Smart Home

Tehnička dokumentacija

Verzija 1.1

Studentski tim: Hrvoje Jurić

Dario Šimunović

Petar Jerković

Marko Malkoč

Martin Modrušan

Marta Tkalec

David Gorup

Nastavnik: Marko Jurčević

Sadržaj

[1. Opis razvijenog proizvoda 3](#_Toc31030636)

[2. Tehničke značajke 4](#_Toc31030637)

[3. Upute za pokretanje sustava 10](#_Toc31030638)

[4. Upute za korištenje 11](#_Toc31030639)

[5. Literatura 19](#_Toc31030640)

# Opis razvijenog proizvoda

U ovom projektu potrebno je u timu od 7 osoba izraditi web aplikaciju koja služi kao grafičko sučelje za upravljanje stanjem u kući na daljinu. Aplikacije se povezuje na niže navedeni model Arduina koji je fizički spojen sa senzorima i kontrolerima pomoću kojih se može utjecati na trenutno stanje u kući.

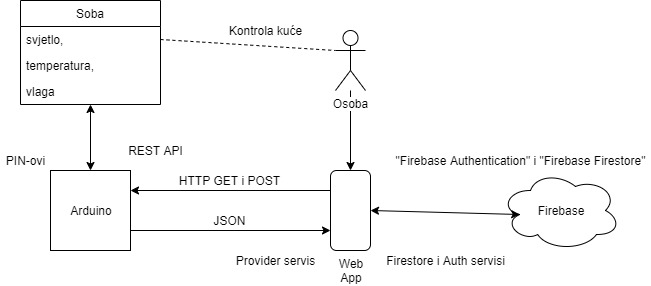
Od fizičkih komponenti koristi se 1 senzor za vlagu i temperaturu, te 1 senzor za svjetlo izveden otpornikom. Što se kontrolera tiče koristi se grijač(demonstracija lampicom, lampica upaljena/ugašena), lampica kao indikator upravljanja svjetlom te ventilator(demonstracija lampicom, lampica upaljena/ugašena).

Sve te komponente spojene su na M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS (u daljnjem tekstu Arduino) koji upravlja kontrolerima te dobiva stanje okoline iz senzora te preko http šalje sve informacije web aplikaciji. Preko http dobiva i informacije od web aplikacije za pokretanje pojedinih kontrolera.

Kao server gdje se vrti aplikacija trenutno se koristi Node.js. Za autentifikaciju i bazu podataka se koristi Firebase.

# Tehničke značajke

**Rad sustava**



Slika 1: Rad sustava

Sustav možemo podijeliti u dvije cjeline: Arduino i Web aplikacija. Arduino je odgovoran za direktnu komunikaciju sa kontrolerima i senzorima. Također je odgovoran za slušanje zahtjeva od strane Web aplikacije. Web aplikacija je odgovorna za efikasan grafički prikaz informacija o kući, te čuvanje određenih podataka o korisniku koristeći Firebase.

**Rad s Firebase-om**

U početku korisnik se treba registrirati pri čemu će unijeti svoju email adresu, korisničko ime i lozinku. Taj će se zahtjev poslati „Firebase Authentication“-u i kako bi se stvorio novi korisnik, te će se poslati „Firebase Firestore“-u svoj identifikator kako bi se stvorio novi dokument u kojem će biti pohranjeni korisnički podatci od kojih je jedan njegovo korisničko ime.

Nakon registracije korisnik treba verificirati svoju email adresu. Verifikacija se vrši tako da se Auth servis pošalje zahtjev na „Firebase Authentication“ da pošalje email s poveznicom na stranicu za verificiranje email adrese.

Poslije registracije korisnik se može prijaviti u Web aplikaciju svojom email adresom i lozinkom. Ako je korisnik prijavljen prije verifikacije moći će pristupiti svojoj korisničkoj stranici i stranici za verifikaciju. Ako je korisnik prijavljen nakon verifikacije može ući u pregled izbora Arduina. Na pregledu izbora arduina korisnik može izabrati svoj Arduino po njegovom zadanom imenu te će time: na svoj dokument u „Firestore Firebase“ pridodati referencu na dokument odabranog Arduina u bazi, pridodati svoj identifikator korisnicima u listi za čekanje odobrenja koja se nalazi u dokumentu odabranog Arduina, te će biti proslijeđen na stranicu čekanja. Korisnik može pričekati da ga „Admin“ Arduina (korisnik koji se prvi spoji na Arduino se smatra „Admin“-om i može odmah upravljati svojim Arduinom) doda ili odabrati neki drugi Arduino. „Admin“ može blokirati sve koji nisu već blokirani, odblokirati samo one koji su blokirani i prihvatiti ljude u listi čekanja. Odblokirani se pomiču natrag u listu čekanja.

„Admin“ zahtjevi su izvedeni s četiri polja na „Firebase Firestore“- u. Glavno polje „admin“ sadrži polja „uid“ („Admin“-ov identifikator), „blocked“ (lista identifikatora blokiranih korisnika), „approved“ (lista identifikatora prihvaćenih korisnika) i „pending“ (lista korisnika na redu čekanja). Ako „Admin“ odobri korisnika, onda korisnik može upravljati arduinom, ali i dalje može izabrati neki drugi arduino ponovnim postavljanjem.

Ako korisnik zaboravi lozinku može na posebnoj stranici za ponovno postavljanje lozinke upisati svoju email adresu, te će u inbox-u naići na email koji mu je poslao „Firebase Authentication“. U email-u će se nalaziti poveznica na stranicu na kojoj je moguće promijeniti lozinku.

Nakon što je korisnik odobren ili „Admin“ korisnik može upravljati Arduinom kako je opisano u nastavku. U nastavku je također opisana komunikacija između web aplikacije i Arduina.

**Rad aplikacije**

**Node.js**

Node.js je razvojna okolina u kojoj se pokreću JavaScript kodovi prije nego se oni prikazuju u pregledniku. Također omogućava pisanje razvoja web aplikacije za server i klijenta u istom jeziku.

**TypeScript**

TypeScript je jezik za razvoj suvremenih internetskih aplikacija. TypeScript se brzo probio na tržište prilikom svog začeća jer je dao odgovor na mnoge nedostatke dotadašnje popularnog JavaScripta kao što su loše podržan objektno orijentiran razvoj te nedostatak tipova podataka. TypeScript je podržan u svim modernim internetskim preglednicima jer se prevodi u čisti JavaScript.

**Angular**

Angular je strukturni programsko sučelje za pisanje dinamičkih web aplikacija. Dopušta korištenje HTML-a kao jezika za prikaz stranice.

**Bootstrap**

Bootstrap podržava HTML5 i CSS3. On predstavlja kolekciju stvari potrebnih za kreiranje web stranica. Sadrži template za izradu formi, tipki, navigacije na stranice te svih ostalih vizualnih komponenti web stranica.

**Firebase**

Firebase je platforma za mobilne i web aplikacije razvijena od strane Firebase, Inc. koju je kasnije kupio Google. Firebase se koristi kao baza podataka koja pruža mnoštvo biblioteka koje imaju jako dobre opise i optimizirane metode za rad s bazom podataka uz mnoge druge funkcije. U aplikaciji se koriste „Firebase Authentication“ i „Firebase Database“. „Firebase Authentication“ služi za registraciju i prijavu korisnika u sustav, a „Firebase Database“ služi kao baza podataka koja se nalazi na oblaku.

**Rad Arduina**

Arduino je poslužitelj kojemu web aplikacija šalje upite i on joj vraća odgovore. Program izvršavan na Arduinu predstavlja jednostavan REST API. Omogućuje da se HTTP zahtjevima dobave informacije o svim senzorima i kontrolerima spojenim na Arduino. Putanja zahtjeva mora precizirati parametar (temperatura, vlaga ili svijetlo) koji se želi očitati, prostorija u kojoj se nalazi taj parametar i da li se traži senzor ili kontroler. Putanja zahtjeva mora biti valjana tako da je ili jednaka „/conf“ uz metodu GET ili se koriste metode GET ili POST na putanjama tipa: „Prostorija/Parametar/Senzor“ ili „Prostorija/Parametar/ Kontroler“. Svi odgovori na zahtjeve su u JSON obliku. Ako je metoda zahtjeva GET i putanja precizira prostoriju, parametar i senzor ili kontroler, Arduino će odgovoriti sa statusom traženog kontrolera ili senzora u traženoj prostoriji. Ako je metoda zahtjeva POST tada će upravljati kontrolerom traženog parametra u traženoj prostoriji (u ovom slučaju to su LED žaruljice). Odgovor na POST zahtjev biti će status kontrolera. Odgovor na zahtjev na putanju „/conf“ uz GET metodu je skup podataka o svim parametrima u svim prostorijama u JSON formatu. Ako bilo kakav zahtjev nije valjan Arduino-v odgovor će imati status kod '400' (Bad request) i prazno tijelo.

