

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Rok Kanduti

**Pozicioniranje v zaprtih prostorih z
uporabo BLE tehnologije**

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: prof. dr. Marko Bajec

Ljubljana, 2018

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

V diplomskem delu preučite tehnologije, ki se uporabljajo za pozicioniranje v zaprtih prostorih. Izdelajte zasnovo in prototip rešitve, ki bo za ta namen uporabljala statične sprejemnike z zmožnostjo poslušanja naprav v njihovi bližini. Rešitev preverite v realnem okolju in poročajte ugotovitve.

Zahvaljujem se svojemu mentorju prof. dr. Marku Bajcu za pomoč in vodenje pri izdelavi diplomskega dela. Prav tako se zahvaljujem svoji družini za uso podporo v času študija.

Diplomsko delo posvečam svojim
najbližjim.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Pregled področja in obstoječih rešitev	3
2.1	Področja uporabe	3
2.2	WiFi	4
2.3	Bluetooth	8
2.4	RFID	8
3	Zasnova rešitve	11
3.1	Uporabljene tehnologije in orodja	12
3.2	Uporabljena strojna oprema	13
4	Arhitektura in opis rešitve	15
4.1	Baza podatkov	16
4.2	Sprejemniki	20
4.3	Centralno središče	22
4.4	Spletna platforma	23
5	Pregled rezultatov	29
6	Sklepne ugotovitve	35

Seznam uporabljenih kratic

kratica	angleško	slovensko
AP	Access point	Dostopna točka
API	Application programming interface	Aplikacijski programski vmesnik
BLE	Bluetooth Low Energy	Nizkoenergijski Bluetooth
IoT	Internet of Things	Internet stvari
MAC	Media Access Control	Nadzor do dostopa predstavnosti
MQTT	Message Queuing Telemetry Transport	Message Queuing Telemetry Transport
MVC	Model View Controller	Model Pogled Upravljaec
ORM	Object-relational mapping	Objektno-relacijsko mapiranje
REST	Representational state transfer	Predstavitveni prenos stanja
RFID	Radio-frequency identification	Radiofrekvenčna identifikacija
RSS	Received signal strength	Prejeta moč signala
RSSI	Received signal strength indication	Indikator moči prejetega signala
SQL	Structured Query Language	Strukturirani povpraševalni jezik
ToF	Time of flight	Čas letenja

Povzetek

Naslov: Pozicioniranje v zaprtih prostorih z uporabo BLE tehnologije

Avtor: Rok Kanduti

Cilj diplomskega dela je bil izdelati sistem za pozicioniranje naprav v zaprtih prostorih. Večina že obstoječih sistemov deluje tako, da naprava, ki se ji določa lokacijo, posluša statične žolne v prostoru in si nato sama določi svojo pozicijo glede na prejete moči signalov. Sistem, ki sem ga implementiral, deluje na drugačen način. Problem rešuje s pomočjo statičnih sprejemnikov, ki poslušajo naprave v njihovi bližini, katerim določamo pozicijo. V prvem delu naloge so predstavljeni raziskava področja in možni pristopi k reševanju problema, v drugem delu pa sta opisani zasnova in arhitektura implementirane rešitve. Na koncu je predstavljeno delovanje sistema v realnem svetu. Rešitev je primerna za uporabo na področju sledenja sredstev, pri nadzoru gibanja oseb in pri razvoju pametnih hiš.

Ključne besede: pozicioniranje, zaprti prostori, BLE.

Abstract

Title: Indoor positioning using BLE technology

Author: Rok Kanduti

The goal of the thesis was to create an indoor positioning system. Most of the current solutions are based on a device, which listens for nearby beacons with static positions. The device then tries to calculate its own position based on received signal strengths. I took a different approach to solve the problem. It works with multiple static listeners which transmit received signal strengths of nearby devices to central hub. Central hub then aggregates the data and tries to calculate the position of the device. First part of the thesis introduces the field of usage of such systems, while the second one describes the design and architecture of the system. Last part presents the results of its performance in the real world.

Keywords: indoor positioning, BLE.

Poglavje 1

Uvod

V zadnjih letih se tehnologija razvija z nepredstavljivo hitrostjo. S tem so se pojavile nove možnosti, kako ljudem zanesljivo pomagati pri opravljanju procesov, ki so jih včasih delali ročno in so jim vzeli precej časa. Z njeno pomočjo danes že na skoraj vsakem koraku rešujemo težave in opravljamo vsakodnevne opravke. Tehnologije največji potencial dosegajo v gospodarskih panogah, kjer se z njihovo uporabo avtomatizirajo procesi, ki prinašajo večjo produktivnost, izboljšano varnost, cenejšo proizvodnjo, itd. Velik potencial se je pokazal tudi v zdravstvu, kjer je tehnologija že marsikje zamenjala človeka ali mu olajšala delo. V privatna življenja prinaša lažjo dostopnost do informacij, omogoča komunikacijo na daljavo v realnem času in obilo zabave.

Ena izmed takšnih rešitev je razvoj sistema za pozicioniranje naprav v zaprtih prostorih, ki je cilj te diplomske naloge. To je sistem, s katerim lahko v vsakem trenutku preverimo katere naprave se nahajajo v prostoru in njihove približne lokacije. K reševanju tega problema lahko pristopimo na več različnih načinov, saj so na voljo številne različne tehnologije. Ti sistemi imajo uporabno vrednost na večih področjih, kot so nadzor starostnikov v oskrbovalnih domovih, sledenje osnovnih sredstev, nadzor delavcev v proizvodnih obratih, registracija delovnega časa, kontrola pristopa, pametne hiše, itd.

Med študijem sem se preko študentskega dela srečal s problematiko registracije delovnega časa, kontrole pristopa in sledenja osnovnih sredstev. Na tem področju sem na slovenskem in tujem trgu zasledil porast povpraševanja po takšnih sistemih. Pri sledenju sredstev si podjetja želijo zaščititi vrednejša osnovna sredstva in ugotoviti kršitve ter zlorabe pri njihovi uporabi. Avtomatizacija kontrole pristopa in registracije delovnega časa v tej panogi predstavljata „sveti gral“, saj na svetu zaenkrat še ni ponudnika, ki bi zagotavljal zanesljivo rešitev. Po raziskavi trga sem ugotovil, da imajo ti sistemi mnogo širše področje uporabe in možnosti razvoja.

Poglavje 2

Pregled področja in obstoječih rešitev

Pred zasnovno in razvojem sistema za pozicioniranje naprav v zaprtih prostorih, ki je tema diplomske naloge, sem preučil področje, tehnologije in možne pristope za reševanje tovrstne problematike. Na začetku sem opisal uporabno vrednost teh rešitev in področja uporabe. Sledijo možne kombinacije tehnologij in pristopov, ki jih lahko uporabimo za reševanje problema. Večina obstoječih rešitev temelji na uporabi brezžičnih tehnologij, saj zanje ne potrebujemo veliko dodatne infrastrukture.

2.1 Področja uporabe

Sistemi za pozicioniranje v zaprtih prostorih imajo v svetu velik potencial. To potrjuje tudi analiza trga, ki ocenjuje, da bo do leta 2023 vrednost trga za pozicioniranje v zaprtih prostorih presegla 58 milijard ameriških dolarjev [5].

Uporabimo jih lahko v domovih za starejše občane in pri oskrbi starejših ljudi na domu, kjer bi lahko spremljali njihovo gibanje. V primeru odstopanj v njihovih dnevni rutinah bi lahko v zgodnjih fazah začeli odkrivati probleme, ki so privedli do tega. Sistem je zanimiv tudi na področju pame-

tnih hiš, kjer se avtomatizirajo procesi, kot so prižiganje luči, gretje/hlajenje prostorov in iskanje izgubljenih stvari. Uporabni so tudi za navigacijo v prostorih. Primeri tega so že implementirani v nekaterih večjih muzejih, kjer si obiskovalci na telefon namestijo aplikacijo in si na njem preberejo informacije o znamenitostih, mimo katerih se sprehajajo.

