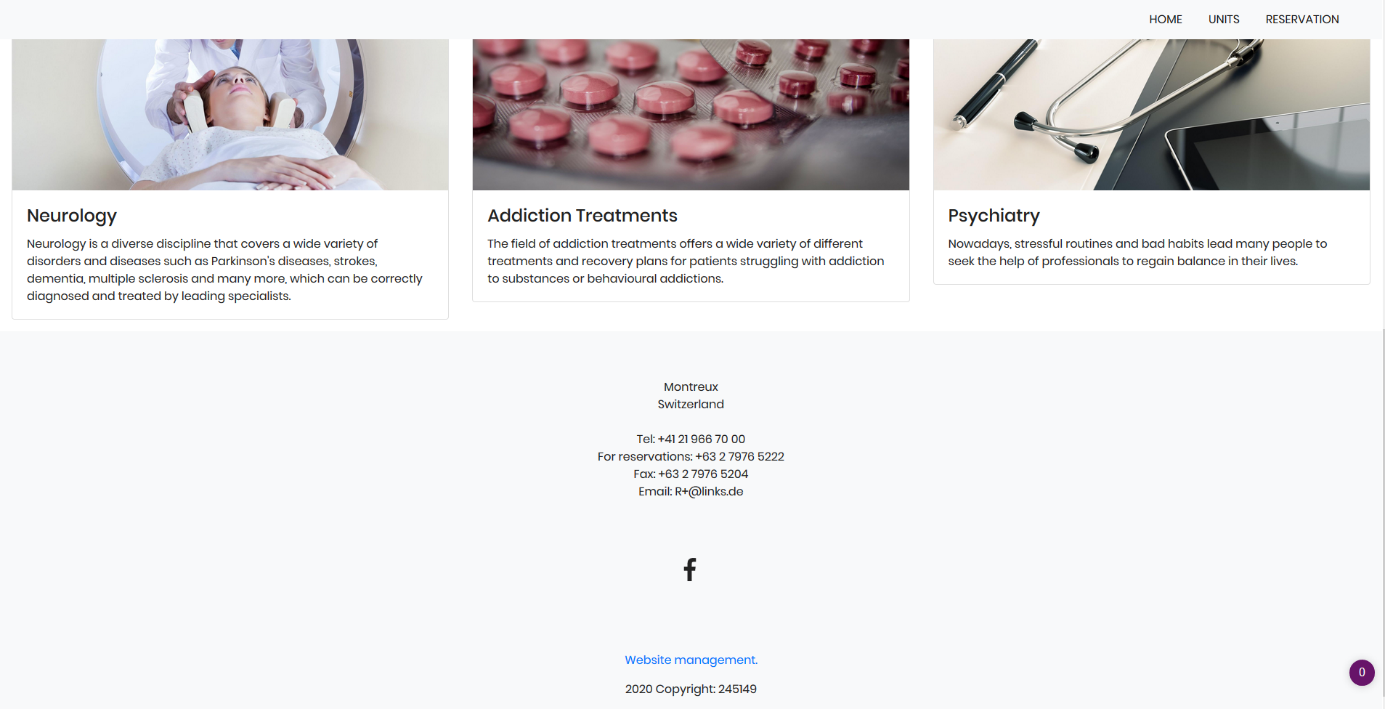
Projekt 1, ClinicWeb

Arkadiusz Kowal 245149

P01-85a (czw. 15:15)