Kod izvršavan na Arduinu oslanja se na razrede i objekte kako bi organizacija koda bila bolja i da se olakšaju buduće nadogradnje. Parametri su predstavljeni apstraktnim razredom „Property“ koja zapravo ima ulogu sučelja, a za svaki parametar postoji zaseban razred („Temperature“, „Humidity“, „Light“) koja nasljeđuje ovaj razred. Kada se stvara instanca nekog od ovih razreda predaje joj se poveznica na senzor koji očitava vrijednost tog parametra (za temperaturu i vlagu to je objekt koji predstavlja senzor vlage i temperature, a za svjetlo to je analogni pin na koji je prispojen senzor). Prostorije su predstavljene razredom „Room“ kojoj predajemo polje (array) razreda „Property“ što predstavlja parametre u toj sobi. Konačno razred „RequestHandler“ služi za prihvat zahtjeva (predstavljenih dvama nizovima znakova: metoda i putanja) i vraćanje odgovora na te zahtjeve u obliku niza znakova. Kada instanciramo razred „RequestHandler“ predaje mu se polje objekata tipa „Room“. Razred „RequestHandler“ sadrži objekte 'Room' koji predstavljaju prostorije, a „Room“ sadrži objekte „Property“ koji predstavljaju parametre u toj sobi. Takvom strukturom se može iz razreda “'RequestHandler“ pristupiti vrijednosti bilo kojeg kontrolera ili senzora u bilo kojoj prostoriji te stvoriti prikladan odgovor na bilo koji valjani zahtjev.

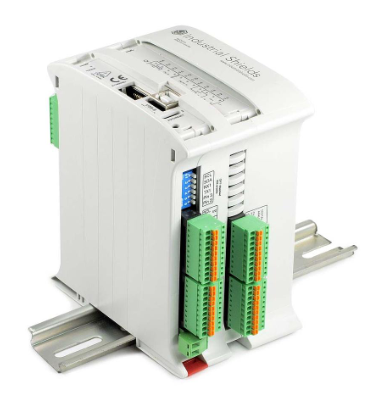
**Hardverska izvedba modela pametne kuće**

Model pametne kuće sastoji se od programibilnog logičkog kontrolera, njegovog napajanja te na njega spojenih senzora i aktuatora.

**Programibilni logički kontroler**

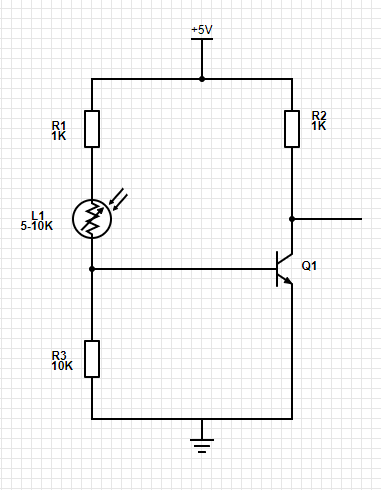
U izvedbi modela pametne kuće korišten je model M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS baziranog na Arduino tehnologiji, ali prilagođen profesionalnoj upotrebi. PLC uređaj ima:

* 13 ulaznih pinova
  + 6 analogno (0 – 10 Vdc) digitalnih (7-24Vdc) programski podesivih
  + 7 digitalno izoliranih (7 – 24 Vdc)
* 8 izlaznih pinova
  + 8 digitalno izoliranih (5 - 24 Vdc) od kojih 3 mogu koristit pulsno širinsku modulaciju



Slika 2: Industrial Shields: M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS

**Senzori modela pametne kuće**U modelu pametne kuće korišten je digitalni senzor vlage i temperature DHT 21 AM2301, a senzor osvjetljenja izveden je analogno pri čemu je korišten NPN tranzistor BC 547, foto-otpornik (5-10 kΩ), dva otpornika (1 kΩ) i jedan otpornik (10 kΩ) spojeni prema shemi na slici 16



Slika 3: Izvedba senzora osvjetljenja

Aktuatori modela pametne kuće simbolizirani su svjetlećim diodama. One zamjenju rasvjetne uređaje i grijače te se njihov intenzitet može mijenjati pulsno širinskom modulacijom.

Senzor temperature i vlage spojen je na ulazni konektor PIN2, Arduino Pin 2, senzor osvjetljena spojen je na anlogni ulazni konektor I0.12, Arduino pin 59, a 3 svjetleće diode spojene su na digitalne izlazne konektore Q0.7, Q0.6 i Q0.5, Arduino pinovi 6,5,4.

# Upute za pokretanje sustava

Za pokretanje sustava potrebno je:

* potpuna hardverska izvedba modela pametne kuće opisana u prijašnjem poglavlju,
* LAN povezana na internet sa mogućnosti spajanja na mrežu Ethernet kablom i odabranom tehnologijom povezivanja računala na mrežu (opisano u sljedećoj točki),
* računalo sa USB portom tipa A za spajanje računala i Arduina, mogućnosti spajanja na LAN mrežu (preko WiFi, Ethernet-a ili neke druge tehnologije – uključuje svu potrebnu opremu za spajanje navedenim tehnologije), te mogućnosti pokretanja Arduino IDE za očitavanje IP adrese Arduina, mogućnosti pokretanja Node.js servera iz terminala za web aplikaciju i mogućnosti pokretanja internetskog preglednika (preporuča se Google Chrome),
* USB kabel sa muškim USB tip A portom na jednoj strani, te muškim USB tip B portom na drugoj za spajanje računala i Arduina,
* Ethernet kabel za spajanje Arduina na LAN,
* strujni utikač koji je spojen na gradsku mrežu (220 – 240 V, 50 Hz) sa osiguračem (u slučaju kratkog spoja, Arduino i njegovo napajanje neće biti zaštićeni bez osigurača) i
* kod za pokretanje Arduina i Node.js servera koji se nalazi u ovom repozitoriju

