

Izrada sustava pozicioniranja unutar prostora pomoću Bluetooth bežične tehnologije

Puljić, Ivana

Master's thesis / Diplomski rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:888000>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-01-09**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Diplomski sveučilišni studij

**IZRADA SUSTAVA POZICIONIRANJA UNUTAR
PROSTORA POMOĆU BLUETOOTH BEŽIČNE
TEHNOLOGIJE**

Diplomski rad

Ivana Puljić

Osijek, 2021

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit

Osijek, 20.09.2021.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za diplomski ispit

Ime i prezime studenta:	Ivana Puljić
Studij, smjer:	Diplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	D-951R, 05.10.2017.
OIB studenta:	49902415699
Mentor:	Prof.dr.sc. Goran Martinović
Sumentor:	Dino Kurtagić
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Izv. prof. dr. sc. Krešimir Nenadić
Član Povjerenstva 1:	Prof.dr.sc. Goran Martinović
Član Povjerenstva 2:	Izv. prof. dr. sc. Mirko Köhler
Naslov diplomskog rada:	Izrada sustava pozicioniranja unutar prostora pomoću Bluetooth bežične tehnologije
Znanstvena grana rada:	Programsko inženjerstvo (zn. polje računarstvo)
Zadatak diplomskog rada:	Korištenjem Bluetooth bežičnih modula i baznih stanica baziranih na željenoj mikroupravljačkoj razvojnoj platformi potrebno je napraviti mrežu stanica koje će kreirati polje Bluetooth signala. Pritom je potrebno sklopovski razviti skupove baznih stanica za primanje i bežičnih modula za odašiljanje Bluetooth signala te ih organizirati u prostoru kako bi mogli precizno odrediti položaj odašiljača u signalnom dometu. Ideja ovog rada je dokazati da se ovim postupkom može razviti sustav koji bi pratio rukometnu ili nogometnu utakmicu i to njihovo kretanje u prostoru kao analizu učinka pojedinog igrača, a ostavio prostora za nadogradnju kompletnog sustava odgovarajućim dodatnim senzorima koji bi pratili sve vitalne funkcije igrača, te njegovu
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	20.09.2021.
Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 28.09.2021.

Ime i prezime studenta:

Ivana Puljić

Studij:

Diplomski sveučilišni studij Računarstvo

Mat. br. studenta, godina upisa:

D-951R, 05.10.2017.

Turnitin podudaranje [%]:

13

Ovom izjavom izjavljujem da je rad pod nazivom: **Izrada sustava pozicioniranja unutar prostora pomoću Bluetooth bežične tehnologije**

izrađen pod vodstvom mentora Prof.dr.sc. Goran Martinović

i sumentora Dino Kurtagić

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

Sadržaj

1. UVOD	1
2. IZAZOVI I STANJE U PODRUČJU SUSTAVA POZICIONIRANJA	2
2.1. Sustav pozicioniranja unutar prostora i korištene metode lociranja unutar prostora	2
2.2. Određivanje udaljenosti od baznih stanica	4
2.3. Određivanje položaja pokretne stanice metodom trilateracije	4
2.4. Postojeća slična rješenja	6
3. SKLOPOVSKO I PROGRAMSKO OSTVARENJE SUSTAVA.....	7
3.1. Opis korištenih tehnologija.....	7
3.1.1. Bežična tehnologija Bluetooth	7
3.1.2. Razvojna platforma Arduino	9
3.1.3. Razvojna okolina Android	10
3.2. Sklopovski dio sustava	11
3.2.1. Položaj baznih stanica	11
3.2.2. Promjena naziva baznih stanica	12
3.3. Programsko rješenje	15
3.3.1. Korisničko sučelje aplikacije	15
3.3.2. Android Manifest	17
3.3.3. Izračun lokacije pametnog telefona.....	18
3.3.4. Prikaz prethodnih lokacija.....	21
3.3.5. Prikaz lokacije pametnog telefona	22
4. PRIKAZ NAČINA RADA I ISPITIVANJE SUSTAVA S ANALIZOM REZULTATA ...	25
4.1. Prikaz načina rada aplikacije	25
4.2. Testiranje preciznosti određivanja lokacije	27
4.3. Analiza rezultata za provedena mjerenja	42
4.4. Primjena sustava u analizi rukometne utakmice.....	43
5. ZAKLJUČAK	46
LITERATURA.....	47
SAŽETAK.....	50
ABSTRACT	51
ŽIVOTOPIS	52
PRILOZI.....	53

1. UVOD

U današnjem svijetu tehnologija je sveprisutna i na mnoge načine olakšava svakodnevne aktivnosti. Globalni pozicijski sustav (GPS) omogućuje jednostavno i precizno određivanje trenutnog položaja bilo gdje na Zemlji. Nažalost, GPS tehnologija ne može se koristiti unutar zatvorenih prostora jer se signal značajno gubi prolaskom kroz zidove. Iz tog razloga se za pozicioniranje unutar prostora koriste druge tehnologije, poput bežičnih tehnologija Wi-Fi i Bluetooth. Glavni zadatak ovog rada je izraditi sustav za pozicioniranje unutar prostora koristeći bežičnu tehnologiju Bluetooth. Bežična tehnologija Bluetooth dostupna je svakoj osobi koja posjeduje mobilni uređaj, niske je cijene i troši malo energije pa je zbog toga idealna za pozicioniranje unutar prostora. Ideja ovog rada je dokazati da se ovim postupkom može razviti sustav koji bi pratio rukometnu ili nogometnu utakmicu analizom kretanja pojedinog igrača. Za potrebe realizacije ovog diplomskog rada izrađen je sustav koji se sastoji od 4 bazne stanice koje predstavljaju bežični Bluetooth HC-06 moduli te pokretne stanice koju predstavlja Android pametni telefon.

U drugom poglavlju ovog rada opisane su metode koje se koriste u sustavima pozicioniranja unutar prostora, način izračuna udaljenosti temeljen na jačini primljenog signala i izračun lokacije pokretne stanice koristeći metodu trilateracije. U trećem poglavlju opisane su tehnologije korištene u realizaciji sustava: Bluetooth, Arduino i Android. Treće poglavlje također sadrži opis sklopovlja i programsko rješenje, odnosno izradu Android aplikacije za pozicioniranje unutar prostora. U zadnjem poglavlju prikazan je način rada izrađene mobilne aplikacije, prikaz rezultata testiranja preciznosti određivanja lokacije te primjena realiziranog sustava na analizu rukometne utakmice.

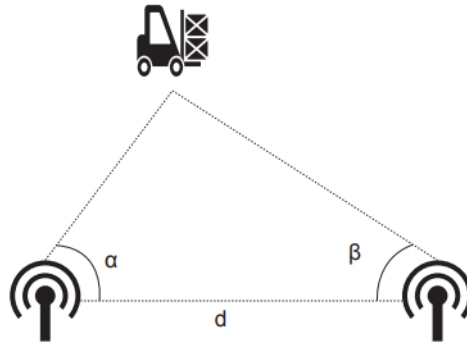
2. IZAZOVI I STANJE U PODRUČJU SUSTAVA POZICIONIRANJA

Unutar ovog poglavlja objašnjen je pojam sustava pozicioniranja unutar prostora te metode koje se koriste za lociranje unutar prostora. Također, objašnjen je izračun udaljenosti između bazne i pokretne stanice koristeći primljenu vrijednost RSSI, računanje položaja pokretne stanice metodom trilateracije te dostupna postojeća rješenja.

2.1. Sustav pozicioniranja unutar prostora i korištene metode lociranja unutar prostora

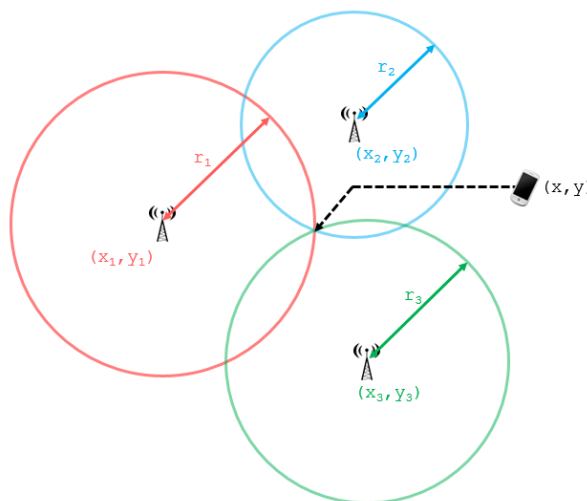
Sustav pozicioniranja unutar prostora (engl. *Indoor Positioning System*) se definira kao sustav mrežno povezanih uređaja korištenih za bežično lociranje objekata i osoba unutar zgrada i djelomično pokrivenih područja [1]. Zbog konstrukcijskih materijala, GPS (skraćeno od engl. *Global Positioning System*) značajno gubi signal unutar zgrada, tunela, podzemnih lokacija i sl. pa se ne može koristiti za lociranje unutar zatvorenih prostora. Sustav pozicioniranja unutar prostora za lociranje koristi radio valove, magnetna polja, zvučne signale ili prikupljene podatke sa senzora [2]. Sustavi za pozicioniranje unutar prostora generiraju podatke koji su nužni za izračun trenutne lokacije određenog korisnika [3]. RSSI (skraćeno od engl. *Receive Signal Strength Indicator*) predstavlja indikator jačine primljenog signala, TOA (skraćeno od engl. *Time of Arrival*) predstavlja vrijeme dolaska signala, TDOA (skraćeno od engl. *Time Difference of Arrival*) predstavlja vremensku razliku dolaska signala, dok AOA (skraćeno od engl. *Angle of Arrival*) predstavlja kut dolaska signala. Dvije najčešće metode korištene za izradu sustava pozicioniranja unutar prostora su triangulacija i trilateracija.

Lociranje metodom triangulacije koristi geometrijska načela trokuta i kutova [4]. Vrijednosti indikatora jačine primljenog signala RSSI koriste se za izračun udaljenosti između bazne i pokretne stanice. Da bi ova metoda radila, potrebne su dvije ili više baznih stanica postavljenih tako da prekrivaju područje cijele prostorije. Bazna stanica mjeri kut pod kojim dolazi signal s pokretne stanice te ograničava položaj pokretne stanice na liniju koja prolazi i kroz pokretnu i kroz baznu stanicu. Svaka bazna stanica definira svoju liniju mogućih lokacija pokretne stanice te se konačni položaj pokretne stanice nalazi u presjeku svih tih linija [5]. Na slici 2.1 prikazan je princip određivanja lokacije koristeći metodu triangulacije, gdje α i β predstavljaju oznake za kut dolaska signala, dok d predstavlja udaljenost između dvije bazne stanice.