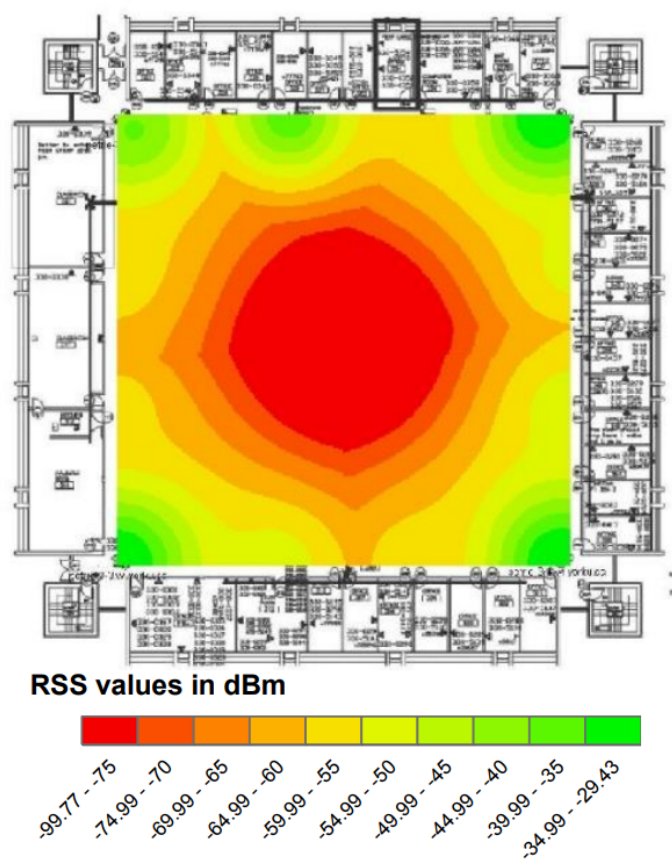
Sistem je primeren tudi za podjetja, ki morajo skrbeti za svoja osnovna sredstva. Nanje lahko namestimo oddajnike in tako v realnem času spremljamo njihovo lokacijo. Z implementacijo alarmnega sistema bi lahko v primeru kršitve pravil takoj ugotovili, da je neko sredstvo zapustilo prostor ali zgradbo, v kateri bi se moralo nahajati. Ti sistemi imajo uporabno vrednost tudi pri registraciji delovnega časa in kontroli pristopa. Podjetja, ki delujejo na tem področju, aktivno iščejo rešitve, kako bi uporabniku čim lažje in s čim manjšo interakcijo olajšala in avtomatizirala proces.

2.2 WiFi

V zadnjem desetletju priljubljenost WiFi-ja hitro narašča. Prav tako se hitro razvija trg mobilnih naprav, ki imajo vedno večje računske in procesorske zmožnosti. Sistemi za pozicioniranje naprav z uporabo te tehnologije delujejo na osnovi WiFi signala, ki ga oddajajo dostopne točke. Najbolj razširjena sta pristop analize prejete moči signala in pristop prstnega odtisa (angl. fingerprinting).

2.2.1 Fingerprinting

Pristop prstnega odtisa za delovanje potrebuje bazo podatkov prejetih moči signalov, na podlagi katerih se izdelava zemljevid, viden na sliki 2.1, ki se uporablja za določanje pozicije naprave. Vsak zapis v bazi podatkov predstavlja prejete moči signalov vseh dostopnih točk v okolici zajema in koordinate pozicije. Naprava lahko nato na podlagi zajema v tej bazi podatkov poišče najbližji približek in določi svojo pozicijo [1]. Ta pristop ne potrebuje veliko dodatne infrastrukture, saj dandanes v stavbah najdemo številne dostopne

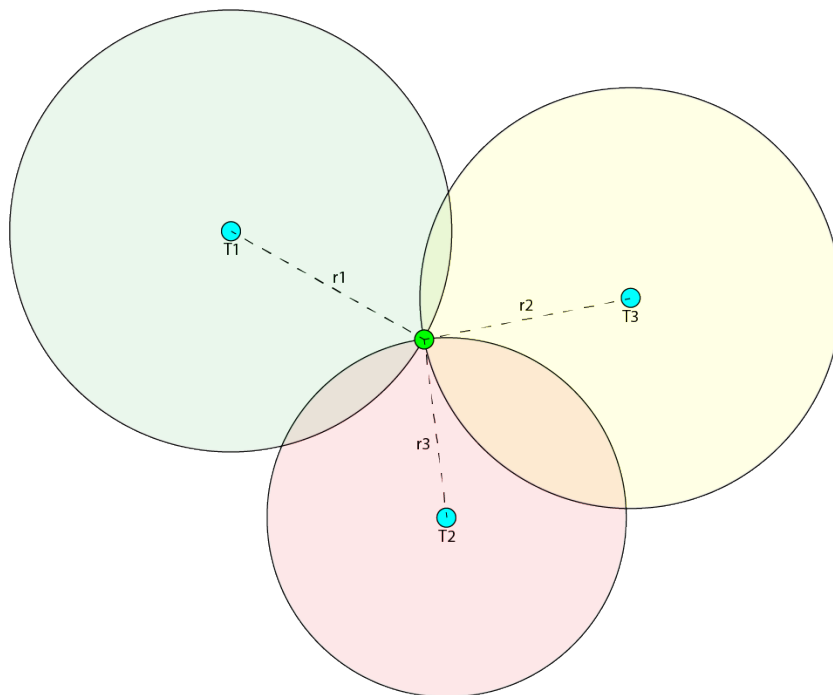


Slika 2.1: Zemljevid prejetih moči signala [1]

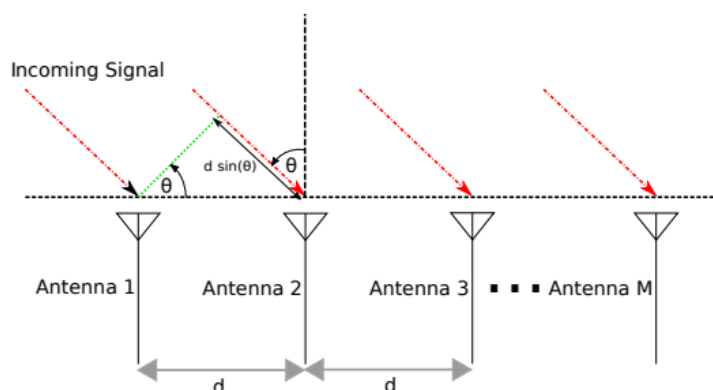
točke, ki jih lahko uporabimo za implementacijo rešitve.

2.2.2 RSS

Podobno kot fingerprinting tudi ta pristop v osnovi deluje na prejeti moči signala. Sprejemnik v prostoru spremlja prejete moči signalov različnih dostopnih točk. Iz prejete moči signala lahko izračunamo razdaljo med oddajnikom in sprejemnikom. Ker imajo dostopne točke statično pozicijo, lahko s pomočjo njihove lokacije in razdalje med sprejemnikom in oddajniki izračunamo približno pozicijo sprejemnika. Ta postopek se imenuje trilateracija. S pomočjo preseka krožnic okoli vsaj treh znanih točk izračunamo izvor signala, kjer polmeri krožnic predstavljajo razdalje med oddajniki in sprejemnikom, slika 2.2.



Slika 2.2: Primer trilateracije



Slika 2.3: Prikaz sprejema signala na različne antene [2]

2.2.3 ToF

Pristop ToF je zasnovan okoli časa prenosa signala od dostopne točke do sprejemnika. Od pristopov, ki temeljijo na prejeti moči signala, se razlikuje po tem, da nanj ne vplivajo odboji signala. Hitrost prenosa radijskega signala po zraku je približek hitrosti svetlobe. Tako lahko iz časa potovanja signala od dostopne točke do sprejemnika izračunamo približno razdaljo med njima. Ko imamo izračunanih več razdalj med sprejemnikom in različnimi dostopnimi točkami, lahko s pomočjo trilateracije izračunamo približno lokacijo sprejemnika, slika 2.2. Za izračun moramo poznati točne pozicije dostopnih točk.

2.2.4 AoA

Pozicijo naprave lahko določimo s kotom prihoda signala od oddajnika do sprejemnika. WiFi dostopne točke so za boljše delovanje opremljene z večimi antenami. Iz razdalje med antenami in časovnim razmikom med časom prihoda signala do vsake antene, lahko ocenimo kot, pod katerim je dostopna točka prejela signal, slika 2.3. Ko poznamo več kotov prihoda od oddajnika do dostopne točke, lahko s pomočjo triangulacije izračunamo približno lokacijo oddajnika [2]. Triangulacija je postopek, kjer s pomočjo krožnic okoli

vsaj treh znanih točk in kotov prihoda izračunamo lokacijo izvora signala.

2.3 Bluetooth

Bluetooth je priljubljenost na tem področju pridobil, ko je izšel protokol BLE. Glavni poudarek pri uporabi bluetootha je približek in ne točna lokacija. Podjetje Apple Inc. je leta 2013 izdalo protokol iBeacon, ki temelji na protokolu BLE. Naprave, ki ga podpirajo, se v okolici oglašajo s svojim unikatnim identifikatorjem. Ker bluetooth ne dosega velikih razdalj, so takšne naprave še posebej primerne za razvoj sistemov za pozicioniranje v zaprtih prostorih.

2.3.1 iBeacon

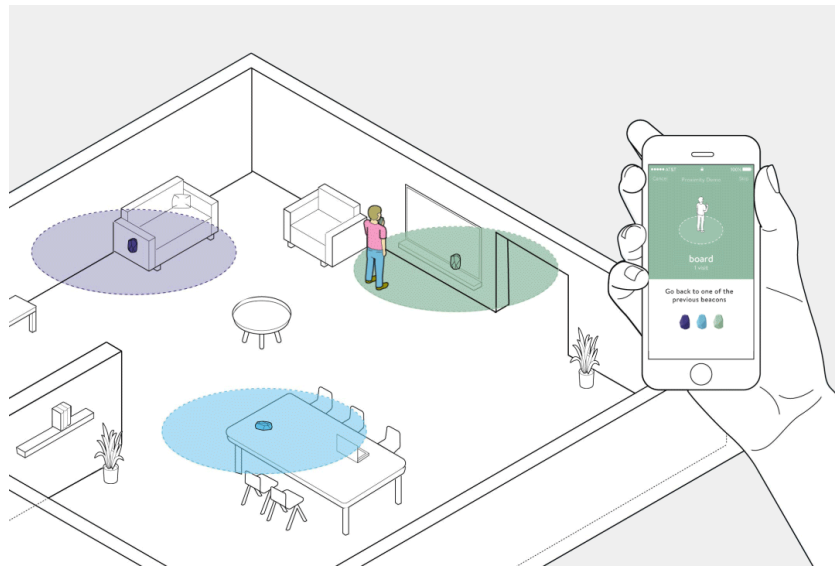
iBeacon naprave imajo v prostoru fiksno pozicijo, ki je shranjena v bazi podatkov, in se okolici oglašajo s svojim unikatnim identifikatorjem. Tako lahko ob prihodu sprejemnika v okolico oddajnika takoj ugotovimo približno lokacijo naprave, slika 2.4.