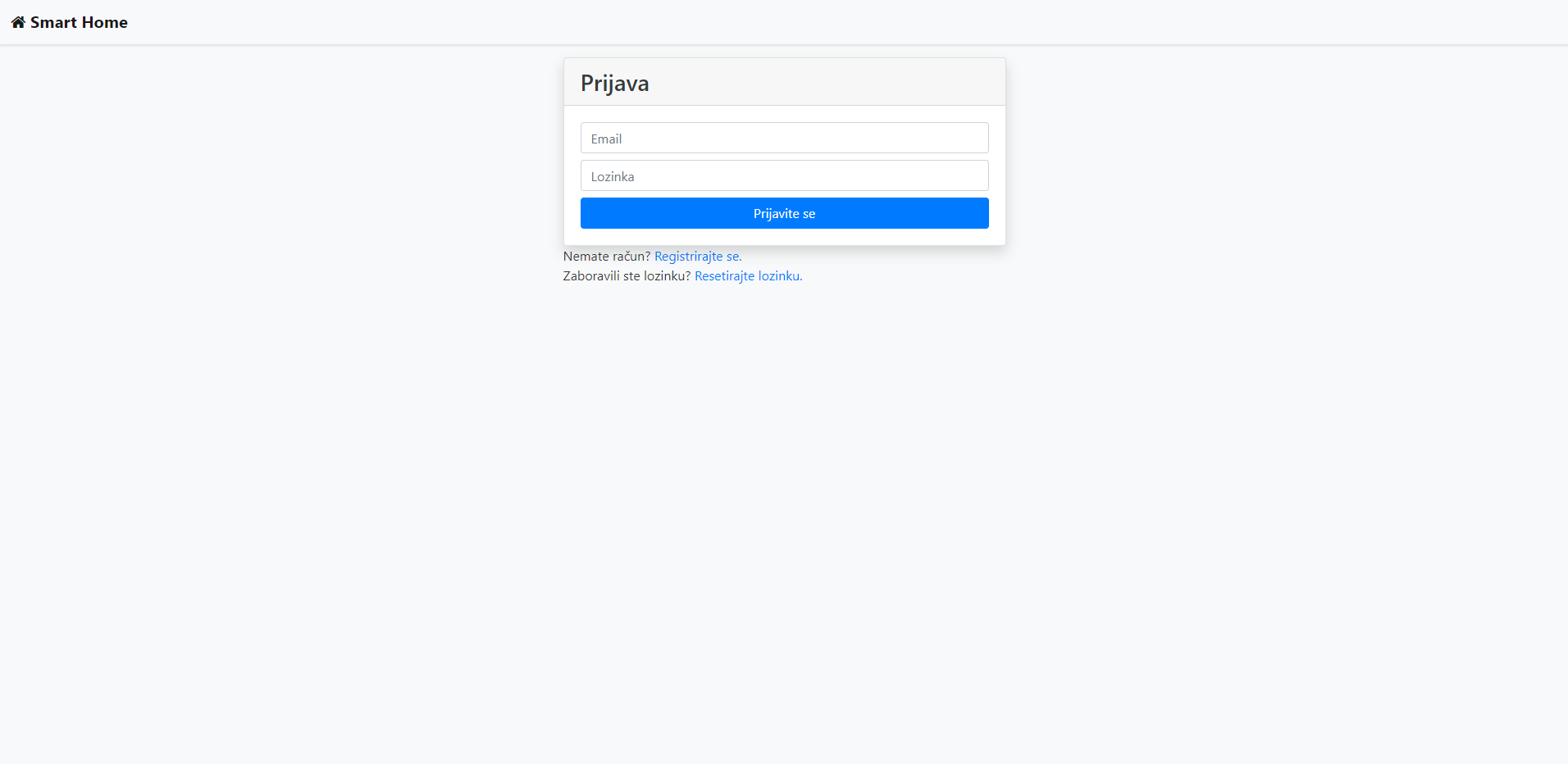
Pokretanje sustava:

1. povezivanje Arduino na LAN Ethernet kablom
2. povezivanje računala i Arduina USB kablom
3. spajanje napajanja Arduina na strujni utikač
4. pokretanje računala i spajanje računala na LAN
5. pokretanje Arduino IDE, učitavanje „Ethernet.ino“ početne točke (nalazi se u mapi „arduino“ ovog repozitorija) na Arduino, te očitavanje IP adrese Arduina iz Serial monitora (ako vaša LAN ima DHCP server moguće je da će trebati nekoliko minuta da Arduinu bude dodijeljena IP adresa)
6. povezivanje aplikacije sa Firebase projektom (aplikacija je već povezana sa Firebase projektom, ali s obzirom da će trebati administrativna prava nad bazom podataka za nastavak pokretanja najbolje je da se aplikacija poveže sa novim Firebase projektom):
   1. kreira se novi Firebase projekt na stranici: <https://firebase.google.com/> (potrebno je povezivanje sa Google računom što se može napraviti ovdje: <https://accounts.google.com/signup/>)
   2. preuzima se konfiguracija novog Firebase projekta uputama na ovoj stranici <https://support.google.com/firebase/answer/7015592> (koristite upute za web aplikaciju)
   3. kopira se konfiguracija novog Firebase projekta i priljepljuje se u datoteke „environment.ts“ i „environment.prod.ts“ (u polja „firebase“ varijable „environment“) koje se nalaze u mapi pod putanjom „\frontend\src\environments“ počevši od putanje repozitorija
7. povezivanje Arduina sa novim Firebase projektom:
   1. na stranici novog Firebase projekta klikne se na „Database“ opciju u izborniku s lijeve strane pod kategorijom „Develop“
   2. dodaje se nova kolekcija klikom na polje „+ Start collection“ imena „arduino“
   3. dodaje se novi dokument u novoj kolekciji klikom na polje „+ Add document“
   4. dodaje se novi ID dokumentu klikom na polje „Auto-ID“, dodaje se polje („field“) tipa mapa („map“) imena „network“ sa poljima „ip“ i „port“ tipa niza znakova („string“), te vrijednostima očitane IP adrese Arduina i porta „80“ i polje „name“ tipa niza znakova („string“) i vrijednosti proizvoljnog imena Arduina (ovo će ime trebati za daljnje povezivanje na Arduino)
8. pokreće se Node.js server na računalu tako da se u terminalu pozicionira u repozitorij, te u mapu „frontend“ i pokrene naredba „npm start“
9. pokreće se internet preglednik na adresi <http://localhost:4200/>
10. registrira se novi korisnik kako je objašnjeno u korisničkim uputama u sljedećem poglavlju, te se spaja na Arduino odabranog imena (odabrano ime pod točkom 7.4)

# Upute za korištenje

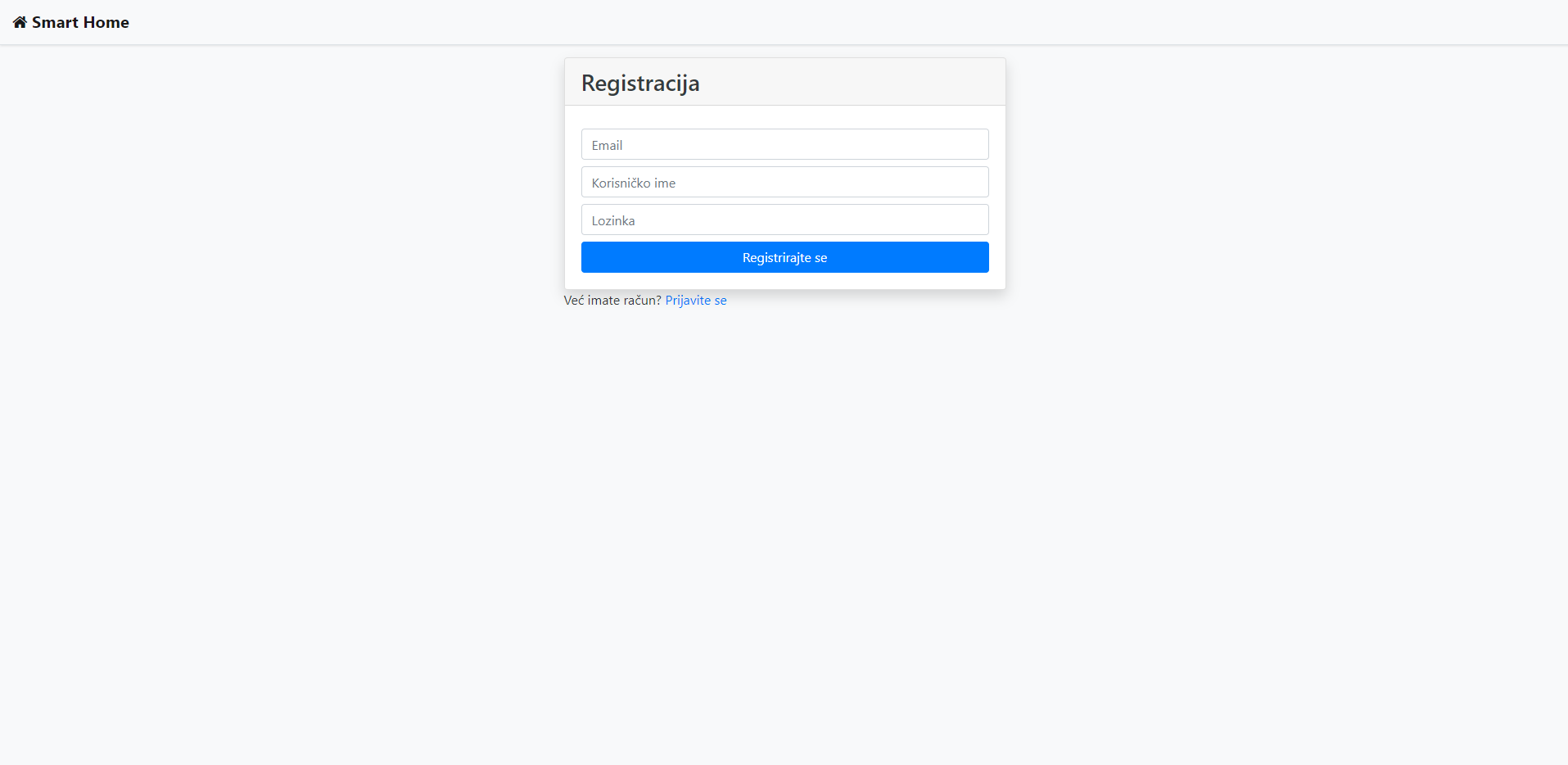
Prilikom pokretanja web aplikacije prvo se pokreće zaslon za prijavu korisnika u sustav. To je prikazano na slici 1.