Sl. 2.1. Princip određivanja lokacije metodom triangulacije [6]

Za razliku od metode triangulacije koja za određivanje položaja koristi kut dolaska signala, metoda trilateracije za određivanje položaja koristi udaljenosti od baznih do pokretne stanice. Za određivanje lokacije objekta u dvodimenzionalnom prostoru potrebne su koordinate tri bazne stanice te njihova udaljenost od objekta (pokretne stanice) čija se lokacija pokušava odrediti [7]. Kod ove metode svaka udaljenost bazne stanice od pokretne predstavlja polumjer kružnice te se konačni položaj pokretne stanice nalazi u presjeku tih kružnica. Na slici 2.2 prikazan je princip određivanja lokacije koristeći metodu trilateracije, gdje (x_1, y_1) , (x_2, y_2) i (x_3, y_3) predstavljaju položaj baznih stanica, r_1 , r_2 i r_3 predstavljaju udaljenosti od baznih do pokretne stanice te (x, y) predstavlja položaj pokretne stanice.



Sl. 2.2. Princip određivanja lokacije metodom trilateracije [8]

Za realizaciju sustava pozicioniranja unutar prostora odabrana je metoda trilateracije. U sljedeća dva potpoglavlja objašnjeno je računanje udaljenosti između bazne i pokretne stanice te računanje konačne lokacije pokretne stanice.

2.2. Određivanje udaljenosti od baznih stanica

Za realizaciju sustava koristeći metodu trilateracije potrebno je odrediti udaljenost od pokretne do baznih stanica. RSSI (skraćeno od engl. *Received Signal Strength Indicator*) je mjera jakosti primljenog signala koji odašilje bazna stanica, a prima pokretna stanica. RSSI se može koristiti za procjenu udaljenosti na temelju vrijednosti primljenog signala baznih stanica [9]. Što je veća udaljenost od bazne stanice, vrijednost signala koji prima pokretna stanica je manja. U idealnim uvjetima pretpostavka je da je jačina primljenog signala obrnuto proporcionalna kvadratu udaljenosti. U stvarnim uvjetima faktori poput refrakcije, difrakcije i raspršenja valova uzrokovanog objektima u blizini mogu ometati signal koji putuje zatvorenim prostorom [10]. Izraz (2-1) objašnjava slabljenje na koje signal nailazi prolaskom kroz zgradu ili gusto naseljeno područje [9]:

$$RSSI = -20 * \log(d) + A \quad (2-1)$$

gdje je:

- RSSI - jakost primljenog signala koji odašilje bazna stanica, izražen u dBm
- d - udaljenost između bazne i pokretne stanice
- A - jakost primljenog signala kada je udaljenost između bazne i pokretne stanice 1 metar

Ako se izraz (2-1) zapiše po varijabli d, dobije se izraz (2-2) iz kojeg se može izračunati udaljenost između bazne i pokretne stanice:

$$d = 10^{\frac{A-RSSI}{20}} \quad (2-2)$$

2.3. Određivanje položaja pokretne stanice metodom trilateracije

Nakon određivanja udaljenosti pokretne stanice od svake bazne stanice, slijedi izračun konačne lokacije pokretne stanice. Za izračun položaja pokretne stanice odabrana je metoda trilateracije. Da bi se odredio položaj pokretne stanice u dvodimenzionalnom prostoru, potrebne su koordinate tri najbliže bazne stanice te njihove udaljenosti od pokretne stanice [7].

Izrazi (2-3), (2-4) i (2-5) predstavljaju jednadžbe tri kružnice. Položaj pokretne stanice nalazi se u presjeku te tri kružnice gdje (x_1, y_1) , (x_2, y_2) i (x_3, y_3) predstavljaju poznate koordinate tri najbliže

bazne stanice, vrijednosti r_1 , r_2 i r_3 predstavljaju udaljenost bazne i pokretne stanice, dok (x, y) predstavlja nepoznate koordinate pokretne stanice.

$$(x - x_1)^2 + (y - y_1)^2 = r_1^2 \quad (2-3)$$

$$(x - x_2)^2 + (y - y_2)^2 = r_2^2 \quad (2-4)$$

$$(x - x_3)^2 + (y - y_3)^2 = r_3^2 \quad (2-5)$$

Proširivanjem izraza (2-3), (2-4) te (2-5) dobiju se izrazi (2-6), (2-7) te (2-8).

$$x^2 - 2x_1x + x_1^2 + y^2 - 2y_1y + y_1^2 = r_1^2 \quad (2-6)$$

$$x^2 - 2x_2x + x_2^2 + y^2 - 2y_2y + y_2^2 = r_2^2 \quad (2-7)$$

$$x^2 - 2x_3x + x_3^2 + y^2 - 2y_3y + y_3^2 = r_3^2 \quad (2-8)$$

Oduzimanjem izraza (2-7) od izraza (2-6) te izraza (2-8) od izraza (2-7) dobiju se izrazi (2-9) i (2-10).

$$(-2x_1 + 2x_2)x + (-2y_1 + 2y_2)y = r_1^2 - r_2^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2 \quad (2-9)$$

$$(-2x_2 + 2x_3)x + (-2y_2 + 2y_3)y = r_2^2 - r_3^2 - x_2^2 + x_3^2 - y_2^2 + y_3^2 \quad (2-10)$$

Na kraju, rješenje sustava jednažbi s dvije nepoznanice daje izraze (2-11) i (2-12) za izračun precizne lokacije pokretne stanice.

$$x = \frac{(r_1^2 - r_2^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2)(-2y_2 + 2y_3) - (r_2^2 - r_3^2 - x_2^2 + x_3^2 - y_2^2 + y_3^2)(-2y_1 + 2y_2)}{(-2y_2 + 2y_3)(-2x_1 + 2x_2) - (-2y_1 + 2y_2)(-2x_2 + 2x_3)} \quad (2-11)$$

$$y = \frac{(r_1^2 - r_2^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2)(-2x_2 + 2x_3) - (-2x_1 + 2x_2)(r_2^2 - r_3^2 - x_2^2 + x_3^2 - y_2^2 + y_3^2)}{(-2y_1 + 2y_2)(-2x_2 + 2x_3) - (-2x_1 + 2x_2)(-2y_2 + 2y_3)} \quad (2-12)$$

2.4. Postojeća slična rješenja

Budući da se GPS tehnologija ne može koristiti za lociranje unutar zatvorenih prostora, mnoge tvrtke bave se razvojem sustava za unutarnje pozicioniranje. U ovom potpoglavlju bit će objašnjena dva postojeća rješenja sustava pozicioniranja unutar prostora.

Jedno od postojećih rješenja pozicioniranja unutar prostora je HERE Indoor Positioning platforma nizozemske tvrtke HERE Technologies. Platforma nudi razvoj iOS ili Android mobilne aplikacije za pozicioniranje unutar prostora visoke preciznosti te detekciju prostorije i zgrade unutar kojih se korisnik nalazi koristeći Bluetooth odašiljače ili postojeću Wi-Fi infrastrukturu. Za početak razvoja mobilne aplikacije potrebno se registrirati na HERE Developer Portal te preuzeti Software Development Kit (SDK) za Android ili iOS. Za prikupljanje podataka o Bluetooth ili Wi-Fi odašiljačima te kreiranje karte prostora unutar kojeg će se sustav koristiti potrebno je preuzeti Android aplikaciju HERE Indoor Radio Mapper. Platforma podržava Bluetooth odašiljače koji koriste Eddystone ili iBeacon protokole [11]. Eddystone je protokol otvorenog koda koji je razvila tvrtka Google te odašiljače koji koriste Eddystone protokol mogu detektirati i Android i iOS uređaji. Za razliku od odašiljača koji koriste Eddystone protokol, odašiljače koji koriste iBeacon protokol razvijen od strane tvrtke Apple Inc. mogu detektirati samo iOS uređaji [12].

Sustav za pozicioniranje unutar prostora naziva StepInside je također jedno od postojećih rješenja problema pozicioniranja unutar prostora. Sustav je razvijen od strane švedske tvrtke Senion AB. StepInside također nudi Software Development Kit (SDK) koji se može lako integrirati u bilo koju aplikaciju pametnog telefona. Osim prikaza lokacije unutar prostora, ovaj sustav nudi i dodatne opcije poput izvještaja o kretanju korisnika te broju korisnika koji su se nalazili u određenom području u određeno vrijeme. Tehnologija korištena za realizaciju ovog sustava nije izričito spomenuta [1].

3. SKLOPOVSKO I PROGRAMSKO OSTVARENJE SUSTAVA

U ovom poglavlju opisane su tehnologije koje se koriste za sklopovsko i programsko ostvarenje sustava. Također, opisan je sklopovski dio sustava s pripadajućim komponentama te programsko rješenje sustava, odnosno izrada Android aplikacije.

3.1. Opis korištenih tehnologija

3.1.1. Bežična tehnologija Bluetooth

Bluetooth je bežični protokol pomoću kojeg se podaci između dva ili više uređaja razmjenjuju na malim udaljenostima. Razvoj bežične tehnologije Bluetooth započela je kompanija Ericsson 1994. godine. Udruga Bluetooth SIG (skraćeno od engl. *Special Interest Group*), osnovana od strane tvrtki Ericsson, IBM, Intel, Nokia i Toshiba, analizirala je i javno predstavila specifikaciju 20. ožujka 1998. godine [13]. Bluetooth SIG organizacija danas broji više od 35000 pripadnika u cijelom svijetu. Projekt je nazvan Bluetooth prema kralju Danske iz srednjeg vijeka Haroldu Plavozubom (danski *Harald Blåtand*, engl. *Harold Bluetooth*) koji je u desetom stoljeću sjedinio plemena Danske i Norveške koja su ratovala [13].