2.3.2 RSS

Podobno kot pri WiFi-ju lahko tudi pri bluetoothu napravi določimo pozicijo s prejeto močjo signala. Razlika se pojavlja pri dosegu signala, kjer je bluetooth signal šibkejši in dosega manjši domet, zato moramo poskrbeti za gostejše naseljeno senzorsko omrežje.

2.4 RFID

Princip pozicioniranja naprav z uporabo RFID pasivnih značk je podoben kot pri iBeacon napravah. Značke imajo svoj unikatni identifikator, ki ga oddajo samo ob prisotnosti ustreznega radijskega signala. Njihove lokacije



Slika 2.4: Primer prostora opremljenega z iBeacon napravami [10]

so statične in so shranjene v bazi podatkov. Tako lahko določimo približno lokacijo naprave, ko poznamo značke, ki se nahajajo v njeni okolici.

Poglavje 3

Zasnova rešitve

Cilj diplomske naloge je bil razviti univerzalni sistem za pozicioniranje naprav v zaprtih prostorih. Na trgu številni različni ponudniki za nizko ceno ponujajo zanesljive BLE oddajnike v različnih oblikah (kartice, nalepke in obeski za ključe), katerih življenjska doba je nekaj let. Prav tako za sprejemljivo ceno najdemo tudi sprejemnike, katerih naloga je zajem prejetih moči signalov oddajnikov. Odločil sem se za razvoj rešitve z večimi statičnimi sprejemniki in oddajnikom, kateremu se določa pozicija, saj s tem rešitev pokrije največ področij uporabe. Na podlagi zgornjih ugotovitev je najbolj primeren pristop analize prejete moči signala (RSS), rešitev pa temelji na uporabi bluetooth tehnologije. Rešitve, ki so zasnovane okoli WiFi-ja, delujejo v obratni smeri. Imajo statične oddajnike oziroma dostopne točke, sprejemnik pa si na podlagi podatkov o prejetih signalih sam določa svojo pozicijo. V tem primeru so implementacije takšnih sistemov dražje, saj morajo biti sprejemniki zmogljivejši, skrbeti pa moramo tudi za zadostno število dostopnih točk.

3.1 Uporabljene tehnologije in orodja

3.1.1 BLE

BLE je protokol, ki temelji na specifikaciji Bluetooth 4.0. Predstavljen je bil leta 2010 in na IoT področju predstavlja veliko prelomnico. Njegova glavna prednost je, da za veliko manjšo porabo in ceno vzdržuje podoben doseg signala kot klasični bluetooth [4]. Prav zaradi tega se je začel uporabljati v manjših napravah, ki komunicirajo med seboj in pošiljajo svoje podatke v oblačne storitve. Na trgu so se pojavile BLE žolne, ki se oglašajo v okolici. Njihova življenjska doba se je na račun novega protokola znatno povečala.

3.1.2 MQTT

MQTT je objavi/naroči protokol za prenos sporočil, ki sta ga leta 1999 razvila dr. Andy Stanford-Clark iz podjetja IBM in Arlen Nipper iz podjetja Arcom. Protokol je izredno enostaven za uporabo, sporočila pa so okleščena. Zaradi tega je primeren v okoljih, kjer je podatkovna širina omejena, ali v tistih okoljih, kjer prihaja do velike latence. Namen protokola je, da zmanjša porabo pasovne širine, a vseeno ostane zanesljiv. Zaradi tega je uporaben na področju IoT, kjer želimo zaradi velikega števila naprav zmanjšati porabo energije in pasovne širine. Protokol sam ne zagotavlja visoke varnosti komunikacije, so pa za to poskrbele različne implementacije [6].

3.1.3 MySQL

MySQL je najpopularnejši odprtokodni SQL sistem za upravljanje s podatkovnimi bazami. Od leta 2010 ga razvijajo in podpirajo pri podjetju Oracle Corporation. MySQL uporablja relacijsko podatkovno bazo, kar pomeni, da podatke shranjuje v različne tabele in ne v eno veliko podatkovno skladišče. Struktura baze je organizirana v različnih datotekah, kar omogoča optimizacijo pri dostopu do podatkov. Podatkovna baza je odprtokodna in si jo lahko brezplačno namestimo na vsak računalnik. Poleg tega za osnovno delovanje

ne potrebuje zelo zmogljivih sistemov. MySQL ima razvitih veliko vmesnikov za dostop do podatkov in je zato uporabna v skoraj vseh primerih [11].

3.1.4 Sails

Sails je odprtokodni strežniški okvir na zalednih sistemih, ki omogoča razvoj spletnih aplikacij po konceptu MVC [13]. Razvil ga je Mike McNeil in je prvič izšel leta 2012. Primarni cilj razvoja je bil okvir za lažji razvoj spletnih aplikacij na ogrodju NodeJS. Od prvega izida je postal eden najbolj uporabljenih okvirjev za razvoj teh vrst aplikacij na svetu.

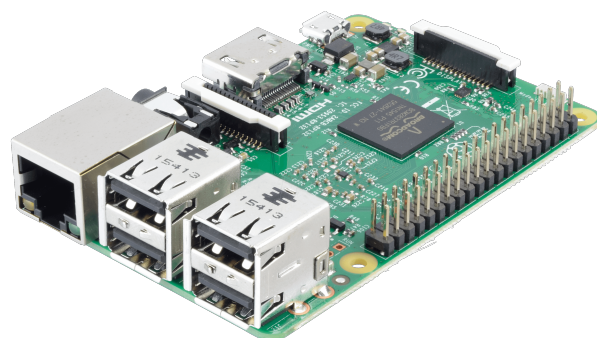
3.2 Uporabljena strojna oprema

3.2.1 Raspberry Pi

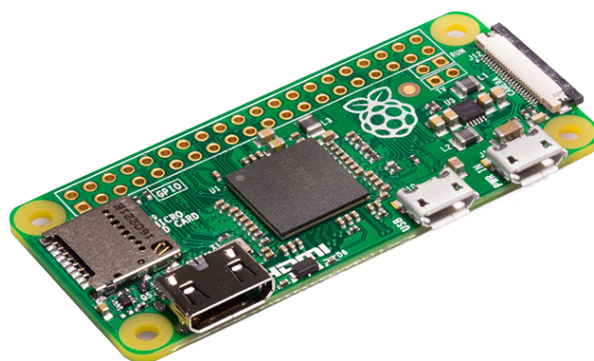
Raspberry Pi je računalnik z nizko ceno v velikosti kreditne kartice, ki ga lahko priključimo na monitor ali televizijo in uporablja klasično tipkovnico in miško [12]. Prvo generacijo računalnika Raspberry Pi je fundacija The Raspberry Pi Foundation leta 2012 izdala za spodbujanje poučevanja osnov računalništva. Računalniki so po izidu postali izredno priljubljeni tudi izven trga, kateremu so bili namenjeni in leta 2015 postali najbolj prodajani računalniki iz Velike Britanije. Leta 2017 so postali tretji najbolj prodajani računalniki za širšo uporabo na svetu.

Raspberry Pi 3 Model B

Raspberry Pi 3 Model B je bil izdan februarja 2016, slika 3.1. Je prvi model tega računalnika, ki ima vgrajen WiFi in Bluetooth čip za povezovanje z zunanjim svetom. Računalnik je opremljen s 64-bitnim štirijedrnim procesorjem Broadcom BCM2837 s frekvenco 1.2 GHz in 1 GB delovnega pomina [8]. S temi specifikacijami je primeren tako za uporabo kot osebni računalnik za nezahtevnega uporabnika, kot v namene izdelave prototipov kompleksnejših računalniških sistemov.