Slika 4: Login



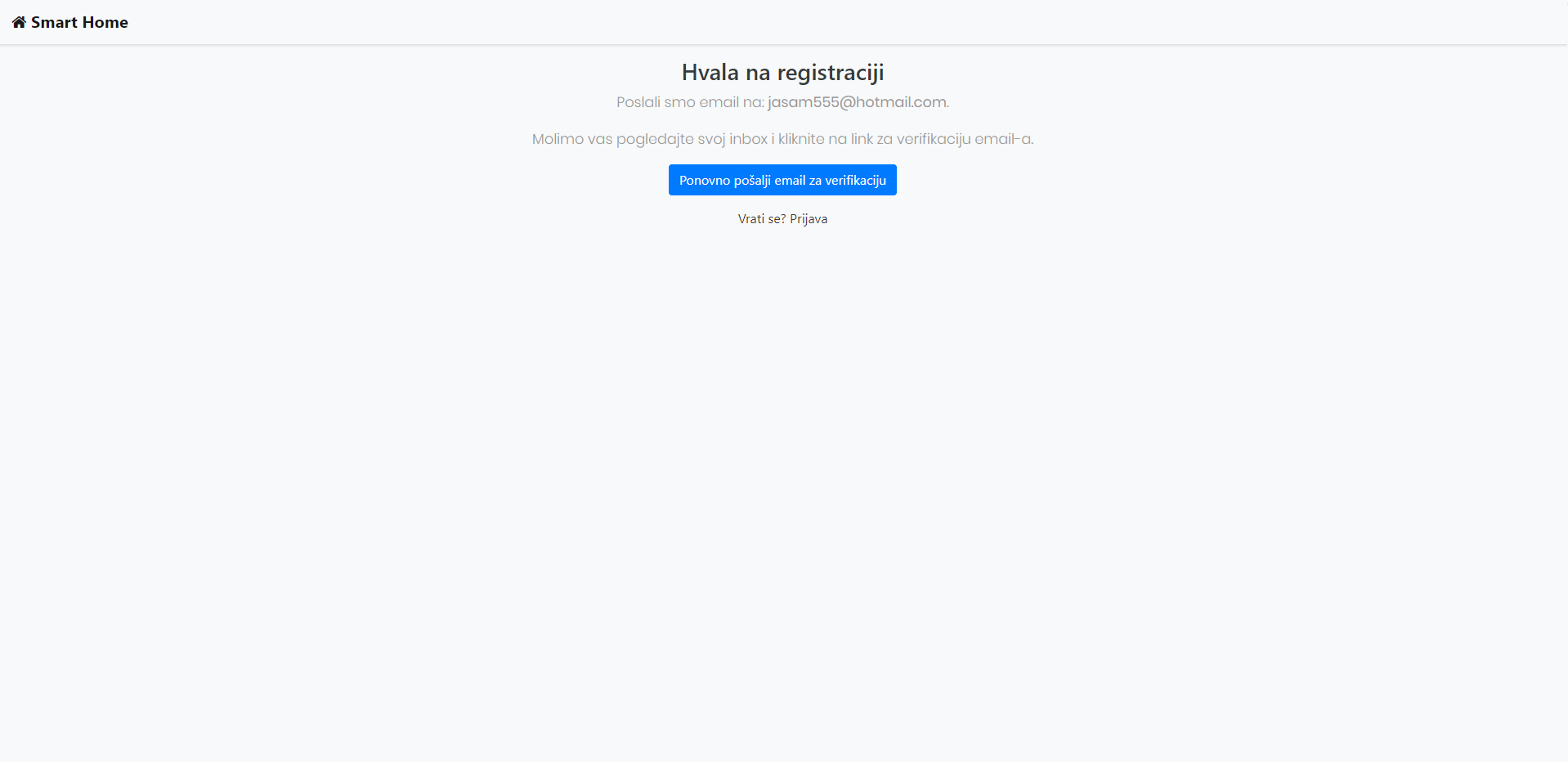
Ukoliko se neki korisnik prvi puta koristi aplikacijom on mora izraditi korisnički račun. Za to je potrebno otići na zaslon za registraciju klikom na tipku „Registrirajte se“ ispod forme za prijavu na slici 1. Na slici 2 vidimo zaslon za registraciju korisnika.

Slika 5: Registracija

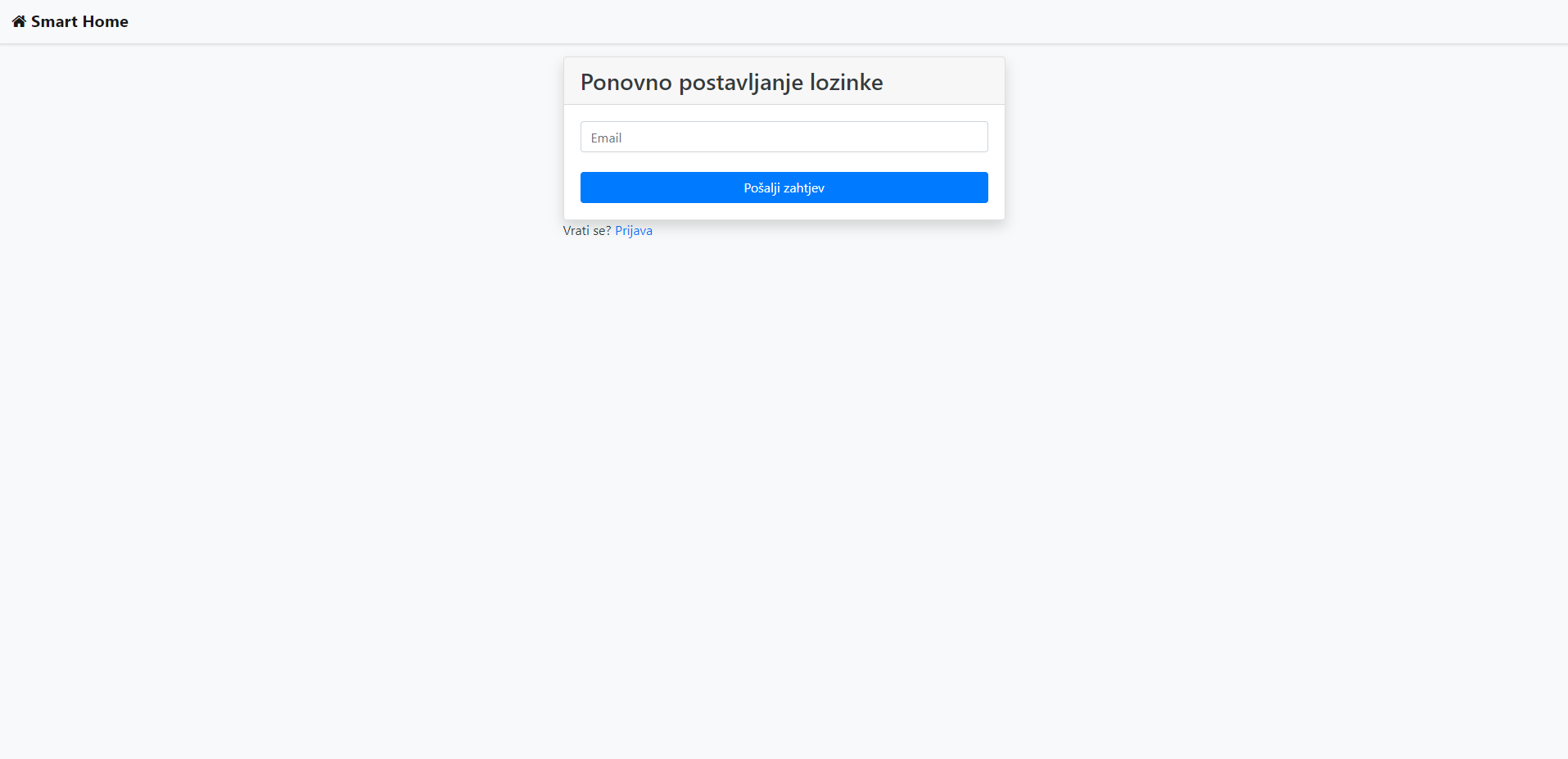


Kada korisnik ispravno popuni formu za registraciju, tj. unese valjanu email adresu i lozinku te korisničko ime aplikacija ga odvodi na zaslon za verifikaciju prikazan na slici 3. S te stranice je moguće ponovno poslati mail za verifikaciju ukoliko je došlo do greške slanja istog prvi put. Također korisnik se može vratiti na stranicu za prijavu.