Bluetooth se zasniva na kreiranju mreže PAN (skraćeno od engl. *Personal Area Network*) korištene za komuniciranje uređaja na kratkim udaljenostima. Razvijen je kao zamjena podatkovnim kablovima RS-232 (skraćeno od engl. *Recommended Standard 232*) standardu [14].

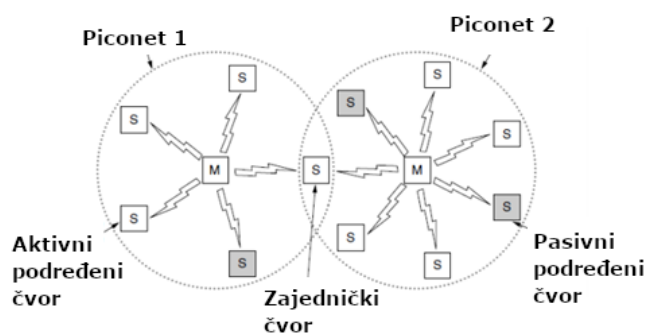
Prve inačice Bluetootha 1.0. i 1.0.B imale su brojne probleme s omogućavanjem međusobne komunikacije između uređaja. Također, problem je predstavljala nemogućnost osiguranja anonimnosti korisnika. Greške koje su nađene u prethodnim Bluetooth inačicama korigirane su u sljedećoj inačici naziva Bluetooth 1.1. Inačica 1.1. dobila je oznaku IEEE (skraćeno od engl. *Institute of Electrical and Electronics Engineers*) standarda 802.15.1-2002. Također, uvedeno je mjerenje RSSI vrijednosti te podrška za nekriptirane signale [14].

Sljedeća inačica Bluetooth 1.2. u cijelosti je usklađena s prethodnom inačicom uz određena unaprjeđenja poput bržeg povezivanja i detekcije uređaja, anonimnog načina rada koji osigurava skrivanje adrese Bluetooth uređaja, poboljšanja kvalitete govora, uvođenja provjere toka podataka, veće brzine prijenosa podataka (do 721 kb/s). Inačica 1.2. poznata je kao IEEE standard 802.15.1-2005. Inačica Bluetooth 2.0. uvodi još brži prijenos podataka korištenjem tehnologije EDR (skraćeno od engl. *Enhanced Data Rate*), čija je normalna brzina oko 3Mbps te smanjenu uporabu energije [14]. Inačica 2.1. uvodi pružanje više informacija prilikom procesa pretraživanja uređaja, pojednostavljenu razmjenu ključa za kriptiranje, poboljšanu sigurnost uparivanja uređaja i

mogućnost sinkronog i asinkronog prijenosa koristeći istu logičku vezu. Sljedeća inačica, Bluetooth 3.0., donosi nova poboljšanja kao što su: uporaba MAC ili fizičkog sloja za prijenos podataka, mogućnost prijenosa podataka bez uspostavljanja eksplicitnih kanala, otkrivanje veličine ključa za kriptiranje te uvod dodatnog nadzora energije [14]. Sljedeća inačica Bluetootha 4.0. donosi tri protokola: Classic Bluetooth, Bluetooth high speed i Bluetooth low energy. Ova inačica omogućuje broadcast, automatsku konfiguraciju Piconet tehnologije te omogućuje prenošenje audio i video signala visoke kvalitete. Posljednje dvije inačice Bluetooth 5 i Bluetooth 5.1. donose veću brzinu prijenosa (do 48Mbps) te poboljšanje lokacije i praćenja uređaja.

Osnovnu jedinicu Bluetooth sustava predstavlja piconet koji je sastavljen od jednog glavnog nadređenog čvora (engl. *master*) koji kontrolira promet i do sedam aktivnih sporednih podređenih čvorova (engl. *slave*) udaljenih do desetak metara. Piconet je vremenski multipleksirani sustav, gdje glavni nadređeni čvor vrši kontrolu nad vremenskim odsječcima te ih dodjeljuje sporednim podređenim čvorovima [15]. Svaki čvor ima svoj sat (engl. *clock*) te pripadajuću unikatnu 48-bitnu IEEE MAC adresu koja označava identifikacijski broj ID uređaja u mreži i programski je promjenjiva. Za međusobnu komunikaciju uređaji se moraju sinkronizirati na isti *clock* te koristiti isti slijed frekvencijskog preskakivanja (engl. *Frequency Hopping Spread Spectrum*). *Clock* glavnog nadređenog čvora se koristi za sinkroniziranje svih drugih uređaja u mreži, dok svaki sporedni podređeni čvor određuje razliku (engl. *offset*) između svog unutarnjeg *clock*-a i *clock*-a glavnog nadređenog čvora što omogućuje razmjenu paketa između nadređenog i podređenog čvora [15].

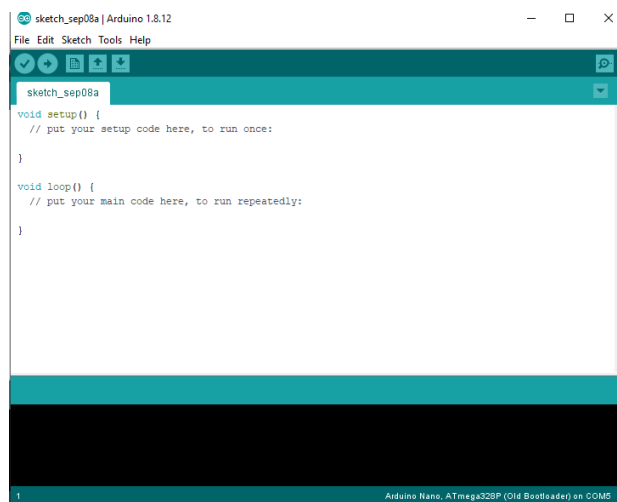
Više piconeta mogu se međusobno povezati posredstvom zajedničkih čvorova i tako formirati scatternet. Povezivanje scatternet mreža postiže se preko jednog čvora koji može predstavljati sporedni podređeni u dvije ili više piconet mreže, ali glavni nadređeni isključivo u jednoj piconet mreži. Na slici 3.1 prikazano je povezivanje dva piconeta u scatternet.



Sl. 3.1. Prikaz dva piconeta povezana u scatternet [15]

3.1.2. Razvojna platforma Arduino

Arduino je razvojna platforma koja se koristi za stvaranje elektroničkih projekata. Dijelovi Arduino razvojne platforme su sklopovski dio, odnosno mikroupravljač, te razvojna pločica pomoću koje je omogućen pristup pinovima mikroupravljača. Osim sklopovskog dijela, Arduino nudi i programsku podršku, odnosno integrirano okruženje za razvoj programa IDE (skraćeno od engl. *Intregrated Development Environment*), koje je potrebno pokrenuti na računalu za upravljanje razvojnom pločicom [16]. Na slici 3.2 prikazano je Arduino integrirano razvojno okruženje.

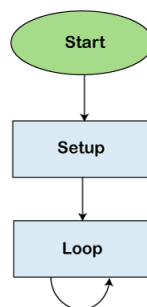


Sl. 3.2. Arduino IDE

Programiranje Arduino pločice obavlja se u prilagođenoj inačici C++ programskog jezika. Program je sastavljen od dva osnovna dijela:

- Setup - funkcija koji se izvodi jednom kada se Arduino pokrene
- Loop - funkcija koja se neprestano ponavlja svo vrijeme dok je Arduino spojen na napajanje

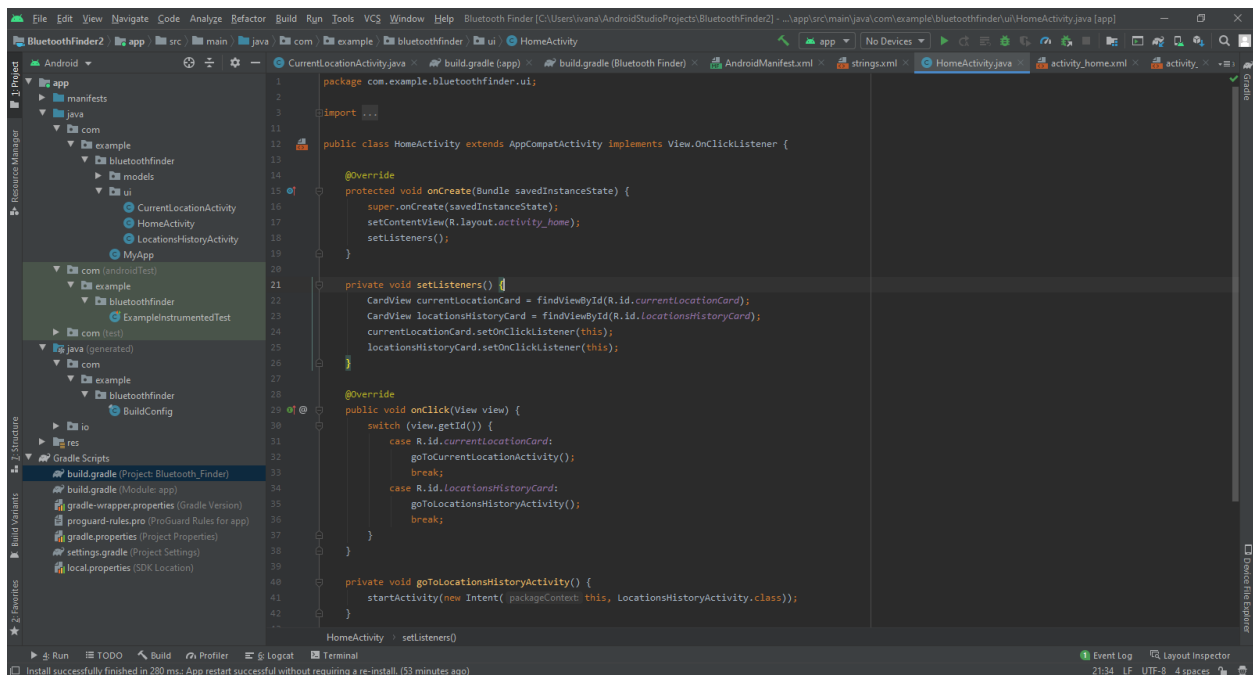
Na slici 3.3 prikazani su glavni dijelovi Arduino koda.