Slika 3.1: Računalnik Raspberry Pi 3 Model B [8]



Slika 3.2: Računalnik Raspberry Pi Zero W [9]

Raspberry Pi Zero W

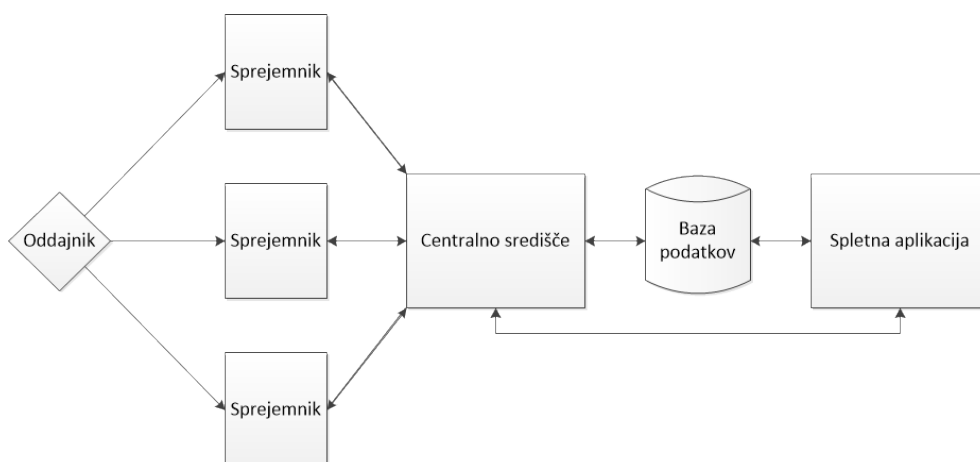
Fundacija je leta 2017 izdala drugo verzijo računalnika Raspberry Pi Zero imenovanega Raspberry Pi Zero W, slika 3.2. Zero W je v primerjavi z osnovnim modelom veliko manjši in cenejši. Tudi temu računalniku so dodali možnost povezave z zunanjim svetom z WiFi in Bluetooth modulom. Opremljen je z enojedrnim procesorjem s frekvenco 1 GHz in 512 MB delovnega spomina [9].

Poglavje 4

Arhitektura in opis rešitve

Cilj diplomske naloge je bil razviti univerzalni sistem za pozicioniranje naprav v zaprtih prostorih. Sistem zaznava vse naprave, ki imajo podprt BLE. Razdeljen je na štiri glavne komponente.

Prva komponenta je baza podatkov, v kateri shranjujemo vse naprave, prostore, sprejemnike in vse prejete zaznave ter lokacije. Druga komponenta so v prostor umeščeni sprejemniki, ki poslušajo naprave v njihovi bližini. Po zaznavi poznane naprave njeno moč signala sporočijo naprej, v tretjo komponento. Ta je centralno središče, ki iz različnih sprejemnikov prejema njihove podatke in jih shranjuje v bazo podatkov. Na vsake N časa, za N časovno obdobje agregira prejete podatke in z njimi izračuna oddaljenost naprave od sprejemnika. Druga in tretja komponenta med seboj komunicirata preko MQTT protokola. Nato za vse naprave po oddaljenosti od sprejemnikov izračuna pozicijo in jo shrani v bazo. Pozicijo posreduje četrti komponenti, spletni platformi, ki podatke prikaže. Spletna platforma je namenjena tudi konfiguraciji sistema, saj v njej določimo naprave, velikosti prostorov, v katerih določamo pozicijo in za te prostore lokacije sprejemnikov. Arhitektura sistema je vidna na sliki 4.1.



Slika 4.1: Arhitektura sistema

4.1 Baza podatkov

Baza podatkov je shranjena na MySQL strežniku. Sestavljena je iz šestih tabel: Devices, Places, Readers, Reading, Reading_Distance in Positions. Shema podatkovne baze se kreira s pomočjo Sails ogrodja, ki skrbi za migracije. Tako se ob prvem zagonu zalednega sistema ustvarijo vse tabele, ki so potrebne za delovanje.

4.1.1 Devices

Tabela Devices (4.1) ima shranjene vse naprave, ki jih sistem zaznava oziroma določa njihovo pozicijo v prostoru. Vsaka v njej shranjena naprava ima svoj unikatni identifikator ID, naziv naprave, MAC naslov, datum in čas vnosa v bazo ter zadnje spremembe. Dodajanje naprav je omogočeno preko spletne platforme.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator naprave
Naziv	VARCHAR(255)	X	Naziv naprave
MAC	VARCHAR(255)	X	MAC naslov naprave
CreatedAt	DATETIME		Datum in čas vnosa v bazo
UpdatedAt	DATETIME		Datum in čas zadnje spremembe

Tabela 4.1: Tabela Devices

4.1.2 Places

Tabela Places (4.2) ima shranjene vse prostore, v katerih sistem zaznava oziroma določa pozicije zaznanih naprav. Vsak prostor, ki je shranjen v njej, ima svoj unikatni identifikator ID, naziv prostora, dolžino po X in Y osi v metrih, datum in čas vnosa v bazo ter zadnje spremembe. Dodajanje prostorov je mogoče preko spletne platforme.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator prostora
Name	VARCHAR(255)	X	Naziv prostora
PosX	DOUBLE	X	Dolžina prostora po X osi
PosY	DOUBLE	X	Dolžina prostora po Y osi
CreatedAt	DATETIME		Datum in čas vnosa v bazo
UpdatedAt	DATETIME		Datum in čas zadnje spremembe

Tabela 4.2: Tabela Places

4.1.3 Readers

Tabela Readers (4.3) ima shranjene vse sprejemnike, ki so umeščeni v znane prostore. Vsak sprejemnik, ki je shranjen v njej, ima svoj unikatni identifikator ID, naziv sprejemnika, prostor, v katerem se nahaja, pozicijo v tem prostoru po X in Y osi, datum in čas vnosa v bazo ter zadnje spremembe. Definiranje sprejemnikov je mogoče preko spletne platforme.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator sprejemnika
Name	VARCHAR(255)	X	Naziv sprejemnika
Place	INT(11)	X	Prostor v katerem je umeščen
PosX	DOUBLE	X	Pozicija po X osi
PosY	DOUBLE	X	Pozicija po Y osi
CreatedAt	DATETIME		Datum in čas vnosa v bazo
UpdatedAt	DATETIME		Datum in čas zadnje spremembe

Tabela 4.3: Tabela Readers

4.1.4 Reading

Tabela Reading (4.4) ima shranjene vse podatke naprav, ki so jih zaznali sprejemniki. Vsaka zaznava, ki je shranjena v tabeli, ima svoj unikatni identifikator ID, napravo, prostor, v katerem je bila zaznana, kateri sprejemnik jo je zaznal, moč prejetega signala ter datum in čas zaznave. V tabelo jih shranjuje centralno središče, ki zaznave prejema od sprejemnikov.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator zaznave
Device	INT(11)	X	Zaznana naprava
Reader	INT(11)	X	Sprejemnik
Place	INT(11)	X	Prostor zaznave
RSSI	DOUBLE	X	Prejeta moč signala
Timestamp	DATETIME	X	Datum in čas zaznave

Tabela 4.4: Tabela Reading

4.1.5 Reading_Distance

Tabela Reading_Distance (4.5) ima shranjene preračunane razdalje naprav od sprejemnikov, agregirane po času. Vsak zapis, ki je shranjen v tabeli, ima svoj unikatni identifikator ID, napravo, prostor, sprejemnik, razdaljo naprave

od sprejemnika ter datum in čas izračuna. V tabelo jih shranjuje centralno središče na vsake N časa, ko iz prejetih moči signala izračuna razdaljo.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator zapisa
Device	INT(11)	X	Naprava
Reader	INT(11)	X	Sprejemnik
Place	INT(11)	X	Prostor
Distance	DOUBLE	X	Prejeta moč signala
Timestamp	DATETIME	X	Datum in čas zapisa

Tabela 4.5: Tabela Reading_Distance

4.1.6 Positions

Tabela Positions (4.6) ima shranjene pozicije naprav v prostoru. Vsaka pozicija, ki je shranjena v tabeli, ima svoj unikatni identifikator ID, napravo, prostor, pozicijo po X in Y osi, napako pri poziciji, datum in čas izračuna, zapisa v bazo ter zadnje spremembe. V tabelo jih shranjuje centralno središče na vsake N časa, ko iz prejetih razdalj izračuna pozicijo naprave v prostoru.

Stolpec	Tip	Not null	Komentar
ID	INT(11)	X	Unikatni identifikator
Device	INT(11)	X	Naprava
Place	INT(11)	X	Prostor
PosX	DOUBLE	X	Pozicija po X osi
PosY	DOUBLE	X	Pozicija po Y osi
STD	DOUBLE	X	Možno odstopanje
Timestamp	DATETIME	X	Datum in čas pozicije
CreatedAt	DATETIME		Datum in čas zapisa v bazo
UpdatedAt	DATETIME		Datum in čas zadnje spremembe

Tabela 4.6: Tabela Positions

4.2 Sprejemniki

Sprejemniki so implementirani na malih računalnikih Raspberry Pi. Za namene diplomske naloge sem uporabil en računalnik Raspberry Pi 3 Model B in dva Raspberry Pi Zero W, sliki 3.1 in 3.2. Računalnik s centralnim središčem komunicira preko MQTT protokola. Ob zagonu se zažene Python program, ki od centralnega središča najprej zahteva rekonfiguracijo vseh znanih naprav. Časovnik za rekonfiguracijo je definiran v konfiguracijski datoteki. Hkrati v ozadju zažene proces, ki pokliče sistemsko orodje „hcitool“:

```
self._scanProcess = subprocess.Popen(
    [ 'hcitool', 'lescan', '--duplicates' ],
    cwd='/usr/bin',
    stderr=subprocess.STDOUT,
    stdout=subprocess.PIPE)
```

Ta proces posluša bližnje BLE naprave, na standardni izhod „STDOUT“ pa izpiše njihove MAC naslove in RSSI. Ko program prejme podatke o napravi, preveri ali je ta registrirana. V primeru, da je naprava znana, njeno prejetjo moč signala sporoči centralnemu središču.