Slika 6: Verifikacija

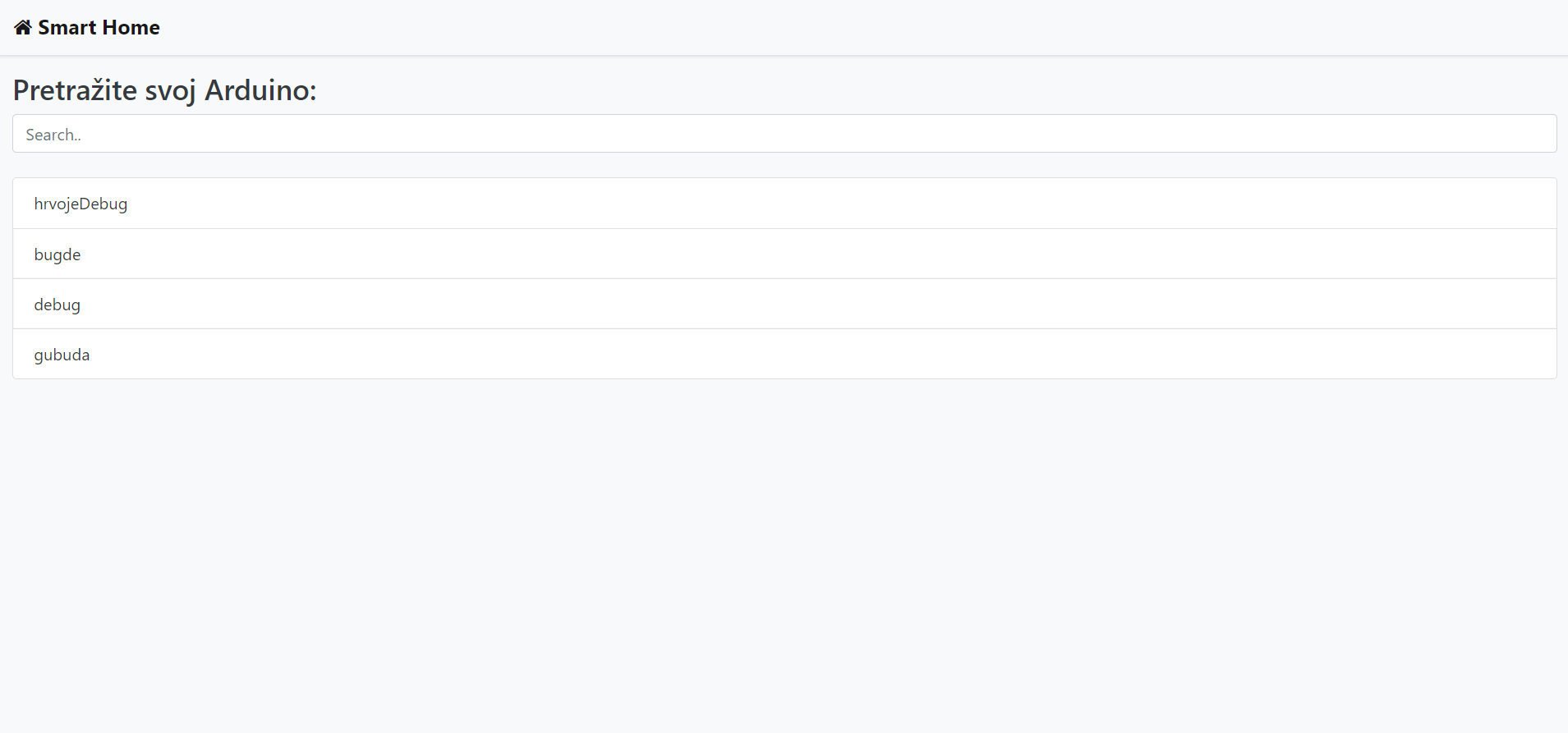


Slika 7: Zaboravljena lozinka



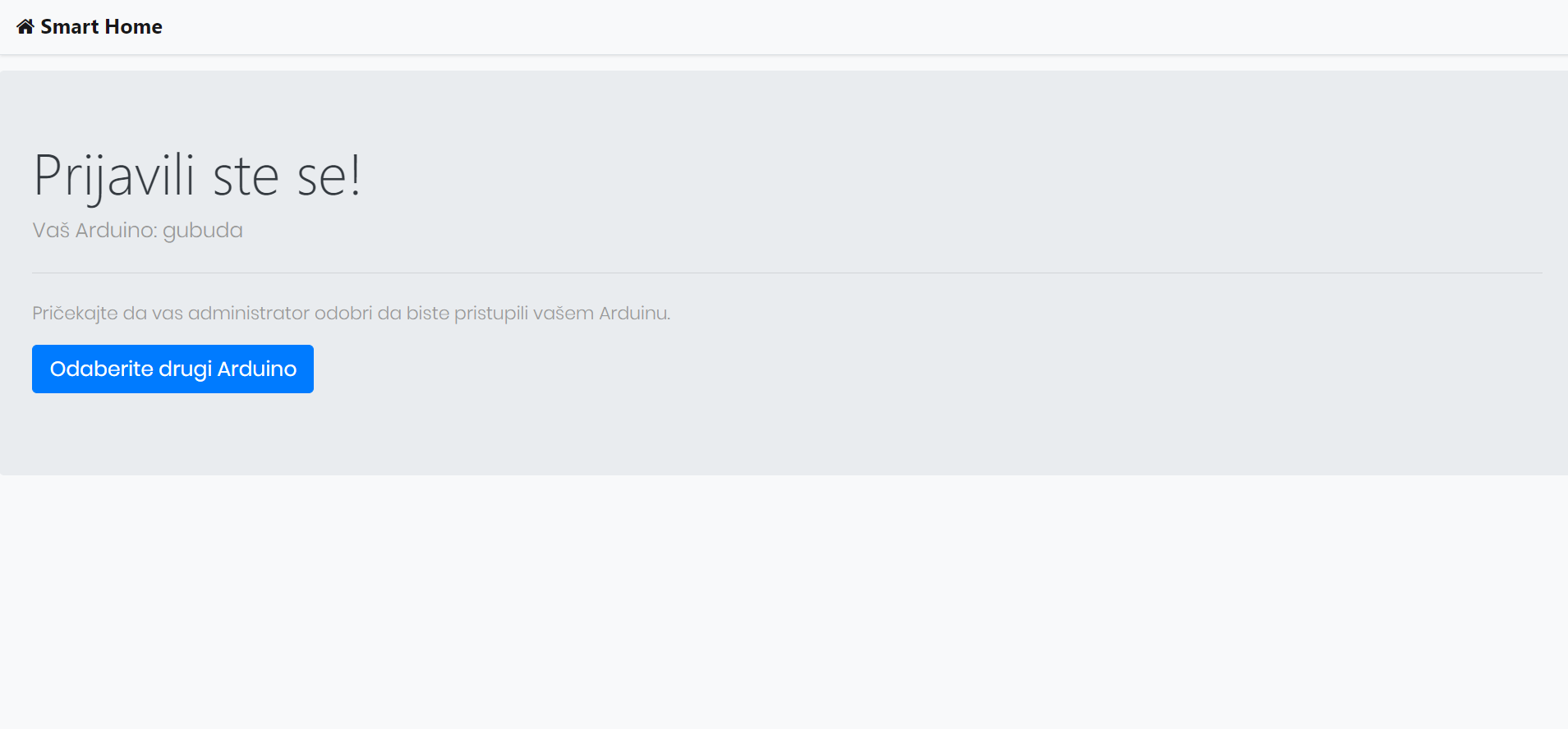
S zaslona za prijavu je moguće pokrenuti stranicu preko koje se ponovno postavlja lozinku ukoliko zaboravimo prije postavljenu lozinku.