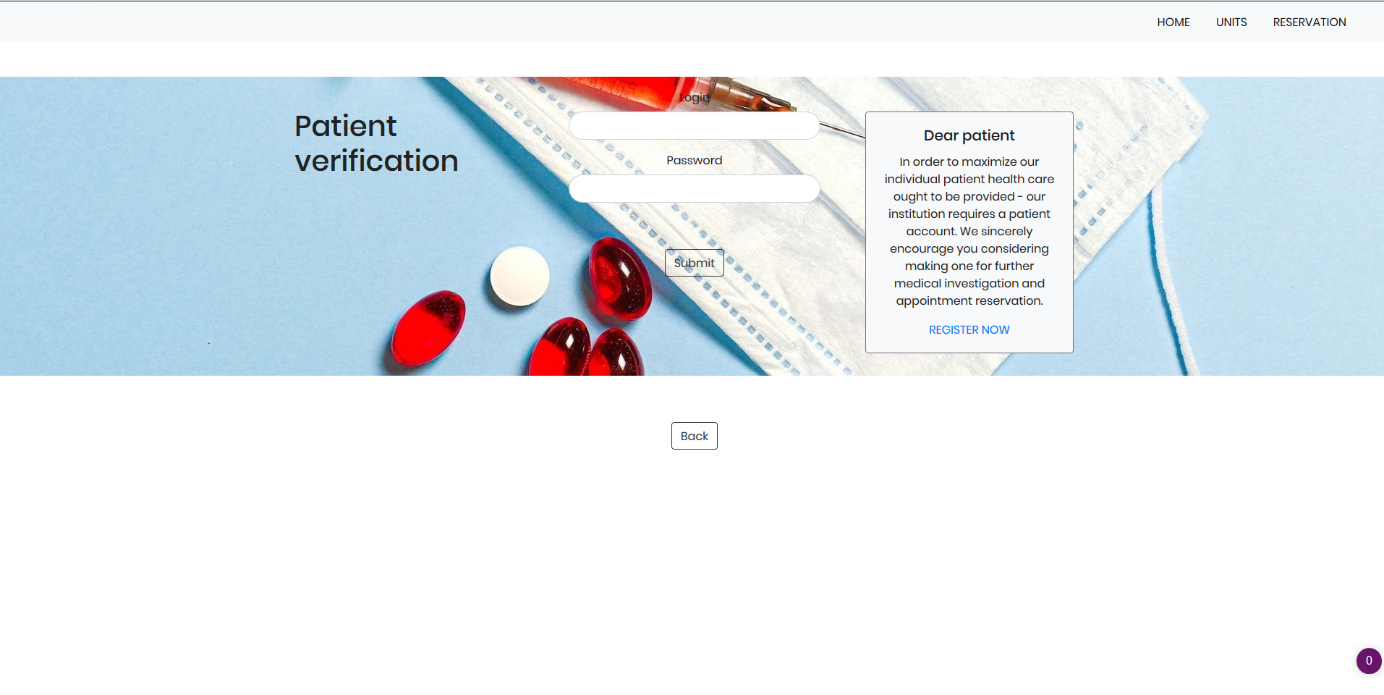
*Repo:* *https://github.com/245149/ClinicWeb*