Sl. 3.3. Dijelovi Arduino koda [17]

3.1.3. Razvojna okolina Android

Android je softverska platforma i operacijski sustav otvorenog koda primarno dizajniran za korištenje na uređajima sa zaslonom osjetljivim na dodir. Android operacijski sustav je baziran na modificiranoj inačici Linux jezgre. Android je prva razvijala tvrtka Android Inc, koju je kupio Google 2005. godine. Android je javnosti predstavljen 2007. godine, dok se prvi uređaj baziran na Android operacijskom sustavu pojavio u rujnu 2008. godine [18]. Android operacijski sustav je napisan u programskom jeziku C/C++, dok se za razvoj Android aplikacija koriste Java ili Kotlin programski jezici. Službeno razvojno okruženje za razvoj Android aplikacija je Android Studio. Android Studio temeljen je na integriranom razvojnom okruženju Intelij IDEA i omogućuje brojne funkcionalnosti poput Gradle sustava za izgradnju, emulatora različitih funkcionalnosti, jedinstvenog razvojnog okruženja za Android uređaje, alate za testiranje, itd. Android Studio službeno okruženje je besplatno za preuzimanje te ga omogućuje Apache 2.0. licenca.

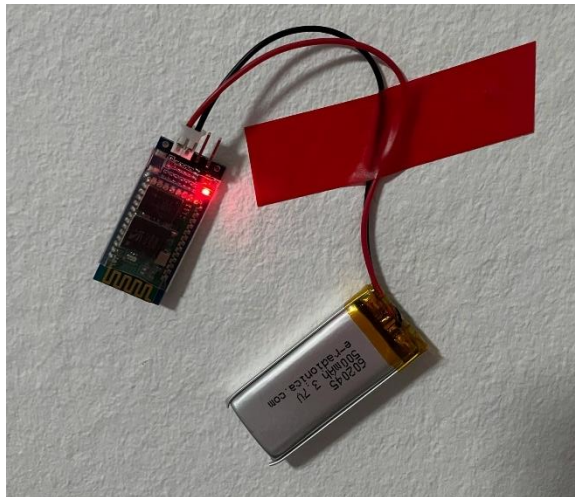


Sl. 3.4. Razvojno okruženje Android Studio

3.2. Sklopovski dio sustava

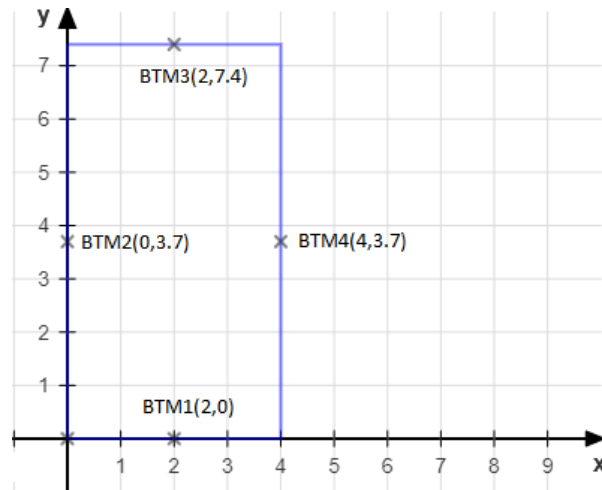
3.2.1. Položaj baznih stanica

Zadatak ovog rada je izraditi sustav pozicioniranja unutar prostora koristeći bežičnu tehnologiju Bluetooth. Sustav za pozicioniranje unutar prostora se sastoji od više baznih stanica za odašiljanje Bluetooth signala te od pokretne stanice koja prima Bluetooth signal. Kao bazna stanica u realizaciji ovog sustava koristi se Bluetooth modul HC-06. Pomoću ovog modula, omogućena je serijska komunikacija Arduina i uređaja koji koristi bežičnu tehnologiju Bluetooth putem bežične Bluetooth veze. Modul koristi Bluetooth 2.0. inačicu i radi na frekvenciji od 2.402 GHz do 2.480GHz [20]. Za napajanje baznih stanica korištena je punjiva litij-ionska baterija. Na slici 3.5 prikazan je Bluetooth HC-06 modul spojen na napajanje.



Sl. 3.5. Bluetooth modul HC-06 spojen na napajanje

Za precizno određivanje položaja pokretne stanice koja prima Bluetooth signal potrebno je pravilno razmjestiti bazne stanice tako da pokrivaju svaku točku unutar prostora. Prostorija za koju je sustav izrađen je širine četiri metra i dužine 7.4 metara. Za izradu sustava koriste se četiri Bluetooth senzora i svaki senzor je postavljen na sredinu jednog zida prostorije na visini od 120 cm. Položaj senzora u prostoriji vidljiv je na slici 3.6, gdje su koordinate senzora prikazane u dvodimenzionalnom koordinatnom sustavu.



Sl 3.6. Prikaz položaja senzora unutar prostorije

3.2.2. Promjena naziva baznih stanica

Za promjenu naziva baznih stanica koristi se Arduino razvojno okruženje. Razvojna pločica koja se koristi za promjenu postavki baznih stanica je Croduino Basic 2, pločica hrvatske proizvodnje 100% kompatibilna s Arudinom Nano. Karakteristike razvojne pločice navedene su u tablici 3.1, dok je na slici 3.7 prikazana razvojna pločica Croduino Basic 2.



Sl. 3.7. Croduino Basic 2 [21]

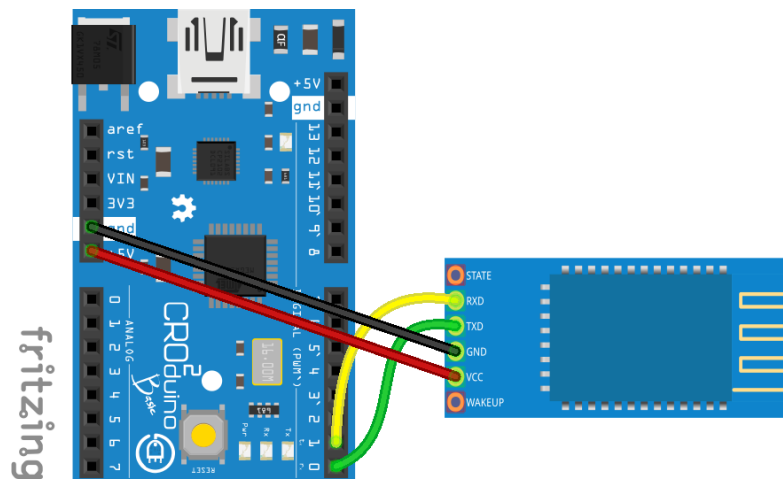
Tab. 3.1. Karakteristike Croduino Basic 2 [22]

Mikrokontroler	Atmel Atmega328
Digitalni input/output pinovi	14
Analogni input pinovi	16
Flash memorija	32 KB
SRAM	2 KB
EEPROM	0.5 KB

Prvi korak u promjeni postavki Bluetooth HC-06 modula je uspješno spajanje pinova modula s pinovima na mikroupravljaču Croduinu Basic 2. Bluetooth HC-06 modul posjeduje 4 pina:

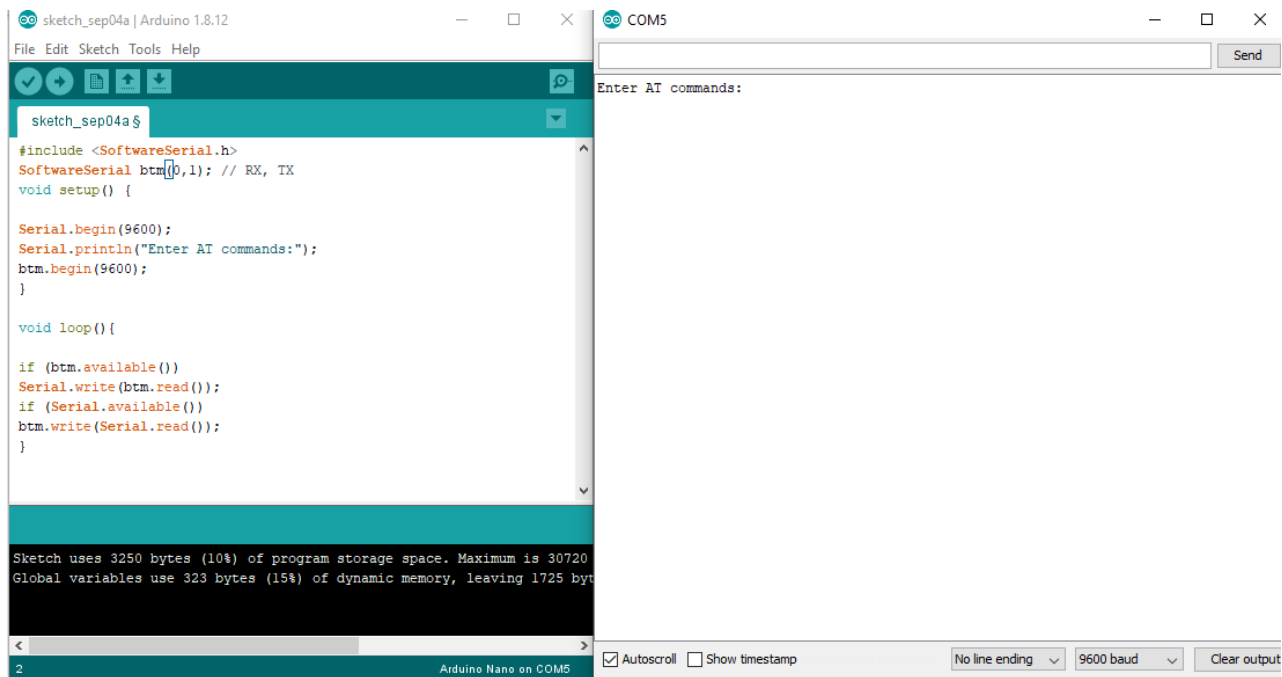
- VCC: napajanje, povezuje se s pinom +5V mikrokontrolera
- GND: uzemljenje, povezuje se s pinom GND mikrokontrolera
- TXD: serijski izlaz modula, povezuje se s digitalnim pinom 0 koji predstavlja rx pin mikrokontrolera
- RXD: serijski ulaz modula, povezuje se s digitalnim pinom 1 koji predstavlja tx pin mikrokontrolera [23]

Na slici 3.8 prikazano je spajanje Bluetooth HC-06 s mikroupravljačem Croduino Basic 2.