Slika 8: Početni zaslon



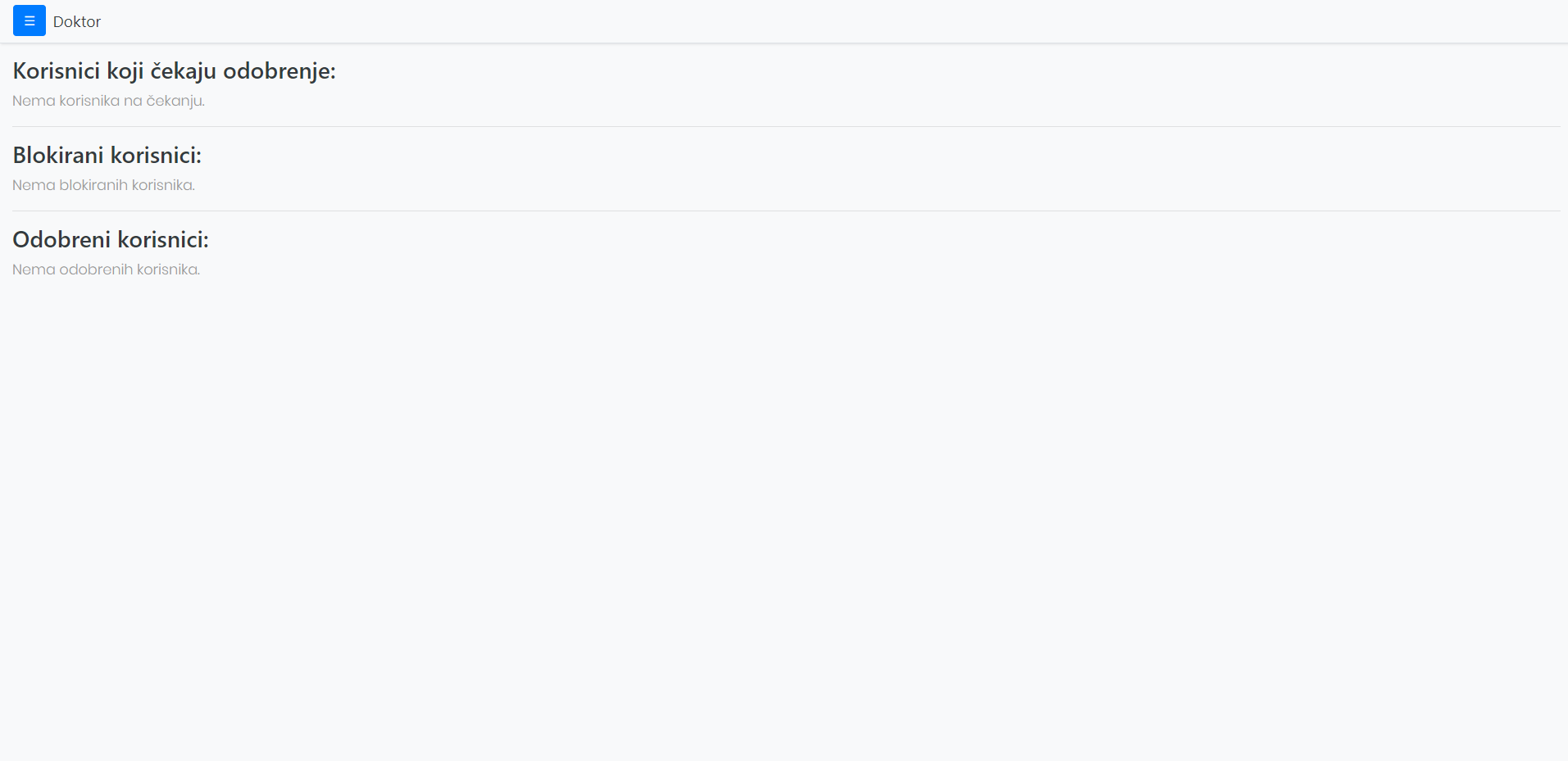
Nakon uspješne prijave pokreće se zaslon na slici 5. Na slici je prikazan popis Arduina na koje se trenutno korisnik može povezati. Klikom na neki od ponuđenih Arduina račun se spaja s određenim Arduinom kojim korisnik želi upravljati. Ako je korisnik prva osoba koja se spaja na taj Arduino on postaje “Admin” toga Arduina i odlučuje koji će se korisnici dalje moći spajati na njega. Ako ipak nije prvi onda će ga aplikacija prebaciti na stranicu prikazanu na slici 6 na kojoj ima mogućnost odabrati neki drugi Arduino ili pričekati dok “Admin” trenutno odabranog Arduina ne prihvati zahtjev za spajanje na Arduino. Jednom kada korisnik počne upravljati nekim Arduinom on ne može promijeniti taj Arduino sve dok ga “Admin” ne blokira na ovom. Tek onda može odabrati neki drugi Arduino.

Slika 9: Odabran Arduino



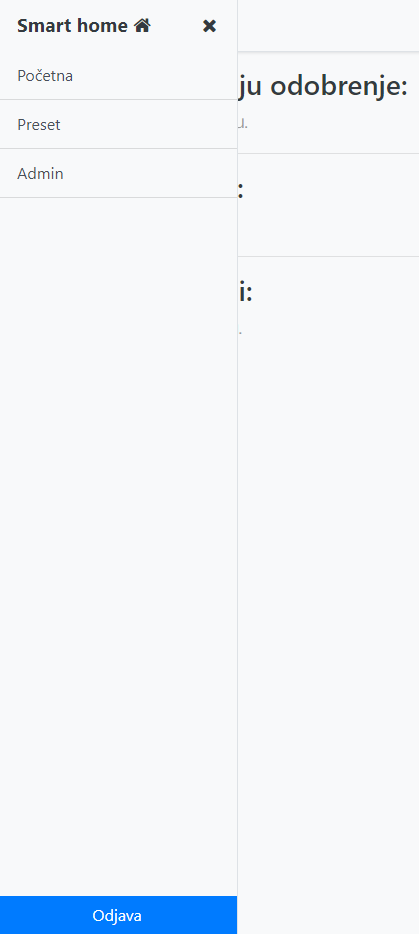
Na slici 7 prikazan je zaslon na kojem “Admin” pojedinog Arduina ima mogućnost prihvatiti određenog korisnika ili odbiti njegov zahtjev za spajanje. Korisnik kojega je “Admin” blokirao stoji u listi „Blokirani korisnici“ sve dok se ne spoji na neki drugi Arduino. Time što taj korisnik stoji u toj listi “Admin” ima mogućnost ponovno dodijeliti tom korisniku mogućnost upravljanja tim Arduinom.