4.2.1 Konfiguracija sprejemnikov

Vsak sprejemnik ima svojo konfiguracijsko datoteko „config.ini“, ki jo program prebere ob zagonu. V njej so zapisani podatki, ki so potrebni za pravilno delovanje in povezavo z MQTT posrednikom.

Primer konfiguracijske datoteke:

```
[General Settings]
ReaderId: 1
Place: 1
MqttAddress: 10.0.0.2
```

Recache: 600

[Logging]

UseLog: False

File: reader.log

MaxSize: 1024

FileCount=5

STDOUT: True

Najpomembnejše so nastavitve v odseku „General Settings“. Tam je nastavljena identifikacijska številka sprejemnika, ki jo dobimo, ko dodamo nov sprejemnik, naslov MQTT posrednika in časovnik za rekonfiguracijo s centralnim središčem.

4.2.2 Rekonfiguracija sprejemnikov

Sprejemnik ob zagonu zažene časovnik, ki ob zagonu in na vsake N časa od centralnega središča zahteva rekonfiguracijo. To stori tako, da pošlje prazno sporočilo na kanal

```
ble/{id}/devices/reconfigure
```

kjer je {id} njegova unikatna identifikacijska številka. Centralno središče nato na kanal

```
ble/{id}/devices
```

pošlje sporočilo z registriranimi napravami.

Primer rekonfiguracijskega sporočila:

```
{
  "C8:0F:10:59:9E:63":
    {
      "name": "Naprava1",
      "id": "1"
    }
}
```

```
}  
}
```

4.2.3 Zaznava naprave

Sprejemnik s pomočjo sistemskega orodja „hcitool“ posluša BLE naprave, ki so v okolici. Če zazna napravo, njen MAC naslov primerja s poznanimi napravami. V primeru, da je naprava poznana, njeno prejeta moč signala pošlje centralnemu središču preko kanala

```
ble/{id}/reading
```

kjer je {id} unikatni identifikator sprejemnika.

Primer sporočila:

```
{  
  "Rssi": "-63",  
  "ReaderId": "2",  
  "Place": "1",  
  "DeviceId": "1",  
  "TimeStamp": "2018-06-10 19:37:23.354885"  
}
```

4.3 Centralno središče

Centralno središče je implementirano v programskem jeziku Python. Python ima na razpolago velik nabor zunanjih knjižnic, ki nam olajšajo delo in je podprt na večini večjih platform [3]. Ob inicializaciji se poveže na MQTT kanal, na katerem sprejemniki zahtevajo rekonfiguracije. Ob zahtevku za rekonfiguracijo v bazi poišče vse registrirane naprave in jih pošlje na sprejemnik, ki je zahteval rekonfiguracijo.

Na drugem kanalu posluša vse zaznave, ki jih pošiljajo sprejemniki. Ko prejme novo zaznavo, jo s časovno oznako shrani v bazo.

Časovnik na vsake N časa zažene proces, ki za vsak sprejemnik in napravo agregira prejete podatke za časovno obdobje N, izračuna povprečno vrednost RSSI, zavrže podatke, ki od nje najbolj odstopajo in izračuna oddaljenost naprave od sprejemnika.

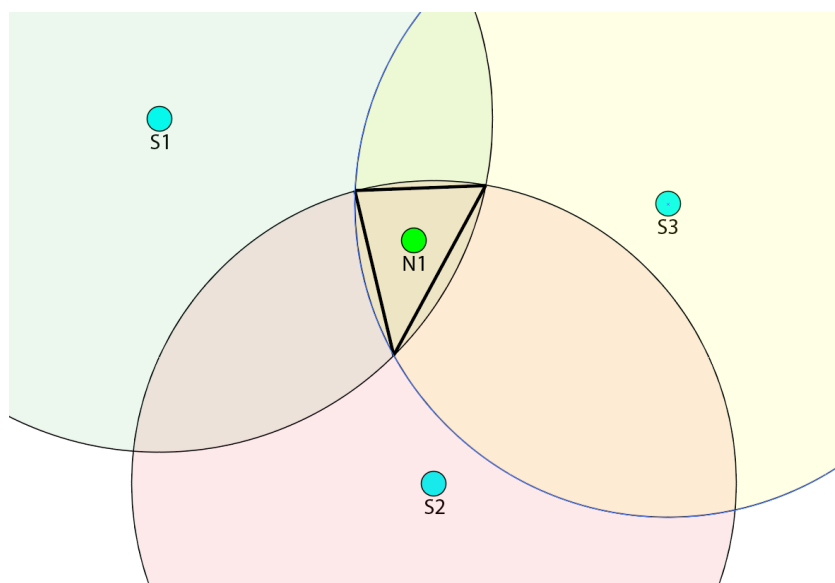
```
for reader in readers:
    devices = getDevicesFromReadings(reader)
    for device in devices:
        readings = getReadings(reader, device)
        rssi = calculateRSSI(readings)
        distance = calculateDistance(rssi)
        saveReadingDistance(reader, device, rssi)
```

Izračunano razdaljo in ostale podatke zapiše v bazo.

Po tem, ko za vse naprave izračuna oddaljenost, izračuna še pozicijo v prostoru. Za vsak prostor najprej preveri, da obstajajo podatki vsaj treh sprejemnikov, saj v nasprotnem primeru ne more izračunati pozicije. Pozicijo računa s pomočjo krožnic, kjer razdalja med napravo in sprejemnikom predstavlja polmer krožnice [14], središče preseka vseh krožnic pa določa položaj naprave, slika 4.2. Pozicijo naprave shrani v bazo podatkov. Prav tako jo pošlje tudi vsem odjemalcem, ki imajo v tistem trenutku preko spletne platforme odprt pregled tega prostora.

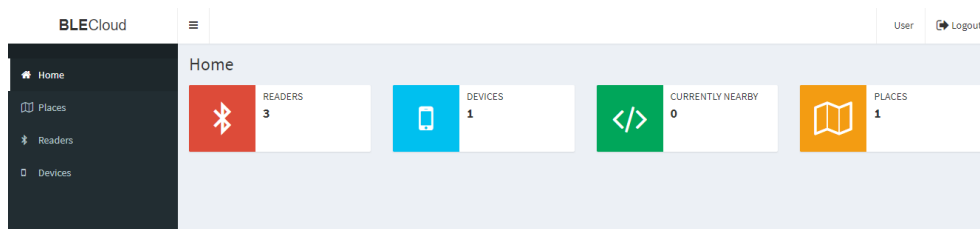
4.4 Spletna platforma

Spletna platforma je razvita s pomočjo odprtokodnega strežniškega okvirja Sails. Razvita je po MVC konceptu, kar pomeni, da ima model, pogled in upravljalnik. Model skrbi za upravljanje s podatki, pogled je uporabniški vmesnik, upravljalnik pa skrbi za uporabnikove zahteve [7]. Implementirana je v angleškem jeziku in ima 5 pogledov. Prvi pogled je naslovna stran, na kateri so prikazani številčni podatki o dodanih napravah, prostorih in sprejemnikih, slika 4.3. Na drugem pogledu je tabela, v kateri so prikazani shranjeni prostori in ima možnost dodajanja novih. Vsak prostor



Slika 4.2: Presečišče treh krožnic

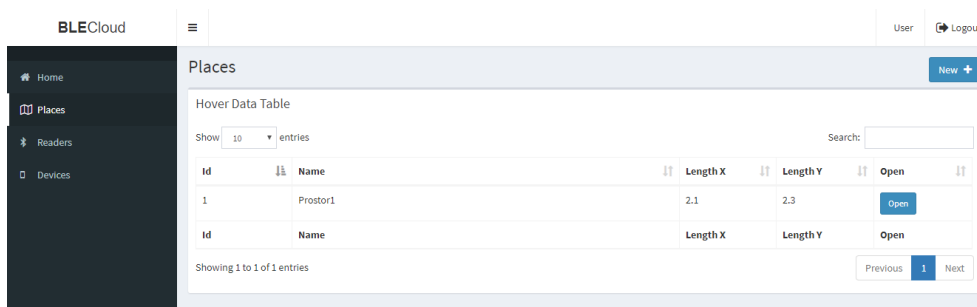
ima tudi povezavo do tretjega pogleda, ki prikazuje njegovo velikost, tloris in pregled naprav, ki so v njem. Četrty pogled prikazuje tabelo vseh registriranih sprejemnikov in njihovih podatkov. Na tem pogledu lahko dodamo nov sprejemnik. Zadnji pogled prikazuje podatke o napravah, ki so registrirane v sistem in omogoča dodajanje novih. Implementiran je tudi pogled za prijavo v aplikacijo z uporabiškim imenom in geslom.



Slika 4.3: Domača stran spletne platforme

4.4.1 Prostori

Na sliki 4.4 je prikazan pogled spletne platforme, kjer je vidna tabela s podatki o prostorih. V njej je za vsak prostor prikazan unikatni identifikator, njegov naziv, dolžina po X in Y osi ter povezava do pregleda prostora.



Id	Name	Length X	Length Y	Open
1	Prostor1	2.1	2.3	Open

Slika 4.4: Pregled vseh prostorov

S pritiskom na gumb „New“ se odpre modalno okno, kjer lahko dodamo nov prostor, prikazano na sliki 4.5.