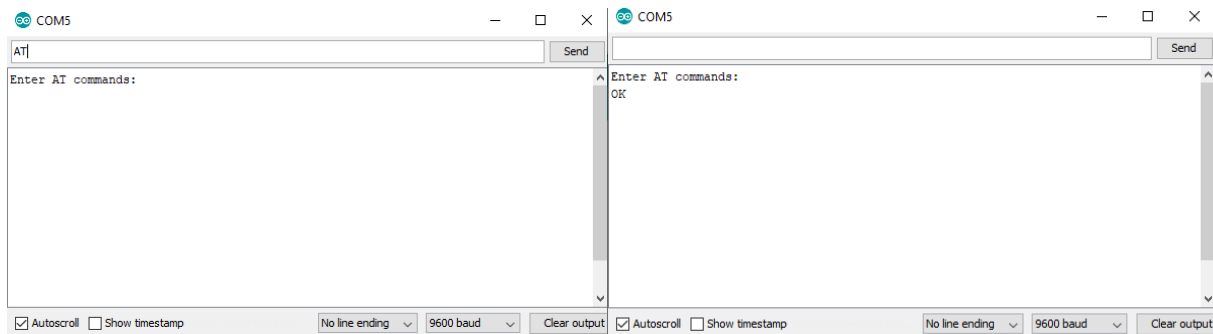
Sl. 3.8. Prikaz spajanja Croduina Basic 2 i Bluetooth modula HC-06 [23]

Nakon uspješnog spajanja Bluetooth modula s mikroupravljačem, slijedi spajanje mikroupravljača s računalom te mijenjanje naziva baznih stanica slanjem AT naredbi Bluetooth modulu. AT naredbe se koriste za ispitivanje i promjenu zadanih postavki Bluetooth modula poput naziva, lozinke te brzine prijenosa podataka (engl. *Baud rate*). Također, koristeći AT naredbe može se provjeriti *firmware* inačica [24]. Na slici 3.9 je prikazan Arduino kod za komunikaciju s Bluetooth modulom i serijski monitor koji služi za primanje i slanje serijskih podataka između računala i Croduina, odnosno Bluetooth modula. AT naredbe upisuju se u serijski monitor.



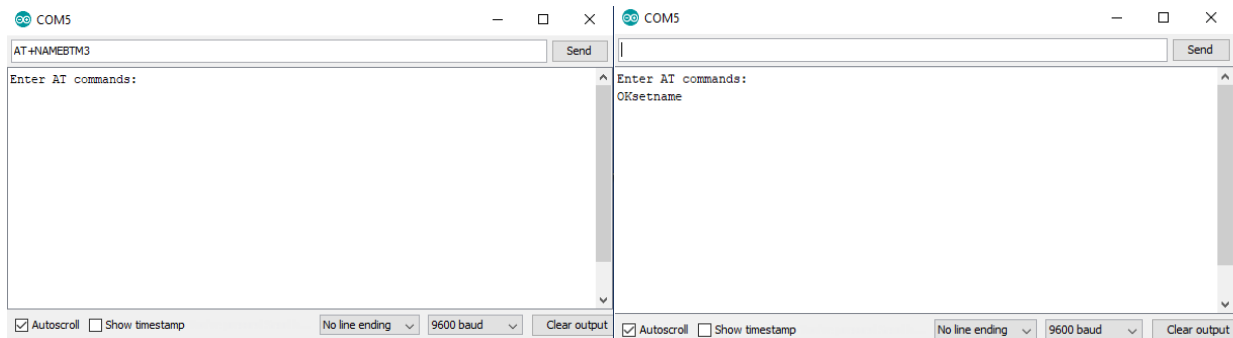
Sl.3.9. Arduino kod za komunikaciju s Bluetooth modulom i serijski monitor

Za provjeru veze s Bluetooth modulom koristi se AT naredba 'AT'. Ako je veza uspostavljena odgovor Bluetooth modula je 'OK' kao što je prikazano na slici 3.10.



Sl. 3.10. Prikaz provjere veze s Bluetooth modulom

Za mijenjanje naziva Bluetooth modula koristi se AT naredba 'AT+NAME'. Uz naredbu se bez razmaka piše naziv koji se želi dodijeliti Bluetooth modulu, npr. 'AT+NAMExyz' ako se modulu dodjeljuje naziv xyz. Ako je naredba prihvaćena i naziv promijenjen, odgovor modula je 'OKsetname', što je vidljivo na slici 3.11. Baznim stanicama, odnosno Bluetooth modulima su dodijeljeni nazivi BTM1, BTM2, BTM3 te BTM4.

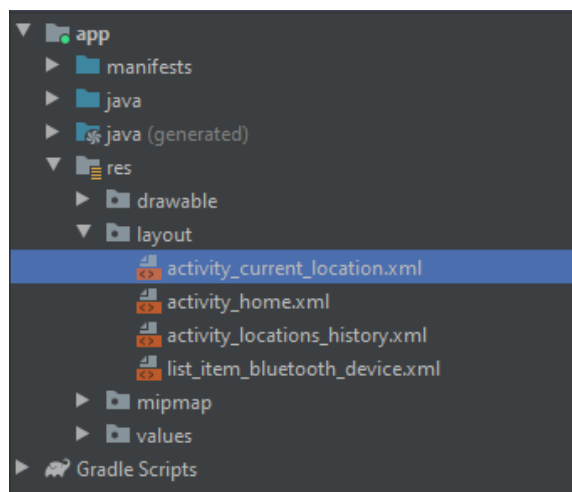


Sl.3.11. Prikaz naredbe za promjenu naziva Bluetooth modula

3.3. Programsko rješenje

3.3.1. Korisničko sučelje aplikacije

Korisničko sučelje predstavlja vezu između računalnog sustava i krajnjeg korisnika. Korisnik upravlja računalom putem korisničkog sučelja korištenjem ulaznih uređaja (miš, tipkovnica, zaslon osjetljiv na dodir, itd.) [25]. Korisničko sučelje Android aplikacije piše se u XML formatu. Datoteke za uređivanje korisničkog sučelja Android aplikacije nalaze se unutar direktorija *res/layout*. Na slici 3.12 prikazana je struktura Android projekta s layout direktorijem.



Sl. 3.12. Prikaz layout direktorija

Za realizaciju sustava pozicioniranja unutar prostora izrađena je Android aplikacija naziva *Bluetooth Indoor Navigation*. Nakon instalacije i pokretanja izrađene Android aplikacije otvara se pogled (engl. *View*) početnog zaslona aplikacije koje se sastoji od *TextView*-a na kojem je prikazan naslov aplikacije te dva *CardView*-a. Također, unutar svakog *CardView*-a nalazi se *TextView* s tekстом *Current Location* te *Location History*. Korisničko sučelje početnog zaslona

aplikacije uređuje se u XML datoteci naziva *activity_home.xml*. Na slici 3.13 nalazi se kod korišten za kreiranje *CardView*-a i *TextView*-a *CurrentLocation*.

```
<androidx.cardview.widget.CardView
    android:id="@+id/currentLocationCard"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="30dp"
    android:layout_marginEnd="16dp"
    android:layout_marginRight="16dp"
    app:cardCornerRadius="4dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/appName">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="15dp"
        android:gravity="center"
        android:text="@string/current_location"
        android:textColor="@color/colorBlack"
        android:textSize="18sp" />

</androidx.cardview.widget.CardView>
```

Sl.3.13. Prikaz koda za *Cardview* i *TextView CurrentLocation*

Klikom na *CardView Current Location* otvara se novi pogled za prikaz trenutne lokacije korisnika, dok se klikom na *Cardview Location History* otvara pogled za prikaz svih prošlih lokacija korisnika. Na Slici 3.14 nalazi se kod za prebacivanje na pogled *CurrentLocation* ili *LocationHistory*.

```
public void onClick(View view) {
    switch (view.getId()) {
        case R.id.currentLocationCard:
            goToCurrentLocationActivity();
            break;
        case R.id.locationsHistoryCard:
            goToLocationsHistoryActivity();
            break;
    }
}
```

Sl.3.14. Prikaz koda za prebacivanje na pogled *CurrentLocation* ili *LocationHistory*

3.3.2. Android Manifest

Sve Android aplikacije moraju sadržavati XML datoteku naziva *AndroidManifest.xml* smještenu u direktoriju *manifests*. U Android Manifest datoteci su definirani osnovni elementi Android aplikacije poput paketa, aktivnosti, početne aktivnosti, dozvola te ostale komponente aplikacije. Budući da Android aplikacija ne može automatski pristupiti određenim značajkama pametnog telefona (npr. pristup internetu, pristup kameri, pristup mikrofonu itd.), u Android Manifest datoteci uključuju se dozvole za ispravan rad aplikacije [26]. Za korištenje Bluetooth bežične tehnologije potrebno je uključiti sljedeće četiri dozvole: *BLUETOOTH*, *BLUETOOTH_ADMIN*, *ACCES_COARSE_LOCATION* te *ACCES_FINE_LOCATION*. *BLUETOOTH* dozvola je neophodna za komunikaciju putem Bluetooth mreže. *BLUETOOTH_ADMIN* dozvola omogućuje pretraživanje i povezivanje s drugim Bluetooth uređajima te mijenjanje Bluetooth postavki. *ACCES_COARSE_LOCATION* se koristi za dohvaćanje približne lokacije, dok se *ACCES_FINE_LOCATION* koristi za dohvaćanje precizne lokacije korisnika i potrebno ju je potvrditi pri pokretanju Android aplikacije [27].