Slika 10: “Admin” zaslon

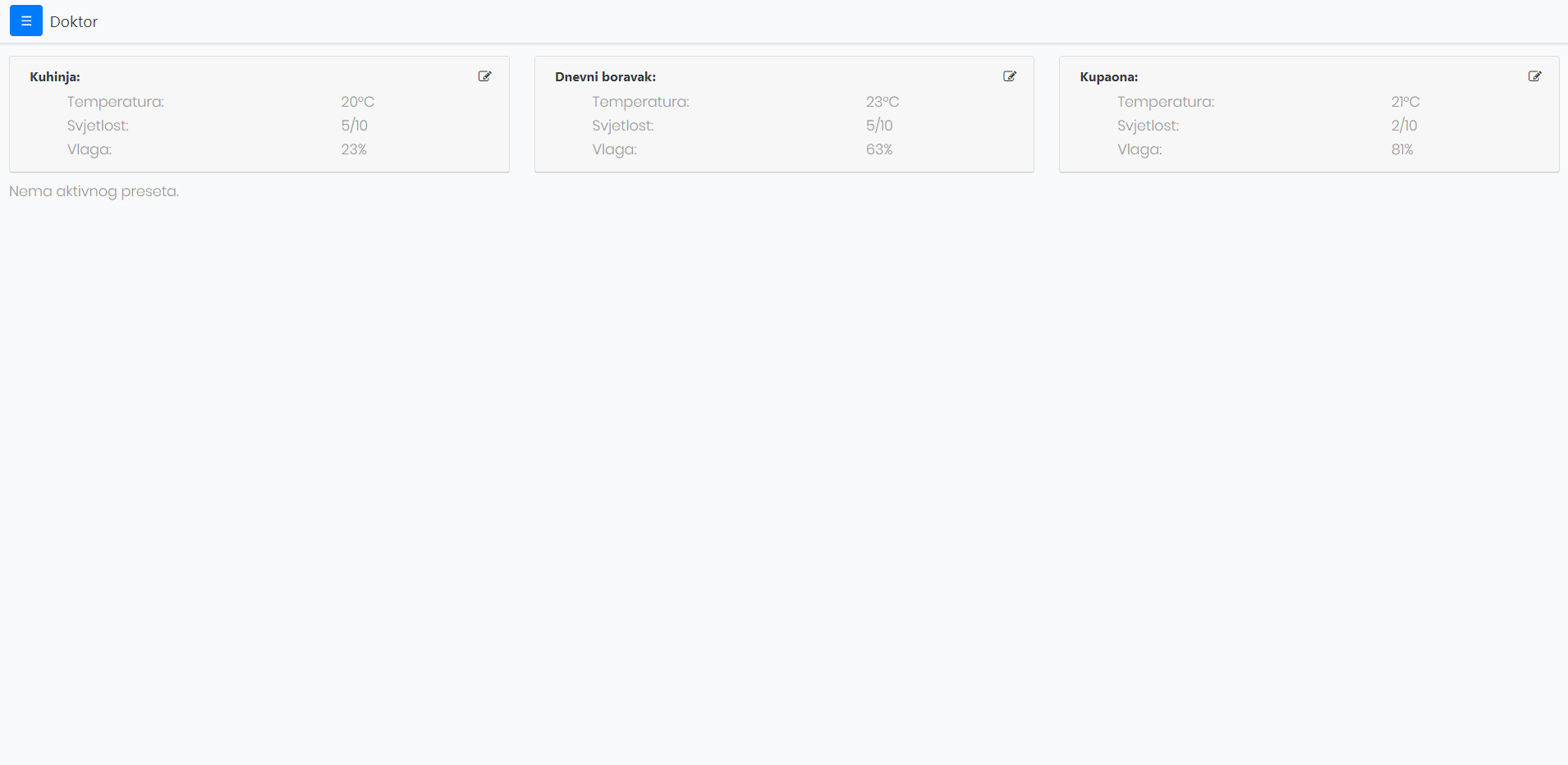


Na slici 8 je prikazana bočna skočna traka pomoću koje se može promijeniti stranica na kojoj se korisnik trenutno nalazi. Postoje tri stranice koje se mogu koristiti. Stranica koja se otvara klikom na “Admin” prikazana je na slici 7. Početna je prikazana na slici 9. dok je stranica “Preset” prikazana na slici 11.

Slika 11: Odabir stranice

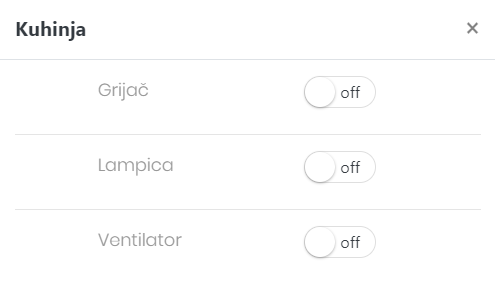


Slika 12: Početna stranica



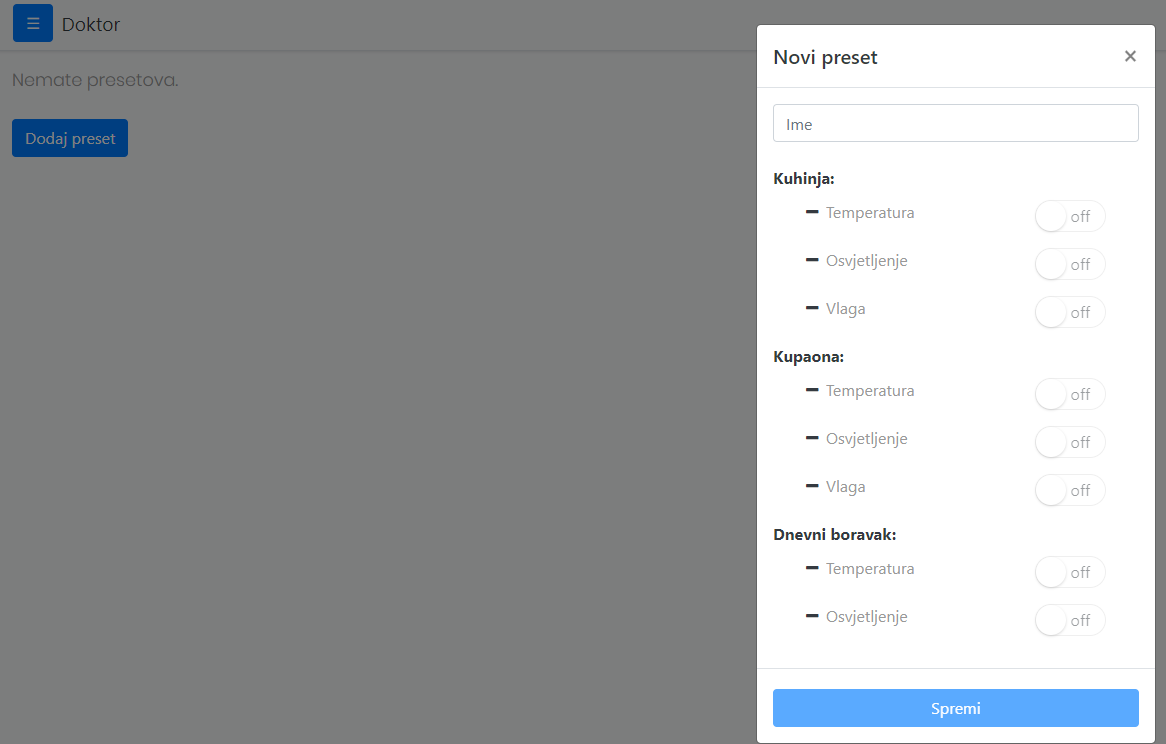
Na početnoj stranici prikazano je stanje na senzorima u pojedinim prostorijama. Ukoliko se želi upravljati bilo kojim kontrolerom u navedenim prostorijama klikne se na ikonu u gornjem desnom kutu pojedine prostorije prikazano na slici 9 te se time otvara skočni prozor prikazan na slici 10. gdje je prikazan primjer upravljanja kontrolerima u kuhinji. Klikom na preklopnik on/off uključujemo ili isključujemo pojedini kontroler.