# Opis działania projektu:

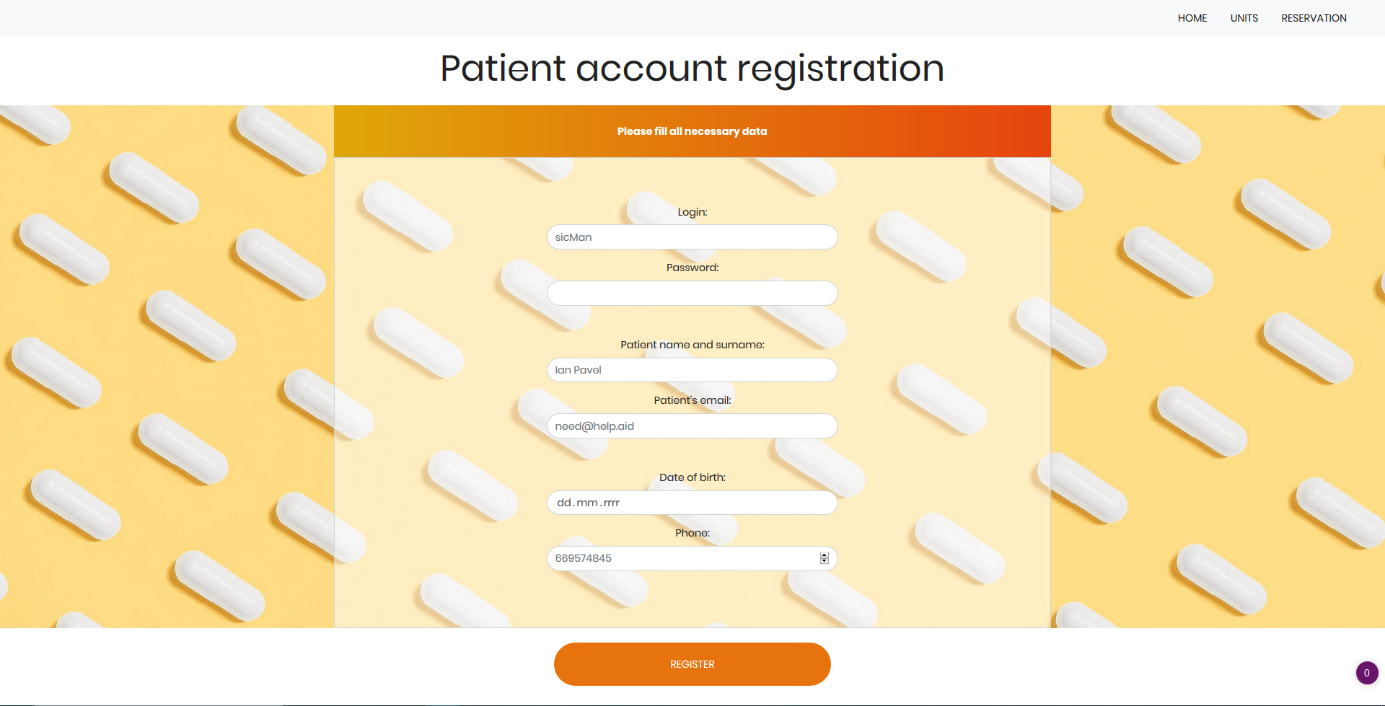
Strona główna aplikacji jest przede wszystkim informatywna, tak samo, jak zakładka ‘Units’ – obie te strony zawierają jedynie (oprócz nav-bar’a) – odsyłacz do części administracyjnej strony, która przenosi do logowania administratora.

Screen 1. index.html (fragment)

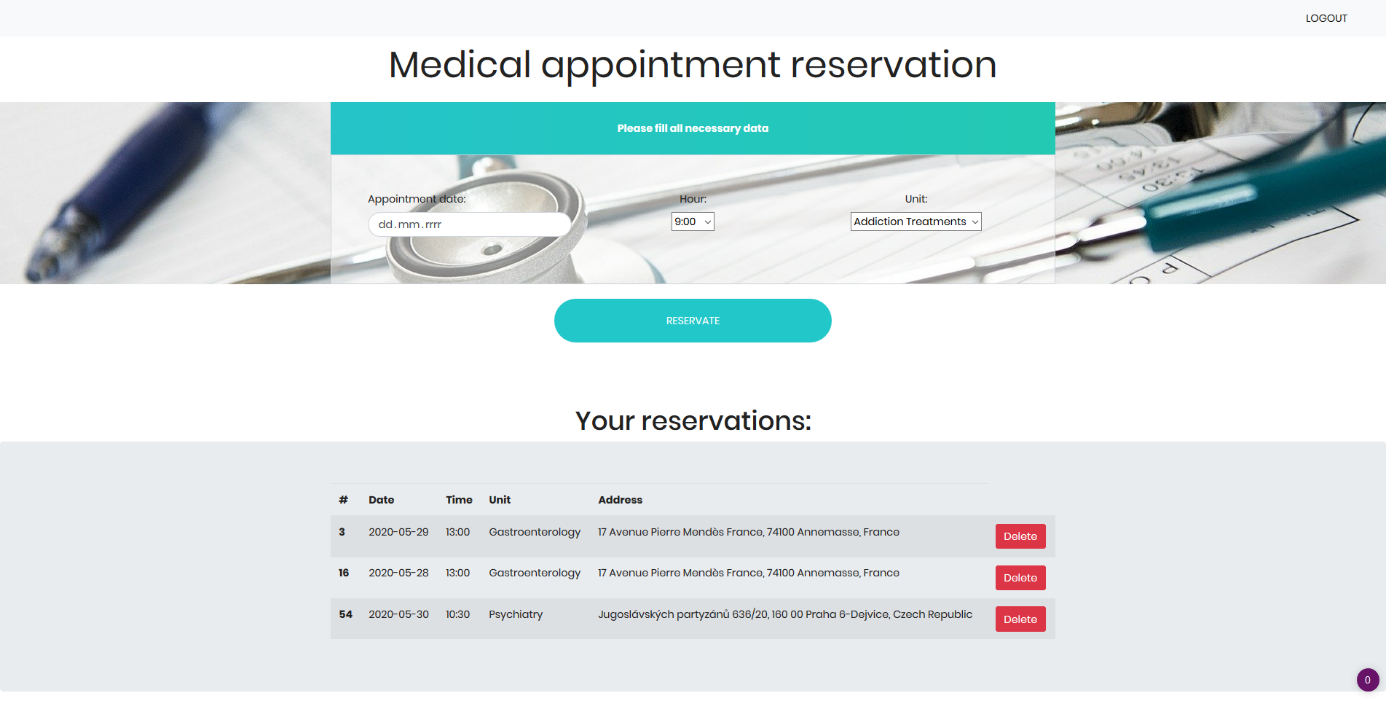
Screen 2. units.html (fragment)