Aktivnost je osnovni element sastava svake Android aplikacije. Svaka aktivnost mora biti definirana u manifest datoteci gdje joj se pridodaje određena oznaka te akcija [28]. Izrađena Android aplikacija se sastoji od tri aktivnosti naziva *HomeActivity*, *CurrentLocationActivity* i *LocationsHistoryActivity*. Aktivnost *HomeActivity* je definirana kao glavna aktivnost te se prva pokreće prilikom pokretanja aplikacije, što je vidljivo na slici 3.15.

```
<activity android:name=".ui.HomeActivity"
    android:screenOrientation="portrait">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

Sl. 3.15. Prikaz koda za *HomeActivity* aktivnost

3.3.3. Izračun lokacije pametnog telefona

Kod za prikaz trenutne lokacije korisnika pisan je u Java programskom jeziku te se nalazi u datoteci *CurrentLocationActivity.java*. Prvi korak za uspješnu realizaciju ovog zadatka je pretraga Bluetooth uređaja u blizini. Paket *android.bluetooth* pruža klase za upravljanje Bluetooth funkcionalnostima poput pretraživanja dostupnih uređaja, povezivanja s dostupnim uređajima te prijenosa podataka između dostupnih uređaja. Konstante i javne metode potrebne za realizaciju ovog projekta nalaze se unutar klasa *BluetoothAdapter* i *BluetoothDevice*. Klasa *BluetoothAdapter* predstavlja lokalni Bluetooth adapter koji omogućuje izvedbu osnovnih Bluetooth zadataka poput započinjanja pretraživanja uređaja u blizini ili prikaza uparenih Bluetooth uređaja.[29]. Klasa *BluetoothDevice* predstavlja Bluetooth uređaj u blizini te omogućuje povezivanje s Bluetooth uređajem u blizini. Također, pomoću klase *BluetoothDevice* može se doći do informacija o Bluetooth uređaju poput imena i MAC adrese [30]. Pojednostavljeno, lokalni Bluetooth adapter predstavlja pokretnu stanicu, odnosno pametni telefon, dok klasa *BluetoothDevice* služi za dohvaćanje informacija o Bluetooth modulima, odnosno baznim stanicama. Opis konstanti i javnih metoda korištenih u realizaciji ovog projekta nalaze se u tablicama 3.2 i 3.3.

Tab. 3.2. Opis korištenih konstanti [29, 30]

Konstanta	Klasa	Opis
<i>ACTION_STATE_CHANGED</i>	<i>BluetoothAdapter</i>	Stanje lokalnog Bluetooth adaptera je promijenjeno
<i>ACTION_FOUND</i>	<i>BluetoothDevice</i>	Otkriven je uređaj u blizini
<i>ACTION_DISCOVERY_STARTED</i>	<i>BluetoothAdapter</i>	Lokalni Bluetooth adapter započinje pretragu uređaja u blizini
<i>ACTION_DISCOVERY_FINISHED</i>	<i>BluetoothAdapter</i>	Lokalni Bluetooth adapter završava pretragu uređaja u blizini
<i>EXTRA_RSSI</i>	<i>BluetoothDevice</i>	Sadrži RSSI vrijednost uređaja u blizini

Tab. 3.3. Opis korištenih javnih metoda [29,30]

Javna metoda	Klasa	Opis
<i>getDefaultAdapter()</i>	<i>BluetoothAdapter</i>	Inicijalizacija lokalnog Bluetooth adaptera
<i>getName()</i>	<i>BluetoothDevice</i>	Dohvaćanje imena uređaja u blizini
<i>startDiscovery()</i>	<i>BluetoothAdapter</i>	Početak pretrage uređaja u blizini

Koristeći konstante i javne metode klasa *BluetoothDevice* i *BluetoothAdapter* obavljena je pretraga Bluetooth uređaja (baznih stanica) u blizini te dohvaćanje njihova naziva i vrijednosti RSSI. Vrijednost RSSI je važna za računanje udaljenosti između pokretne i bazne stanice. Za računanje udaljenosti po izrazu (2-2), osim vrijednosti RSSI, potrebna je i vrijednost A koja predstavlja RSSI kada su bazna i pokretna stanica udaljene jedan metar. Budući da je vrijednost A drugačija za svaki Bluetooth senzor, potrebno je postaviti svaki Bluetooth senzor na udaljenost 1 metar od pametnog telefona te izmjeriti vrijednost A za svaki senzor. Za svaki senzor vrijednost A je izmjerena deset puta te su rezultati mjerenja prikazani u tablici 3.4.

Tab. 3.4. Prikaz rezultata mjerenja vrijednosti A

Naziv senzora	A_1 (Db/m)	A_2 (Db/m)	A_3 (Db/m)	A_4 (Db/m)	A_5 (Db/m)	A_6 (Db/m)	A_7 (Db/m)	A_8 (Db/m)	A_9 (Db/m)	A_{10} (Db/m)
BTM1	-80	-80	-77	-77	-74	-92	-79	-77	-77	-78
BTM2	-86	-74	-74	-86	-83	-74	-70	-74	-74	-71
BTM3	-87	-88	-92	-84	-87	-87	-87	-88	-87	-87
BTM4	-79	-80	-82	-80	-80	-81	-77	-81	-82	-80

U tablici 3.4 je vidljivo kako vrijednost varijable A nije konstantna. Za svaki senzor je odabrana ona vrijednost koja se pojavljuje najviše puta. Vrijednost varijable A senzora BTM1 je -77, vrijednost varijable A senzora BTM2 je -74, vrijednost varijable A senzora BTM3 je -87 i vrijednost varijable A senzora BTM4 je -80 što je vidljivo u tablici 3.4 gdje su vrijednosti koje se pojavljuju najviše puta označene zelenom bojom. Na slici 3.16 prikazan je kod za pretraživanje Bluetooth senzora u blizini i računanje udaljenosti tih istih senzora po izrazu (2-2). Za dohvaćanje imena Bluetooth senzora koristi se javna metoda *getName()* klase *BluetoothDevice*, dok se za dohvaćanje vrijednosti RSSI koristi konstanta *EXTRA_RSSI* klase *BluetoothDevice*. Nakon dohvaćanja imena i vrijednosti RSSI, slijedi računanje udaljenosti do određenog Bluetooth senzora. Udaljenosti se računaju prema izrazu (2-2) i spremaju u vrijednosti *distance1*, *distance2*, *distance3* te *distance4*.


```

if (BluetoothDevice.ACTION_FOUND.equals(action)) {
    BluetoothDevice device =
intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
    String name = device.getName();
    int rssi = intent.getShortExtra(BluetoothDevice.EXTRA_RSSI,
Short.MIN_VALUE);

    if (name.equals("BTM1")) {
        double a1 = (-77 - (rssi));
        double b1 = a1 / 20;
        distance1=Math.pow(10, b1);
    }
    if (name.equals("BTM2")) {
        double a2 = (-74 - (rssi));
        double b2 = a2 / 20;
        distance2= Math.pow(10, b2);
    }

    if (name.equals("BTM3")) {
        double a3 = (-87 - (rssi));
        double b3 = a3 / 20;
        distance3= Math.pow(10, b3);
    }

    if (name.equals("BTM4")) {
        double a4 = (-80 - (rssi));
        double b4 = a4 / 20;
        distance4= Math.pow(10, b4);
    }
}

```

SI.3.16. Prikaz koda za pretragu Bluetooth uređaja u blizini i računanje udaljenosti

Nakon računanja udaljenosti do svakog Bluetooth senzora, slijedi odabir tri najbliža Bluetooth senzora. Udaljenost svakog senzora se uspoređuje s udaljenošću ostala tri senzora te se dolazi do rezultata koja tri senzora se nalaze najbliže pametnom telefonu. Konačne (x,y) koordinate pametnog telefona računaju se po izrazima (2-11) i (2-12) gdje (x1,y1), (x2,y2) te (x3,y3) predstavljaju koordinate tri najbliža Bluetooth senzora, a vrijednosti r1, r2 i r3 predstavljaju udaljenost pametnog telefona do ta tri senzora. Na slici 3.17 je prikazan kod za slučaj da su senzori BTM1, BTM2 i BTM3 najbliži pametnom telefonu. Isti kod se primjenjuje za ostala tri slučaja.

```

if (distance1 <= distance4 && distance2 <= distance4 && distance3 <= distance4) {
    //pozicija BTM1
    x1 = 2;
    y1 = 0;
    //pozicija BTM2
    x2 = 0;
    y2 = 3.7;
    //pozicija BTM3
    x3 = 2;
    y3 = 7.4;
    r1 = distance1;
    r2 = distance2;
    r3 = distance3;
}

```

Sl. 3.17. Prikaz koda za uspoređivanje udaljenosti Bluetooth senzora

Nakon izračuna koja tri senzora su najbliža pametnom telefonu, dostupni su svi podaci za izračun lokacije pametnog telefona koristeći izraze (2-11) i (2-12). Na slici 3.18 prikazan je kod za izračun lokacije pametnog telefona metodom trilateracije. Izrazi (2-11) i (2-12) su podijeljeni na dijelove A, B, C, D, E i F zbog jednostavnijeg zapisivanja.

```

double A = 2 * x2 - 2 * x1;
double B = 2 * y2 - 2 * y1;
double C = Math.pow(r1, 2) - Math.pow(r2, 2) - Math.pow(x1, 2) + Math.pow(x2, 2)
- Math.pow(y1, 2) + Math.pow(y2, 2);
double D = 2 * x3 - 2 * x2;
double E = 2 * y3 - 2 * y2;
double F = Math.pow(r2, 2) - Math.pow(r3, 2) - Math.pow(x2, 2) + Math.pow(x3, 2)
- Math.pow(y2, 2) + Math.pow(y3, 2);
double x = (C * E - F * B) / (E * A - B * D);
double y = (C * D - A * F) / (B * D - A * E);

```

Sl. 3.18. Prikaz koda za izračun lokacije pametnog telefona

3.3.4. Prikaz prethodnih lokacija

Svaki rezultat pretrage Bluetooth uređaja u blizini sprema se u bazu podataka. Kod izrade ove aplikacije koristi se Realm baza podataka. Realm je sustav upravljanja objektima baze podataka otvorenog koda koji se prvotno koristi u Android i iOS operacijskim sustavima [31]. Realm je NoSQL baza podataka te pohranjuje native aplikacijske objekte, što pojednostavljuje postupak osvježavanja podataka unutar baze. Za razliku od SQL baza podataka, Realm koristi princip lijenog dohvata podataka (engl. *lazy loading*) čime se smanjuje opterećenje priručne podatkovne memorije jer se podaci iz baze podataka dohvaćaju samo u onom trenutku u kojem su potrebni

[32]. Na slici 3.19 je prikazan kod za spremanje dobivenih koordinata trenutne lokacije pametnog telefona u bazu podataka. Kod sa slike 3.19 se nastavlja na kod sa slike 3.18.