Slika 13. Upravljanje kontrolerom



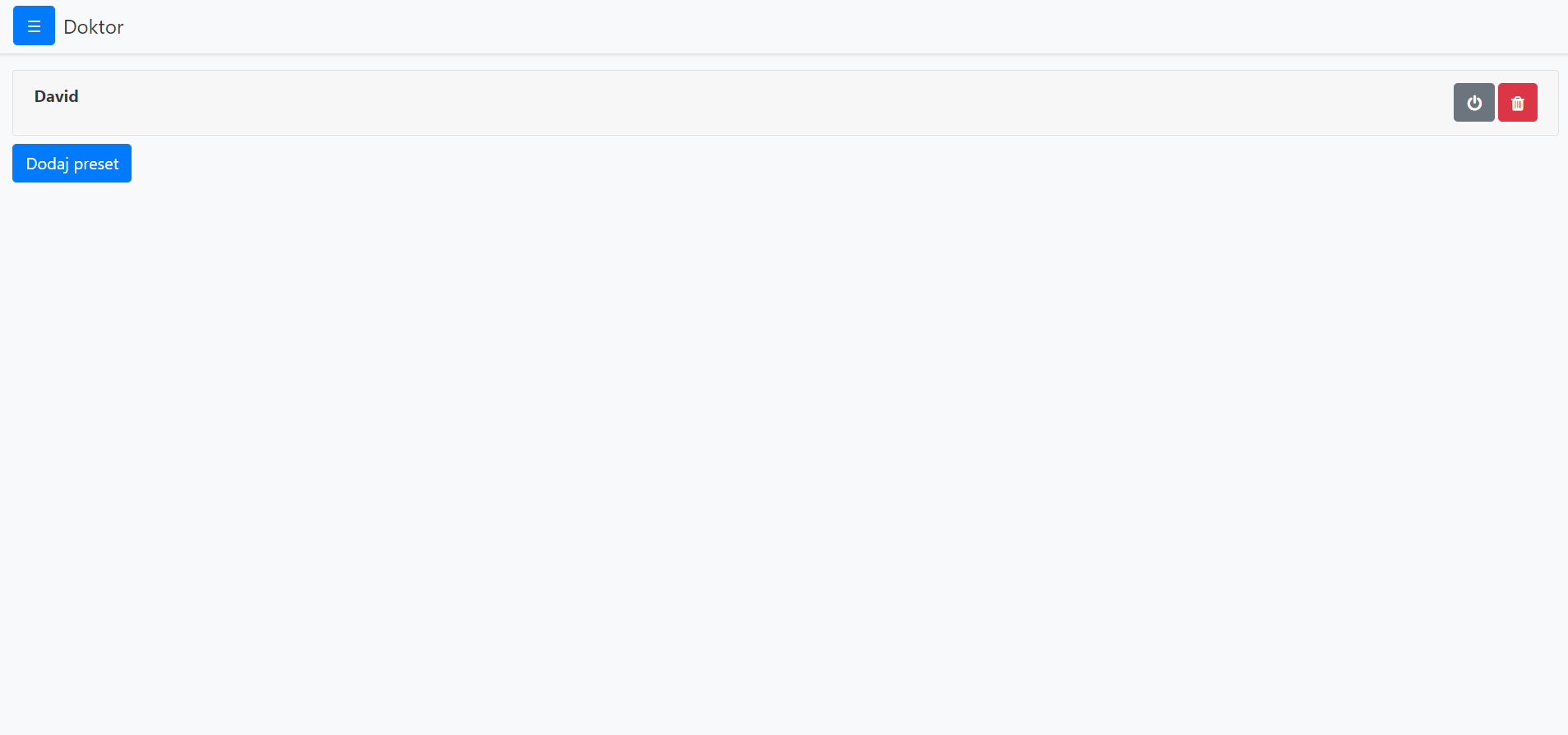
Na slici 11 prikazana je stranica za postavljanje “Preset”-a koja se pokreće klikom na tipku “Preset” prikazanu na slici 8. Tada klikom na tipku dodaj “Preset” otvara se skočni prozor prikazan desno na slici 11 gdje se može postaviti da određeni kontroleri budu uključeni ili isključeni. Svi su po defaultu isključeni te da se uključi pojedini kontroler potrebno je kliknuti na ikonu minus, lijevo te zatim na preklopnik on/off da se uključi.

Slika 14: Postavljanje “Preset”-a



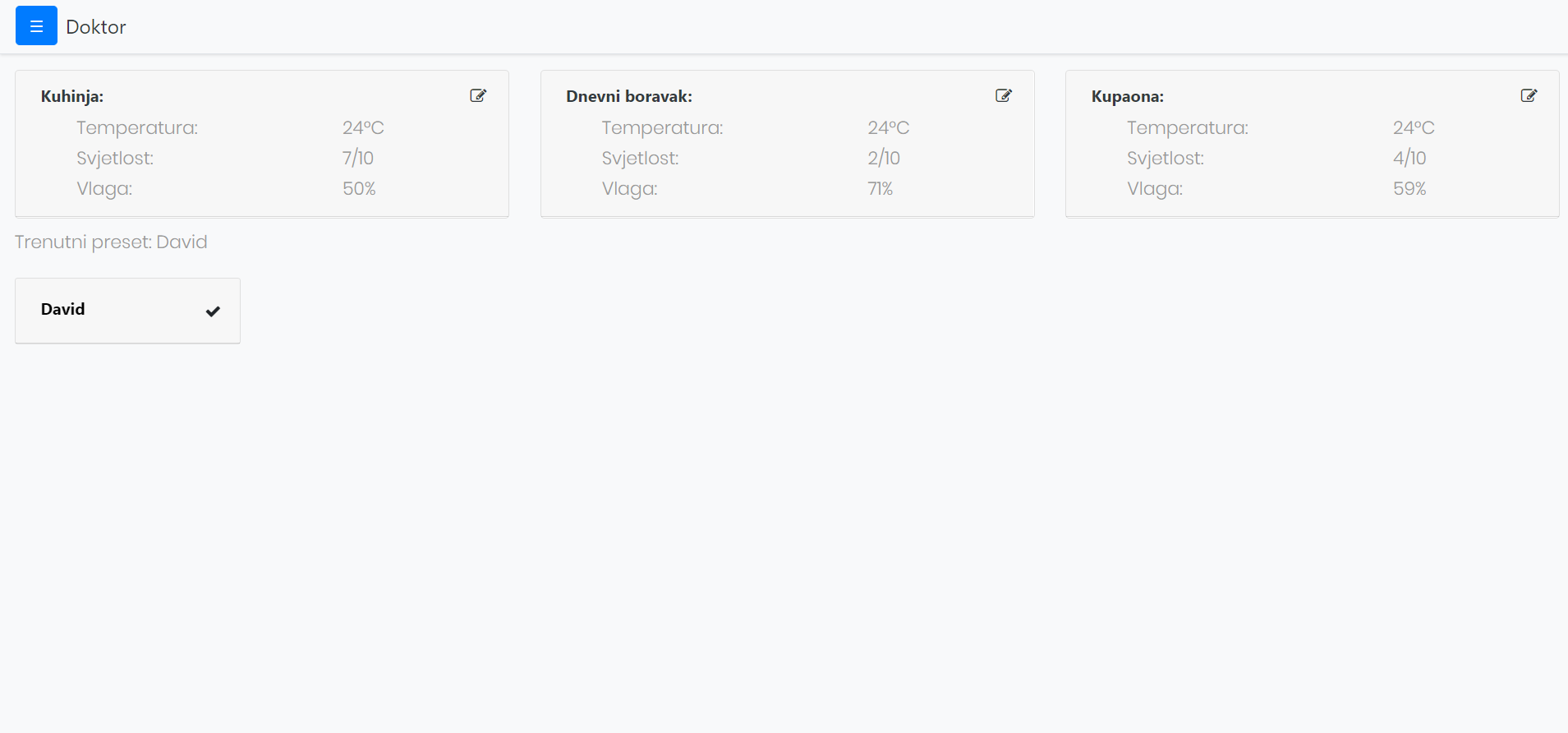
Na slici 12 vidimo dodan “Preset” i ako ga se želi aktivirati potrebno je kliknuti Power tipku prikazanu desno. Pomoću crvene tipke desno može se obrisati određeni “Preset”. Za uređivanje “Preset”-a potrebno je kliknuti na njega te se ponovno otvara skočni prozor prikazan na slici 11.

Slika 15: Dodan “Preset”



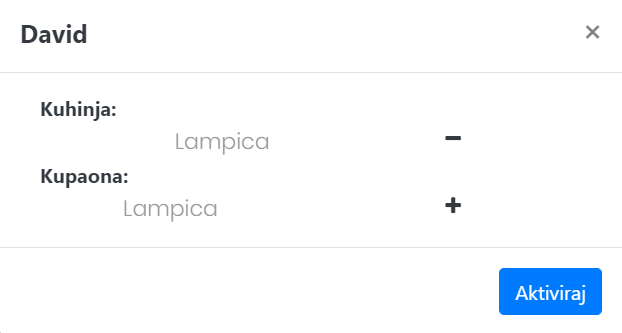
Neki “Preset” se automatski deaktivira ukoliko se promijeni stanje nekog od kontrolera na početnoj stranici kojega on mijenja. Također moguće je dodati više “Preset”a klikom na tipku „Dodaj “Preset”“. Svi “Preset”ovi koji se kreiraju također se pojavljuju na početnoj stranici kao što je prikazano na slici 13.

Slika 16: “Preset” na početnoj stranici



Klikom na “Preset” na početnoj stranici otvara se skočni prozor prikazan na slici 14 kojim se može aktivirati pojedini “Preset”.

Slika 17. Aktivacija “Preset”-a



# Literatura

<https://nodejs.org/en/about/>

<https://www.typescriptlang.org/>

<https://angular.io/>

<https://getbootstrap.com/>

<https://www.survivingwithandroid.com/arduino-rest-api-iot/>

<https://firebase.google.com/>

Ethernet PLC MDuino 21 PLUS User Guide