

Opis projekta: Zakazivanje termina u ordinaciji

Kratak opis

Aplikacija treba da omogući korisnicima da zakazu termin u stomatološkoj ordinaciji. Potrebno je da postoje 4 role, tj vrste korisnika, i to: administrator, lekar, prijavljen korisnik, neprijavljen korisnik. U zavisnosti od tipa korisnika koji posećuje sajt razlikuje se pristup delovima aplikacije kao i sam izgled.

Potrebno je da sadrži sledeće stranice:

- Pocetna: Sadrži opis ordinacije, poslednje vesti o ordinaciji, kontakt informacije, informacije o lokaciji sa mapom, listu lekara i nešto o njima, linkove do ostalih stranica.
- Prijava / Registracija: Strana za prijavljivanje (registrovanje) korisnika.
- Pregled termina: Kalendarski i tabelarni pregled zakazanih / slobodnih termina. Razlikuje se u odnosu na to koji je korisnik posetio stranu. Sadrži formu ili link do forme za zakazivanje termina.
- Pregled korisnika: Dostupna administratoru i lekaru. Mogućnost da se kreiraju, menjaju, brisu korisnici. Standardna tabela sa korisnicima.
- Profil korisnika: Pregled prethodnih i budućih termina, pregled poslatih zahteva za zakazivanje termina.
- Pregled zahteva za zakazivanje: Dostupna administratoru i lekaru. Lekar može da vidi zahteve upućene samo njemu/njoj. Administrator vidi sve zahteve. Zahtevi mogu da se odobre ili odbiju.
- Pregled prostorija / stomatoloških stolica: Dostupna administratoru. Mogućnost da odredi koji lekar radi u nekoj prostoriji u određeno vreme.
- Pregled vesti: Dostupna administratoru. Upravljanje vestima sa početne stranice, dodavanje, izmena, brisanje.

Scenario - Neprijavljen korisnik

Dolazi na početnu stranicu na kojoj ima osnovne informacije o ordinaciji i linkove do stranice za pregled termina i stranice za prijavu, odnosno registraciju. Na stranici za pregled termina ima informacije o slobodnim terminima. Ako želi da zakaze termin, korisnik se vodi do stranice za registraciju. Tek nakon registrovanja može da zakaze termin.

Scenario - Prijavljeni korisnik

Nakon prijavljivanja, korisnik ima mogućnost slanja zahteva za zakazivanje termina. Popunjava formu u kojoj bira datum, vreme i lekara kod koga želi da zakaze pregled.

Potrebno je da lekar odobri zahtev nakon cega se korisnik obavestava o tome. Na stranici sa terminima ima pregled slobodnih termina i vidi svoj termin (odobren ili samo poslat zahtev).

Scenario - Lekar

Nakon prijavljivanja dodaje se dostupnost stranice sa pregledom korisnika. Na stranici sa korisnicima vidi listu korisnika i link do njihovih profila. Vidi da li korisnik ima zakazan termin. Moze da odobri ili odbije zahtev za zakazivanje koji su upuceni njemu/njoj. Na stranici za pregled termina ima pregled slobodnih i zakazanih termina, kao i mogucnost da doda novi zahtev, zahtev se automatski odobrava, potrebno je da odabere samo vreme i korisnika.

Scenario - Administrator

Ima sve mogucnosti kao lekar i dodaje se pristup stranicama za upravljanje prostorijama i upravljanje vestima sa pocetne stranice. Korisnike, prostorije i vesti moze da dodaje, menja i brise. Pored obicnih korisnika ima pregled lekara i drugih administratora. Ima mogucnost da vidi, odobri i odbije zahteve poslate svim lekarima. Zahtev za zakazivanje koji kreira se administrator se automatski odobrava, a potrebno je da odabere vreme, datum, lekara i pacijenta.

Tehnologije koje se koriste za izradu aplikacije

NodeJS - backend, upravljanje bazom podataka, renderovanje stranica

ExpressJS - biblioteka za lakse upravljanje backend-om

MySQL - baza podataka

HTML, CSS - formatiranje i stilizovanje stranica

JavaScript - dodavanje funkcionalnosti stranicama na korisnickoj strani

Bootstrap - biblioteka za dodatno stilizovanje stranica

jQuery - biblioteka za dodatne funkcionalnosti

Moguće alternativne tehnologije za izradu aplikacije

React - framework koji omogućava laksu izradu i dodatne funkcionalnosti stranicama. Daje mogucnost koriscenja biblioteka kao sto su MaterialUI i Semantic. Pruza bolju kontrolu stanja aplikacije na korisnickoj strani. Negativna strana je sto bi se koristio REST API, a ne server-side rendering.

Elm - programski jezik koji se kompajlira u JavaScript, daje vecu garanciju da nece doci do runtime gresaka. Takodje client-side rendering + REST API.

Laravel - Daje veliki broj "out-of-the-box" resenja, kao sto su autentifikacija korisnika, rutiranje, renderovanje HTML-a.

MongoDB - noSQL baza podataka

Moguće dodatne funkcionalnosti i dodaci projektu

Upravljanje početnom stranom sajta - Administrator pored vesti može da kontrolise izgled početne strane, da menja slike, informacije o ordinaciji i slično. Nesto nalik prostijoj verziji CMS-a.

iOS i Android aplikacija - Izrađene u ReactNative-u. Korisnik bi imao mogućnost da zakaze termin, vidi osnovne informacije. Lekar bi mogao da pregleda termine, zakaze termin. Mogle bi da se implementiraju notifikacije, uz pomoć FCM-a, koje bi stizale korisniku kada se odobri zahtev ili lekaru kada pristigne zahtev. Eventualno bi mogao da se implementira chat kao direktna komunikacija lekara i pacijenta.