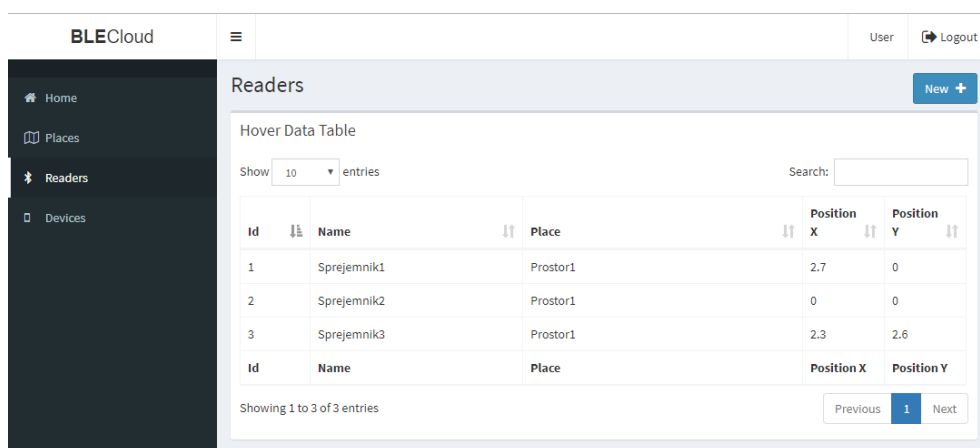


Slika 4.5: Dodajanje prostora

4.4.2 Sprejemniki

Na sliki 4.6 je prikazan pogled spletne platforme, kjer je vidna tabela s podatki o sprejemnikih. V njej je za vsak sprejemnik prikazan unikatni iden-

tifikator, njegov naziv, prostor, v katerega je umeščen in njegova pozicija po X in Y osi.



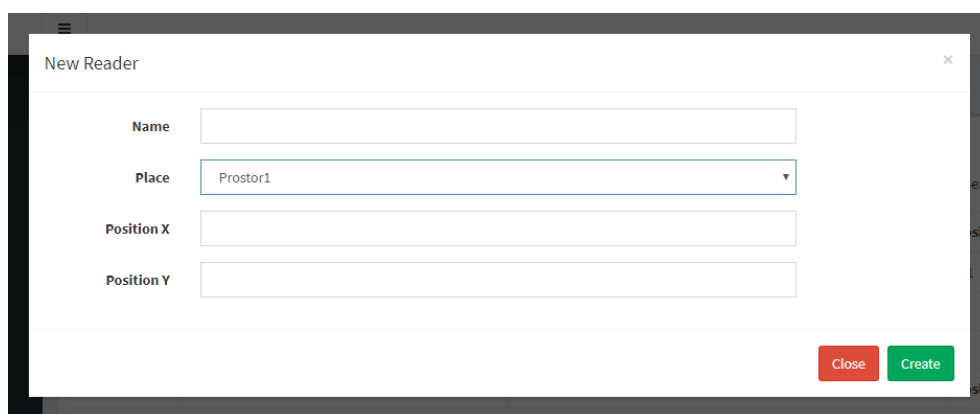
The screenshot shows the BLECloud interface. On the left is a dark sidebar with navigation options: Home, Places, Readers (selected), and Devices. The main content area is titled 'Readers' and features a 'New +' button. Below the title is a 'Hover Data Table' with a search bar and a 'Show 10 entries' dropdown. The table contains three rows of reader data:

Id	Name	Place	Position X	Position Y
1	Sprejemnik1	Prostor1	2.7	0
2	Sprejemnik2	Prostor1	0	0
3	Sprejemnik3	Prostor1	2.3	2.6

At the bottom of the table, it says 'Showing 1 to 3 of 3 entries' and includes 'Previous', '1', and 'Next' navigation buttons.

Slika 4.6: Pregled vseh sprejemnikov

S pritiskom na gumb „New“ se odpre modalno okno, kjer lahko dodamo nov sprejemnik, vidno na sliki 4.7.



The screenshot shows a modal window titled 'New Reader' with a close button (X) in the top right corner. The form contains four input fields:

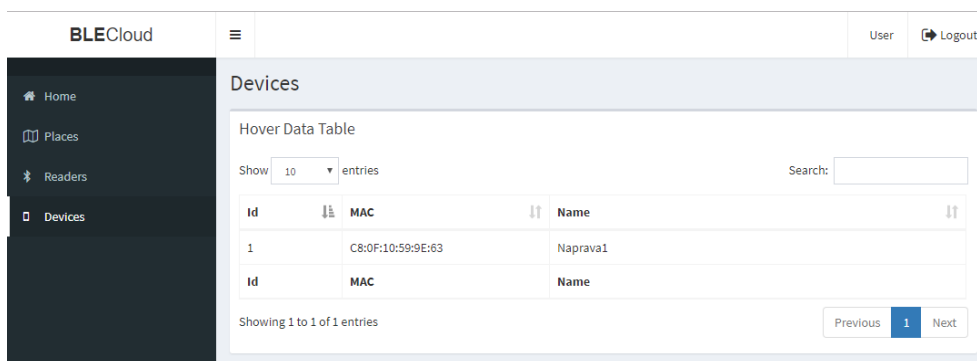
- Name**: A text input field.
- Place**: A dropdown menu with 'Prostor1' selected.
- Position X**: A text input field.
- Position Y**: A text input field.

At the bottom right of the modal, there are two buttons: a red 'Close' button and a green 'Create' button.

Slika 4.7: Dodajanje sprejemnika

4.4.3 Naprave

Na sliki 4.8 je prikazan pogled spletne platforme, kjer je vidna tabela s podatki o napravah. V njej je za vsako napravo prikazan unikatni identifikator, njen naziv in MAC naslov.



Slika 4.8: Pregled vseh naprav

S pritiskom na gumb „New“ se odpre modalno okno, kjer lahko dodamo novo napravo, vidno na sliki 4.9.

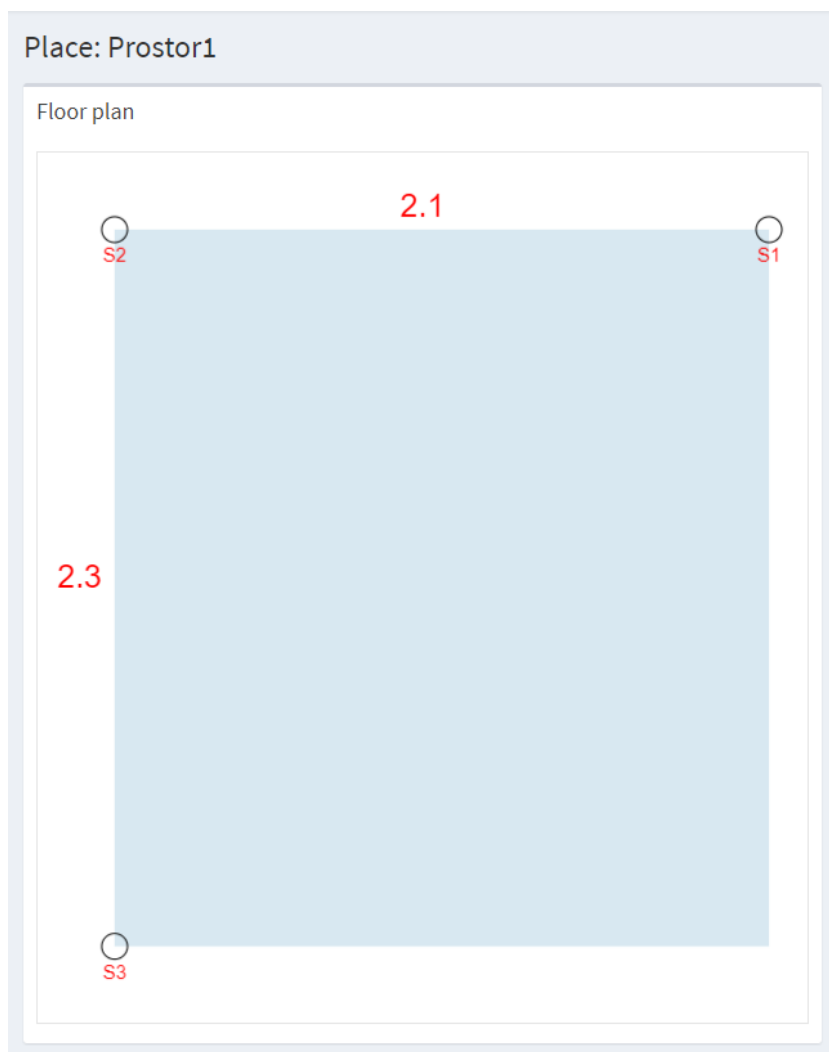


Slika 4.9: Dodajanje naprave

4.4.4 Pregled prostora

V tabeli, kjer so prikazani vsi prostori (4.4), s klikom na gumb „Open“ odpremo prostor. Prikaže se tloris prostora viden na sliki 4.10. Na njem

sta prikazana širina in dolžina prostora. Ustrezno z njihovimi lokacijami so izrisani vsi sprejemniki.