Strona rezerwacji wizyty lekarskiej to sposób weryfikacji użytkownika (pacjenta) – jeżeli takowy posiada w ogóle konto na serwerze – w bazie danych. Jeżeli pragnie on założyć konto, przechodzi do zakładki ‘REGISTER NOW’.

Screen 3. logowanie użytkownika (pacjenta)

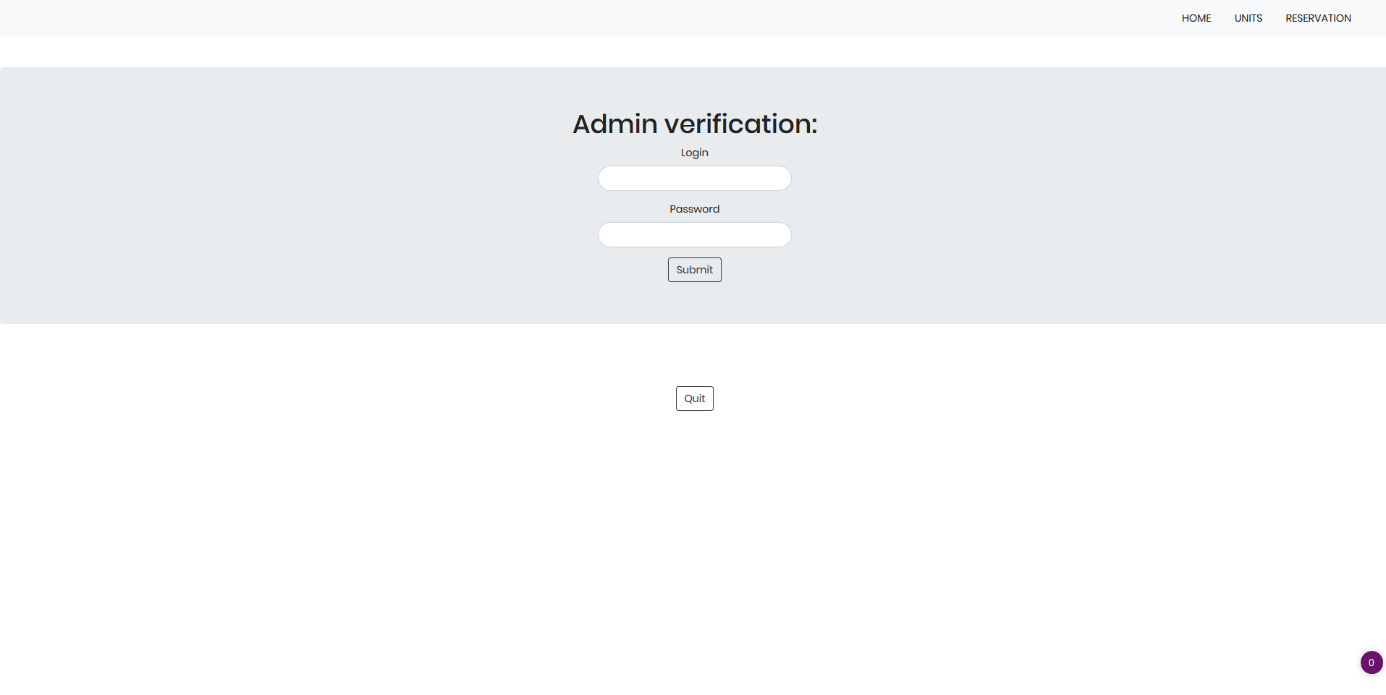
Po przejściu do strony rejestracji użytkownika pojawia się prosty formularz, w którym pacjent uzupełnia niezbędne dane – podczas wprowadzania danych do jakichkolwiek z rubryk w aplikacji brak jest możliwości pozostawienia ich pustych – odpowiednia informacja zostaje wyświetlona, a hasło powinno zawierać co najmniej 5 znaków. Po pomyślnej rejestracji uzyskiwane jest potwierdzenie.