```
newDevice.setPosition(x);
newDevice.setPosition(y);
addDeviceInDatabase(newDevice);
```

Sl.3.19. Prikaz koda za spremanje vrijednosti x i y u bazu podataka

Za prikaz prethodnih lokacija koristi se funkcija koja učitava sve vrijednosti koje su spremljene u bazu podataka. Na slici 3.20 je prikazan kod funkcije za učitavanje svih prethodnih lokacija unutar datoteke *LocationsHistory.java*.

```
private void loadLocations() {
    realm.executeTransaction(new Realm.Transaction() {
        @Override
        public void execute(Realm realm) {
            RealmResults<Device> results = realm.where(Device.class).findAll();
            for (Device device : results) {
                showDeviceLocation(device);
            }
        }
    });
}
```

Sl.3.20. Funkcija za učitavanje prethodnih lokacija iz baze podataka

3.3.5. Prikaz lokacije pametnog telefona

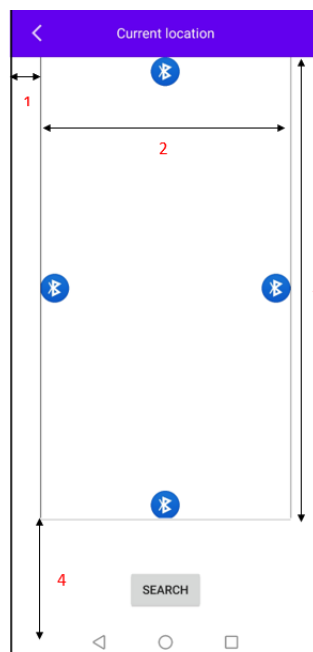
Nakon očitavanja lokacije pametnog telefona potrebno je pravilno prikazati rezultat na zaslonu pametnog telefona. Za prikaz lokacije povučene su četiri linije koje predstavljaju zidove prostorije širine četiri metra i dužine 7.4 metara za koju je sustav izrađen. Za početak, potrebno je dohvatiti rezoluciju zaslona pametnog telefona kako bi prikaz lokacije bio prilagođen svakom zaslonu pametnog telefona, neovisno o rezoluciji. Za dohvaćanje rezolucije zaslona pametnog telefona koriste se polja *heightPixels* i *widthPixels* klase *DisplayMetrics*. Na slici 3.21 prikazan je kod za dohvaćanje rezolucije zaslona te prilagodbu zaslona za postavljanje linija koje predstavljaju zidove prostorije. Nakon dohvaćanja rezolucije zaslona i spremanja u varijable *deviceHeight* i *deviceWidth*, potrebno je ostaviti 400 piksela na dnu zaslona za gumb (engl. *button*) Search.

Varijable *roomWidth* i *roomHeight* predstavljaju broj piksela za širinu i dužinu prostorije, dok varijabla *oneM* predstavlja broj piksela za duljinu jednog metra.

```
private void showBluetoothSensors() {
    DisplayMetrics metrics = new DisplayMetrics();
    getWindowManager().getDefaultDisplay().getMetrics(metrics);
    int deviceHeight = metrics.heightPixels;
    int deviceWidth = metrics.widthPixels;
    int deviceWidth1 = (deviceHeight - 400) / 2;
    int oneM = deviceWidth1 / 4;
    int roomWidth = deviceWidth1;
    int roomHeight = (int) (7.4 * oneM);
    int marginLeft = (deviceWidth - deviceWidth1) / 2;
}
```

Sl. 3.21. Prikaz koda za dohvaćanje rezolucije pametnog telefona i prilagodbu zaslona za postavljanje linija

Na slici 3.22 prikazan je zaslon pametnog telefona s iscrtanim linijama koje predstavljaju zidove prostorije. Strelica 1 predstavlja marginu čija je vrijednost u pikselima jednaka polovini razlike varijable *deviceWidth* koja predstavlja širinu zaslona pametnog telefona i varijable *roomWidth* koja predstavlja širinu prostorije. Strelica 2 predstavlja širinu prostorije, odnosno varijablu *roomWidth*, dok strelica 3 predstavlja dužinu prostorije, odnosno varijablu *roomHeight*. Strelica 4 predstavlja pomak dužine za 400 piksela zbog Search gumba.



Sl. 3.22. Prikaz zaslona pametnog telefona s iscrtanim linijama

Na slici 3.23 prikazan je kod za povlačenje jedne linije koja predstavlja jedan zid prostorije, dok je na slici 3.24 prikazan kod za postavljanje slike koja predstavlja jedan Bluetooth senzor. Na slici 3.25 je prikazan koda za prikaz lokacije pametnog telefona.

```
iv = new ImageView(this);
iv.setBackgroundColor(Color.BLACK);
params = new RelativeLayout.LayoutParams(roomWidth, 2);
params.leftMargin = marginLeft;
params.topMargin = roomHeight;
container.addView(iv, params);
```

Sl. 3.23. Prikaz kod a za iscrtavanje linije koje predstavlja jedan zid prostorije

```
iv = new ImageView(this);
iv.setImageDrawable(ContextCompat.getDrawable(this, R.drawable.bluetooth));
params = new RelativeLayout.LayoutParams(100, 100);
params.leftMargin = marginLeft + roomWidth - 100;
params.topMargin = (roomHeight / 2) - 50;
container.addView(iv, params);
```

Sl. 3.24. Prikaz koda za raspored slike koja predstavlja Bluetooth senzor