Slika 4.10: Pregled prostora

Poglavje 5

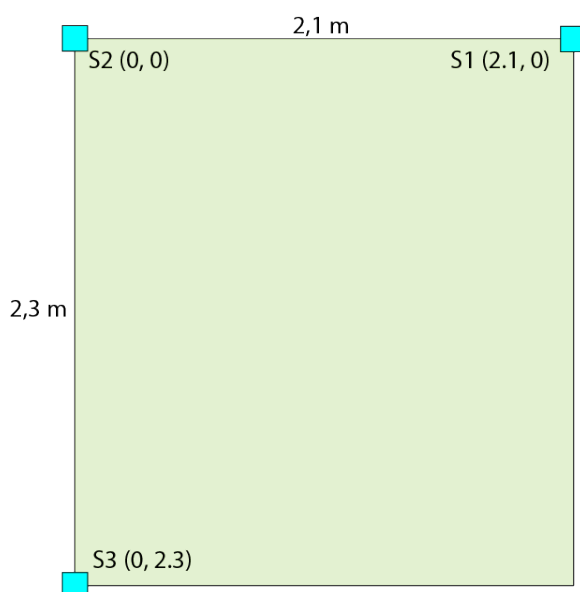
Pregled rezultatov

Sistem je bil testiran v pravokotnem prostoru širine 2,1 metra in dolžine 2,3 metra. V prostor so bili umeščeni trije sprejemniki. Prvi sprejemnik „S1“ na relativnih koordinatah (2.1, 0), sprejemnik „S2“ na koordinatah (0, 0) in tretji sprejemnik „S3“ na koordinatah (0, 2.3). Tloris prostora s sprejemniki je viden na sliki 5.1. V prostor je bil dodan WiFi usmerjevalnik, na katerega so se povezale naprave in centralno središče. Tako je sistem za boljše delovanje in manjšo latenco deloval v lokalni povezavi. Za namen testiranja sistema je bila uporabljena pametna zapestnica podjetja Xiaomi, Xiaomi Mi Band. Zapestnica ima vgrajen BLE, njena baterija pa z enim polnjenjem zdrži nekaj dni. Iz prostora sem odstranil vse predmete, ki bi lahko vplivali na gibanje signala in tako izločil dejavnike, ki bi privedli do napak pri njegovem prenosu.

Zapestnico sem odložil na statično lokacijo na sredino desnega dela prostora, koordinate (1.575, 1.15). Sistem jo je zaznal in izračunal njeno lokacijo.

Rezultat določitve pozicije naprave:

```
{  
  "Naprava1":  
    {  
      "posX": 1.35 ,
```



Slika 5.1: Tloris testnega prostora, modro so označeni sprejemniki in njihove pozicije

```

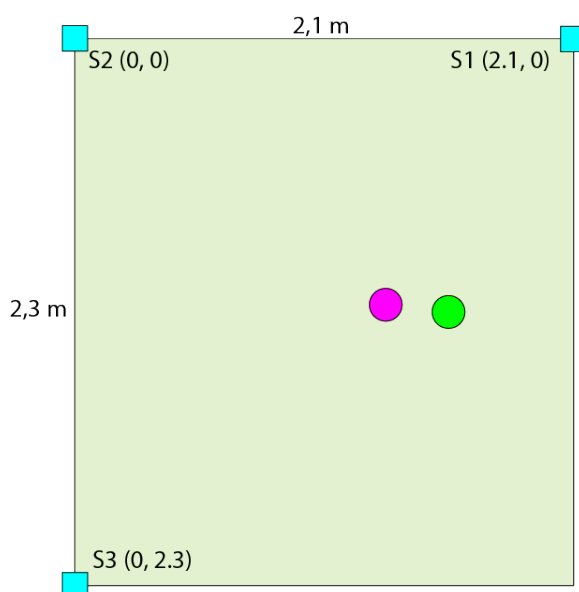
    "posY": 1.12 ,
    "std": 0.33 ,
    "timestamp": "2018-06-10 20:15:35.541265"
  }
}

```

Pri tem izračunu pozicije naprave je prišlo do manjše napake. Po osi X se je sistem zmotil za 22,5 cm, po osi Y pa samo 3 cm. Razlika med izračunano in resnično pozicijo naprave je vidna na sliki 5.2, na sliki 5.3 pa je prikazan izris prostora iz spletne platforme.

Daljša testiranja sistema je pokazalo, da pri izračunih lahko pride do napak tudi do 2 metra. Do tega najpogosteje privedejo odboji signalov od različnih površin. Tako do sprejemnika en signal pride večkrat po različnih poteh, posledično z različno močjo.

Ker je delovanje sistema zelo odvisno od prenosa signala, sem med oddajnik in enega od sprejemnikov postavil monitor. Oddajnik in sprejemniki

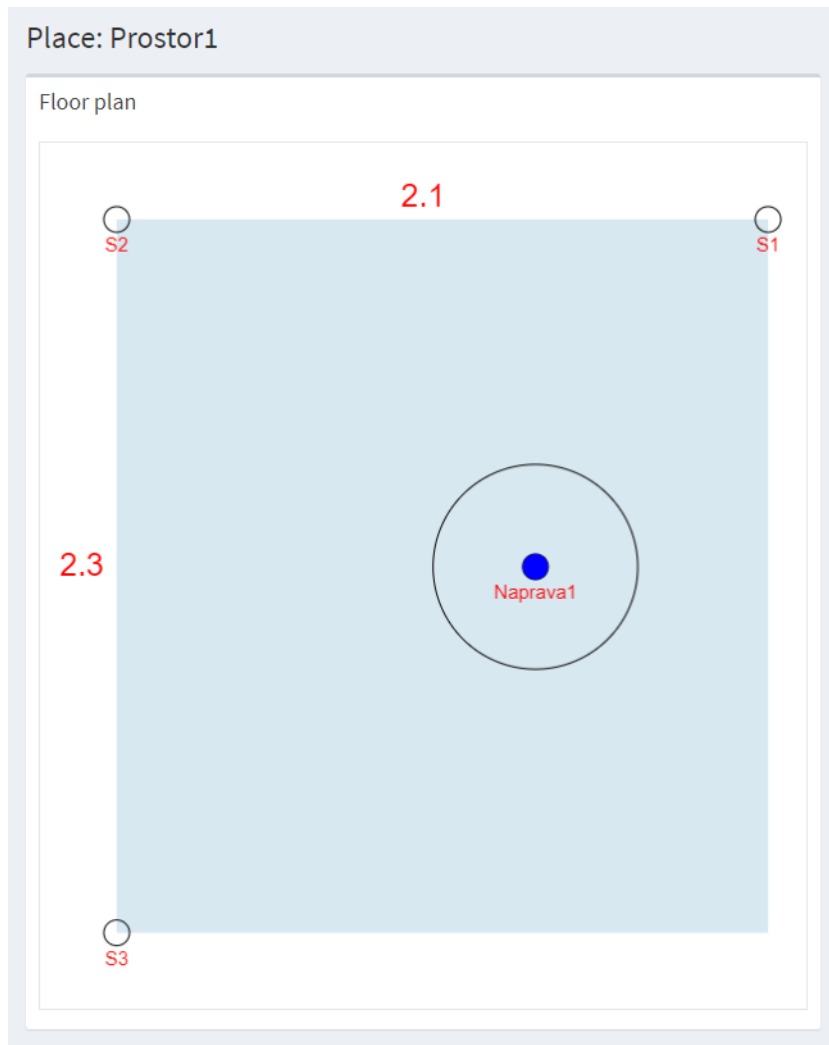


Slika 5.2: Prikaz prostora s sprejemniki in napravo. Zeleno je označena dejanska lokacija naprave, roza pa izračunana lokacija.

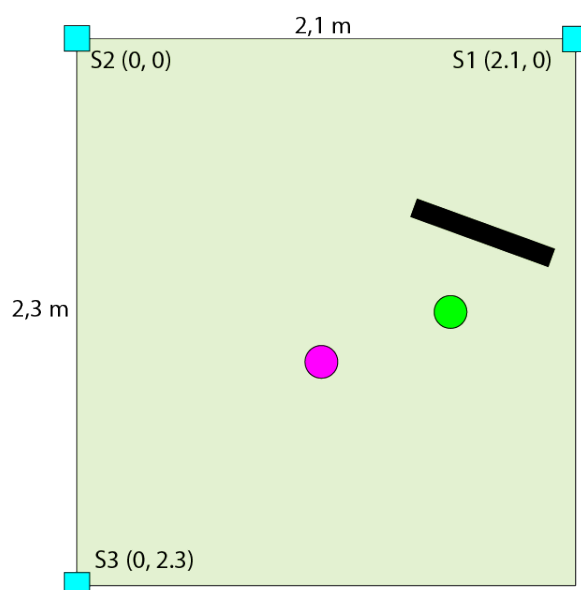
so ostali na istih pozicijah. Monitor sem postavil med sprejemnikom „S1“ in oddajnikom. Odstopanje med dejansko lokacijo in izračunano lokacijo naprave je bilo v večini izračunov večje. Povečala se je tudi možna napaka lokacije, kar privede do večjega območja, v katerem se naprava lahko nahaja.