Screen 4. Rejestracja pacjenta

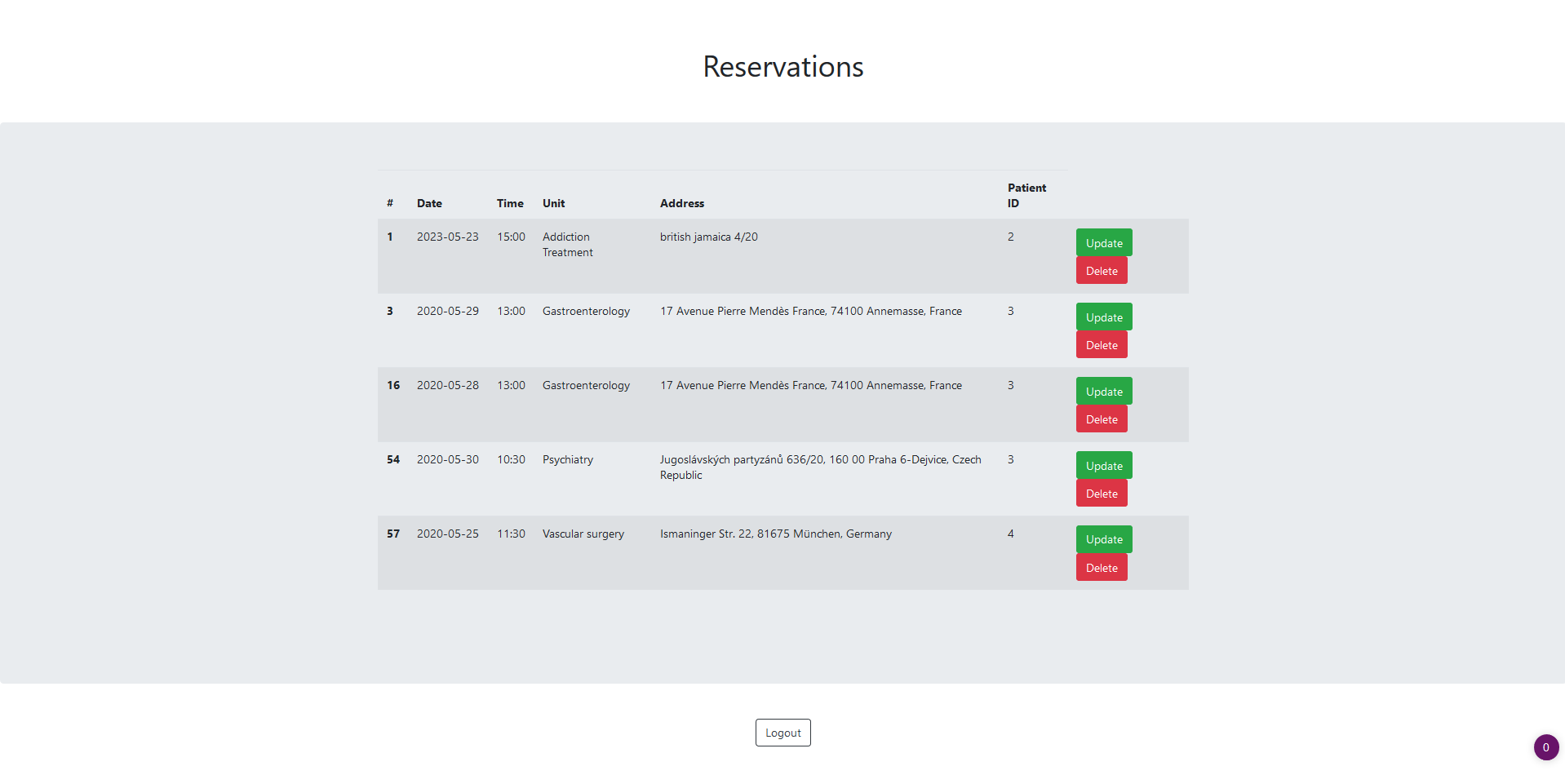
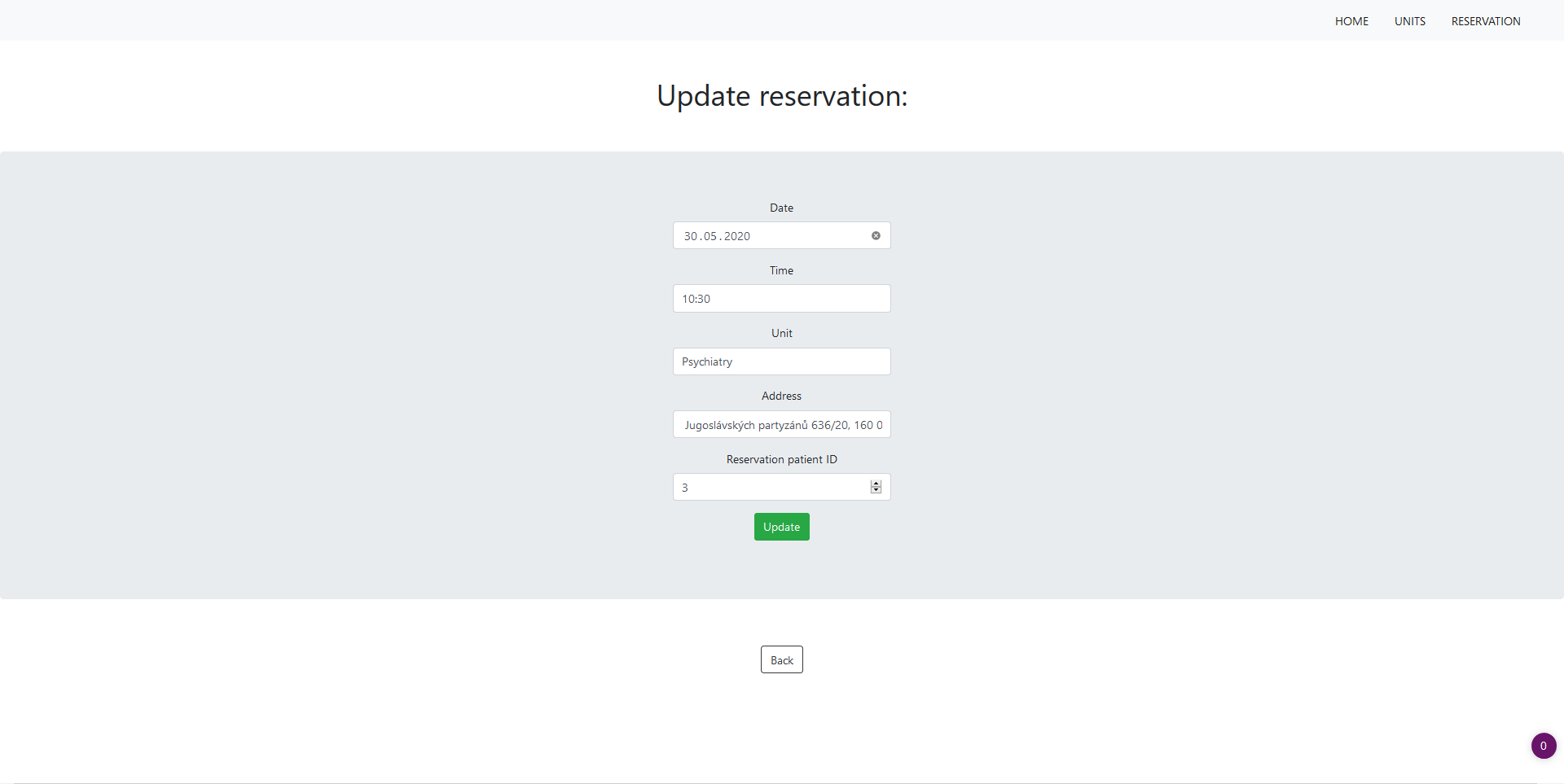
Po zalogowaniu się na istniejące konto pacjenta, ukazuje się strona rejestracji na medyczne wizyty wraz z tabelą tych już zarezerwowanych z możliwością ich usunięcia, jak i tych, które miały już miejsce w przeszłości. Dla wizyt w przeszłości przycisk ‘Delete’ pozostaje nieaktywny. Tabela jest na bieżąco aktualizowana. Po wprowadzeniu niepoprawnych danych (także tych, które wskazują na termin już zajęty) – pacjent zostaje poinformowany i ukazane zostają terminy już zajęte na poszczególnych wydziałach.