Izračun lokacije naprave, kjer pot signala ovira monitor:

```
{  
  "Naprava1":  
    {  
      "posX": 1.07,  
      "posY": 1.36,  
      "std": 0.56,  
      "timestamp": "2018-06-10 20:18:46.751324"  
    }  
}
```



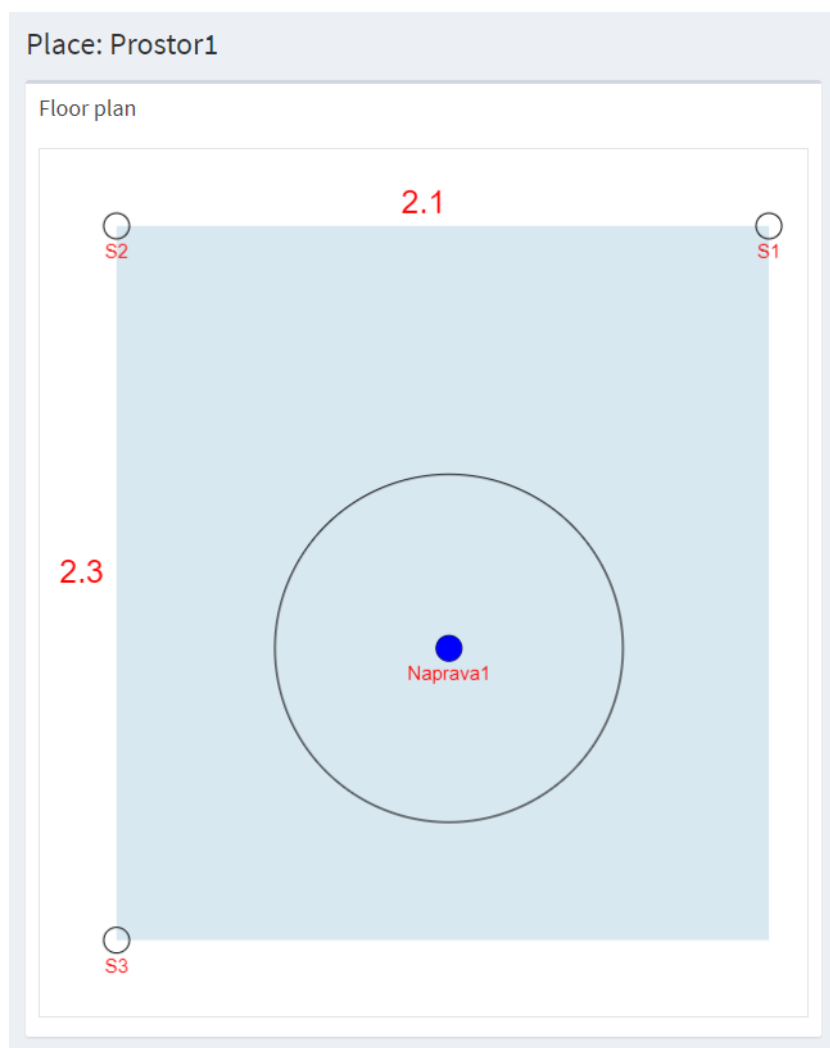
Slika 5.3: Pregled prostora iz spletne platforme



Slika 5.4: Prikaz prostora s sprejemniki in napravo. Zeleno je označena dejanska lokacija naprave, roza izračunana lokacija in črno monitor.

Razlika med izračunano pozicijo in resnično pozicijo naprave, kjer je v prostor dodan monitor, je vidna na sliki 5.4, na sliki 5.5 pa prikaz prostora iz spletne platforme.

Testiranje je pokazalo, da je lahko takšen sistem nezanesljiv oziroma ne moremo določiti točne lokacije oddajnika, saj nanj hitro vplivajo zunanji dejavniki. Da bi jih izločili, bi morali implementirati več različnih pristopov določitve pozicije in jih med seboj povezati. Sistem je velik potencial pokazal v primerih, ko ne potrebujemo točne lokacije naprave, ampak le njen približek. Z uporabo BLE oddajnikov lahko hitro ugotovimo, kdaj je neka naprava prišla v območje sprejemnika. Tako lahko v realnem času spremljamo, katere naprave se nahajajo v območju sprejemnika oziroma v prostoru in ugotavljamo njihove prehode.



Slika 5.5: Pregled prostora iz spletne platforme, kjer je v prostor dodan monitor

Poglavje 6

Sklepne ugotovitve

Sistem za pozicioniranje v zaprtih prostorih lahko implementiramo na več načinov. Cilj diplomske naloge je bil implementirati takšen sistem s pomočjo BLE oddajnikov in statičnih sprejemnikov umeščenih v prostor. Dodana je bila tudi spletna aplikacija, preko katere lahko lažje dodajamo naprave, sprejemnike in definiramo prostore, ki jih želimo spremljati. Za razvoj sistema sem moral povezati veliko različnih tehnologij. Za sprejemnike sem izbral cenovno ugodne in zmogljive računalnike fundacije The Raspberry Pi, programska koda, ki teče na njih, pa je napisana v programskem jeziku Python. V tem jeziku je napisano tudi centralno središče, ki s sprejemniki komunicira preko MQTT protokola. Programski jezik Python ima velik nabor različnih knjižnic, ki nam olajšajo delo pri komunikaciji med napravami, povezavo z bazo podatkov in analitiko surovih podatkov. Spletno aplikacijo sem implementiral v odprtokodnem ogrodju Sails, ki temelji na ogrodju NodeJS. Ogrodje ima implementiran ORM in nam tako olajša upravljanje podatkov v bazi podatkov in omogoča hiter razvoj spletnih aplikacij, saj sam generira API za podatkovne modele.

Sistem je implementiran modularno, kar pomeni, da lahko nekatere dele zamenjamo ali pa jih uporabljamo kot samostojne komponente. Namesto sprejemnikov Raspberry Pi lahko uporabimo računalnike drugih podjetij, saj je programski jezik Python podprt na vseh večjih platformah. V pri-

meru večje zgradbe bi lahko ob pravilni konfiguraciji sistema uporabili več centralnih središč in tako zmanjšali obremenitev sistema. Rešitev je cenovno ugodna, saj skupna cena sprejemnikov in enega oddajnika znaša 70 evrov.

Testiranje je pokazalo, da na delovanje sistema vpliva več dejavnikov. Določanje točne lokacije oddajnika izključno z uporabo BLE signala je mogoča samo v idealnih pogojih, kjer med sprejemniki in oddajniki ni ovir, saj bluetooth signal ni dovolj močan, da to nanj ne bi vplivalo. Sistem se je izkazal v primerih, ko želimo ugotoviti, ali je neka naprava v prostoru ali ne in za premike med prostori. Če bi želeli izboljšati natančnost sistema, bi morali v prostor umestiti več kot tri sprejemnike, lahko bi ga nadgradili s še boljšimi oddajniki in spremljanjem drugih signalov, kot je WiFi.

V prihodnje se bom z razvojem in nadgradnjo implementiranega sistema še ukvarjal, saj vidim velik potencial na trgu sledenja osnovnih sredstev v podjetjih.

Literatura

- [1] S Chan and Gunho Sohn. Indoor localization using wi-fi based fingerprinting and trilateration techniques for lbs applications. XXXVIII-4/C26:1-5, 06 2012.
- [2] Martin Schüssel. Angle of arrival estimation using wifi and smartphones. 2016.
- [3] Unknown. About python. Dosegljivo: <https://www.python.org/about/>, 2018. [Dostopano 03.06.2018].
- [4] Unknown. Bluetooth low energy - part 1: Introduction to ble. Dosegljivo: <https://www.mikroe.com/blog/bluetooth-low-energy-part-1-introduction-ble>, 2018. [Dostopano 05.06.2018].
- [5] Unknown. Global indoor location market analysis (2017-2023). Dosegljivo: <https://www.reportlinker.com/p05207399/Global-Indoor-Location-Market-Analysis.html>, 2018. [Dostopano 18.06.2018].
- [6] Unknown. Mqtt frequently asked questions. Dosegljivo: <http://mqtt.org/faq>, 2018. [Dostopano 08.06.2018].
- [7] Unknown. Mvc architecture. Dosegljivo: <http://www.tutorialsteacher.com/mvc/mvc-architecture>, 2018. [Dostopano 15.05.2018].

-
- [8] Unknown. Raspberry pi 3 model b. Dosegljivo: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>, 2018. [Dostopano 08.06.2018].
- [9] Unknown. Raspberry pi zero w. Dosegljivo: <https://www.raspberrypi.org/products/raspberry-pi-zero-w/>, 2018. [Dostopano 08.06.2018].
- [10] Unknown. Try our improved proximity experience in 5 minutes with the new demo. Dosegljivo: <https://blog.estimote.com/post/166210528555/try-our-improved-proximity-experience-in-5-minutes>, 2018. [Dostopano 21.08.2018].
- [11] Unknown. What is mysql? Dosegljivo: <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>, 2018. [Dostopano 08.06.2018].
- [12] Unknown. What is raspberry pi? Dosegljivo: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>, 2018. [Dostopano 08.06.2018].
- [13] Unknown. What is sails? Dosegljivo: <https://sailsjs.com/whats-that>, 2018. [Dostopano 08.06.2018].
- [14] Eric Weisstein. Circle-circle intersection. Dosegljivo: <http://mathworld.wolfram.com/Circle-CircleIntersection.html>, 2018. [Dostopano 15.05.2018].