Screen 5. Panel użytkownika

Administrator posiada funkcje usuwania wszystkich pozycji w bazie danych oraz wprowadzania do nich zmian – po jego weryfikacji podczas logowania.



Screen 6. Panel logowania - admin



Screen 7. panel aktualizacji wybranej wizyty

Screen 8. Panel administratora

\*Aplikacja jest w pełni responsywna pod względem zmiany rozdzielczości urządzenia, na którym odtwarzane są strony html – dostosowana jest do tabletów/smartfonów.

# Zwięzły opis kodu:

Kod aplikacji zorganizowany jest w pakiety o ukierunkowanej funkcjonalności:

* DbUtil – jest to pakiet z klasami odpowiedzialnymi za wykonywanie w pierwszej linii operacji na bazie danych za pomocą składni SQL odpowiednio dla kompetencji admina oraz pacjenta oraz odpowiednie ich mapowanie;
* Entity – jest to pakiet z klasami o polach takich, jak odpowiadające im atrybuty w bazie danych, które są wykorzystywane do tworzenia ich instancji na bazie informacji pobranych od pacjenta/admina;
* Servlet – pakiet obejmujący klasy m.in. zbierające dane z formularzy, odpowiadające częściowo za autoryzację, operujące na parametrach obecnych   
  na \*.jsp, działania na ciasteczkach, wywołanie metod DButil, oraz wysyłanie danych obiektów do prezentacji na poszczególnych html;
* Security – jest to pakiet zawierający jedynie @Bean do zaimplementowania hashowania od Spring Security – ‘BCrypt’;
* \*.jsp = html + java w najbardziej ograniczonym zakresie – wyświetlanie przysłanych od Servletów atrybutów - w najbardziej obciążającym momencie za pomocą pętli ‘for each’.

# Opis bazy danych:

Całościowa logika aplikacji została zawarta po stronie javy/js/html – w bazie danych znajdują się 2 proste tabele:

1. Tabela użytkowników – zawierająca ich dane osobiste oraz pasy do konta (jeżeli istnieje); login jest unikalnym atrybutem;
2. Tabela wizyt – zawiera dane spotkania, które określił w formularzu użytkownik – powiązana jest z jego id;

W bazie rozróżnia się 2 użytkowników:

1. Admin – autoryzowany, nieograniczony dostęp oraz funkcje (root);
2. clinicClient – jako osoba o ograniczonej kompetencji, która ma dostęp do: rejestracji konta pacjenta, dodawania wizyt oraz ich usuwania (odpowiednio do tego dodatkowo: widoki jako ‘SELECT’) – dot. tylko dwóch opisanych wyżej tabel.