```
devicePosition = new ImageView(this);
devicePosition.setImageDrawable(ContextCompat.getDrawable(this,
R.drawable.Locationmark));
params = new RelativeLayout.LayoutParams(100, 100);
params.leftMargin = (int) (marginLeft + (device.getPosition() * oneM));
params.topMargin = (int) (roomHeight - (device.getyPosition() * oneM));
container.addView(devicePosition, params);
```

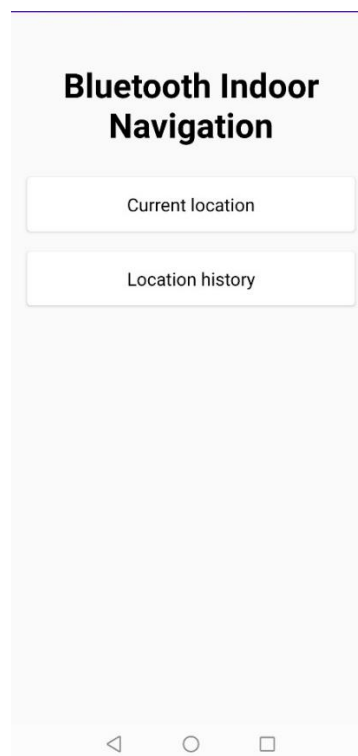
Sl. 3.25. Prikaz koda za prikaz lokacije pametnog telefona

4. PRIKAZ NAČINA RADA I ISPITIVANJE SUSTAVA S ANALIZOM REZULTATA

U ovom poglavlju prikazan je način rada aplikacije s prikazom trenutne lokacije i prethodnih lokacija. Opisano je testiranje rada aplikacije s provedenih deset mjerenja za deset unaprijed određenih lokacija. Također, opisana je analiza testiranja te primjena izrađenog sustava na analizu rukometne utakmice.

4.1. Prikaz načina rada aplikacije

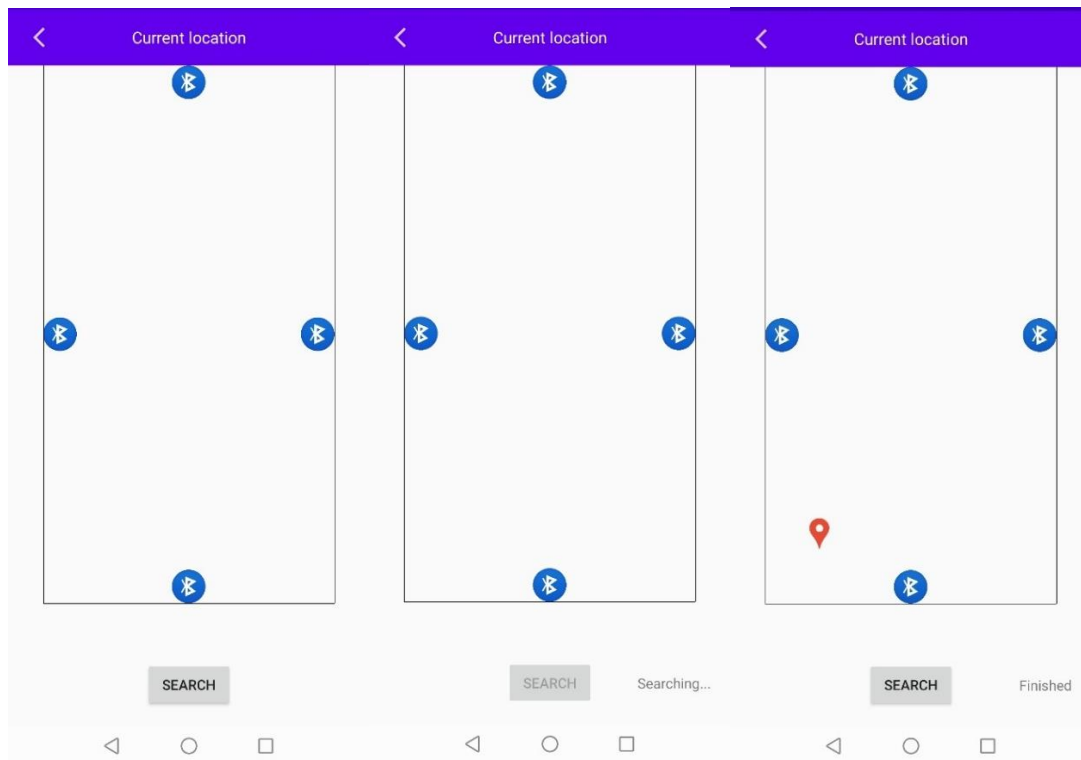
Pokretanjem izrađene aplikacije otvara se pogled početnog zaslona aplikacije gdje se može birati između prikaza trenutne lokacije (*Current Location*) te prikaza prošlih lokacija (*Location History*). Na slici 4.1 prikazan je početni zaslon aplikacije.



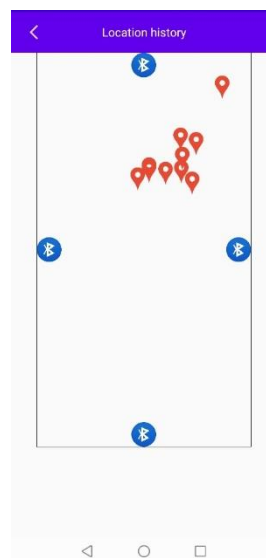
Sl. 4.1. Pogled početnog zaslona aplikacije

Klikom na Current Location otvara se pogled za pretragu i prikaz trenutne lokacije korisnika. Klikom na gumb Search započinje pretraga uređaja u blizini te se *TextView* na kojem je prikazan status mijenja u 'Searching'. Također, klik na gumb Search je onemogućen dok traje pretraga. Završetkom pretraživanja pojavljuje se oznaka lokacije (engl. *location mark*) na mjestu trenutne lokacije pametnog telefona. Također, završetkom pretraživanja se *TextView* za prikaz statusa mijenja u 'Finished' te je ponovni klik na gumb Search omogućen. Ponovnim klikom na gumb

Search briše se oznaka lokacije te počinje pretraga nove lokacije pametnog telefona. Na slici 4.2 prikazan je postupak pretraživanja lokacije pametnog telefona. Klikom na Location History otvara se pogled koji učitava sva prethodna mjerenja iz baze podataka te ih prikazuje na zaslonu, što je vidljivo na slici 4.3.



SI. 4.2. Prikaz postupka pretraživanja lokacije



SI.4.3. Prikaz prethodnih lokacija

4.2. Testiranje preciznosti određivanja lokacije

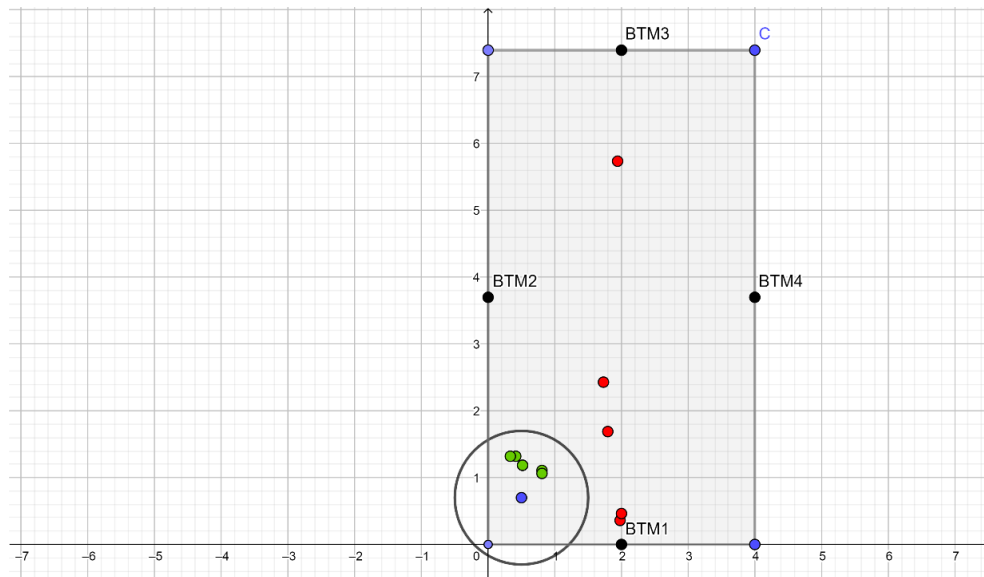
Za testiranje preciznosti određivanja lokacije pametnog telefona unaprijed je određeno deset lokacija s pripadajućim x i y koordinatama. Kako bi se mogla odrediti preciznost određivanja lokacije, za svaku lokaciju je odrađeno deset mjerenja. Rezultat mjerenja je prihvatljiv ako se nalazi unutar radijusa od jednog metra od unaprijed određene lokacije. Rezultati mjerenja prikazani su u tablicama gdje zelena boja predstavlja da je rezultat mjerenja prihvatljiv, a crvena boja predstavlja da se rezultat mjerenja ne nalazi unutar prostorije u kojoj je aplikacija testirana. Također, rezultati mjerenja prikazani su u dvodimenzionalnom koordinatnom sustavu te na zaslonu pametnog telefona. Kod prikaza mjerenja u dvodimenzionalnom koordinatnom sustavu plavom bojom su označene koordinate unaprijed određene lokacije, zelenom bojom koordinate prihvatljivih mjerenja, a crvenom bojom mjerenja koja se ne nalaze unutar radijusa od jedan metar od zadanih koordinata. Također, koordinate prihvatljivih mjerenja nalaze se unutar kružnice polumjera jedan s ishodištem u točki unaprijed određene lokacije. Za testiranje aplikacije koristi se pametni telefon Huawei P Smart 2019. Prva unaprijed određena lokacija za testiranje je (0.5, 0.7). Rezultati mjerenja prikazani su u tablici 4.1.

Tab. 4.1. Rezultati mjerenja za prvu unaprijed određenu lokaciju

ZADANE KOORDINATE	(0.5, 0.7)
1. MJERENJE	(1.795, 1.69)
2. MJERENJE	(1.942, 5.738)
3. MJERENJE	(1.73, 2.43)
4. MJERENJE	(0.414, 1.32)
5. MJERENJE	(1.9793, 0.36)
6. MJERENJE	(0.5165, 1.1858)
7. MJERENJE	(2, 0.4629)
8. MJERENJE	(0.3328, 1.32)
9. MJERENJE	(0.807, 1.107)
10. MJERENJE	(0.8075, 1.0634)

U tablici 4.1 je vidljivo kako se svih deset mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije, dok se pet mjerenja nalaze u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom

koordinatnom sustavu vidljiv je na slici 4.4, a na slici 4.5 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



Sl. 4.4. Prikaz rezultata mjerenja prve unaprijed određene lokacije u koordinatnom sustavu



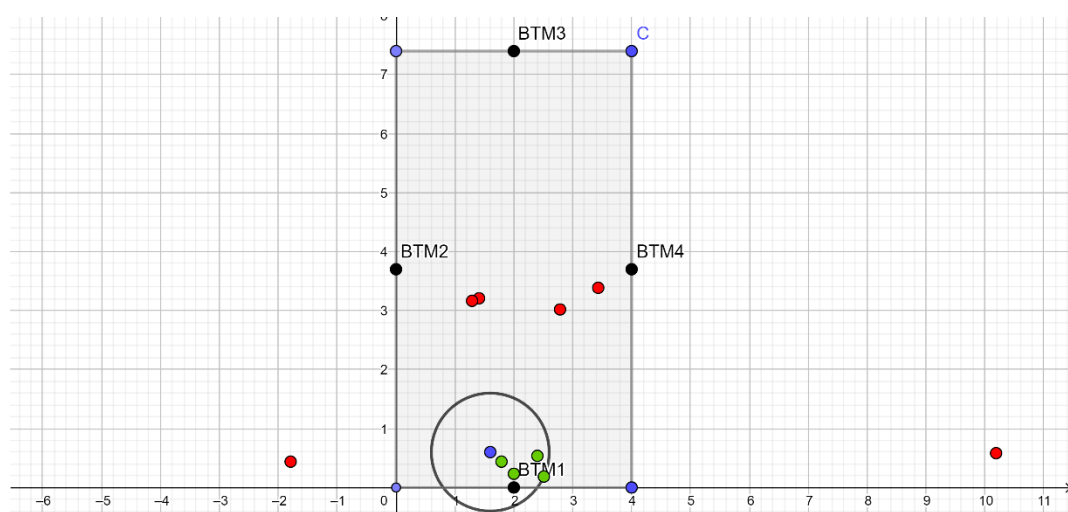
Sl. 4.5. Prikaz rezultata mjerenja za prvu unaprijed određenu lokaciju na zaslonu pametnog telefona

Druga unaprijed određena lokacija za testiranje je (1.6, 0.6). Rezultati mjerenja prikazani su u tablici 4.2.

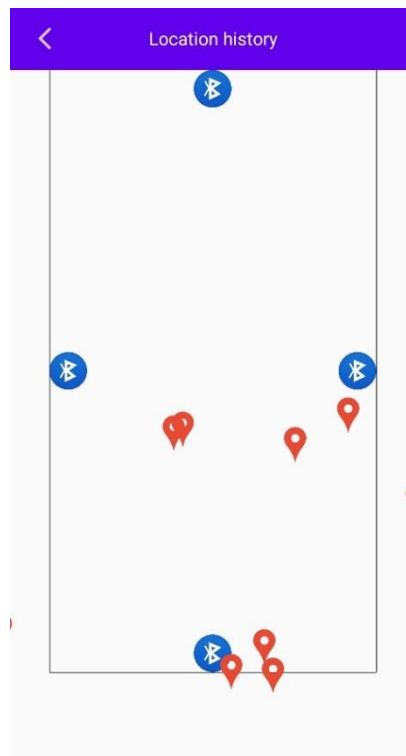
Tab. 4.2. Rezultati mjerenja za drugu unaprijed određenu lokaciju

ZADANE KOORDINATE	(1.6, 0.6)
1. MJERENJE	(1.409, 3.208)
2. MJERENJE	(2.51, 0.186)
3. MJERENJE	(1.29, 3.165)
4. MJERENJE	(1.79, 0.4384)
5. MJERENJE	(-1.79, 0.4384)
6. MJERENJE	(2.785, 3.019)
7. MJERENJE	(3.433, 3.386)
8. MJERENJE	(10.189, 0.5825)
9. MJERENJE	(2.0, 0.2322)
10. MJERENJE	(2.4, 0.536769)

U tablici 4.2 vidljivo je kako se osam od deset mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije. Mjerenje koje je dalo najbližu lokaciju zadanoj je četvrto mjerenje u kojem su rezultat koordinate (1.79, 0.4384). Četiri mjerenja su dala rezultat u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.6, a na slici 4.7 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



Sl. 4.6. Prikaz rezultata mjerenja druge unaprijed određene lokacije u koordinatnom sustavu



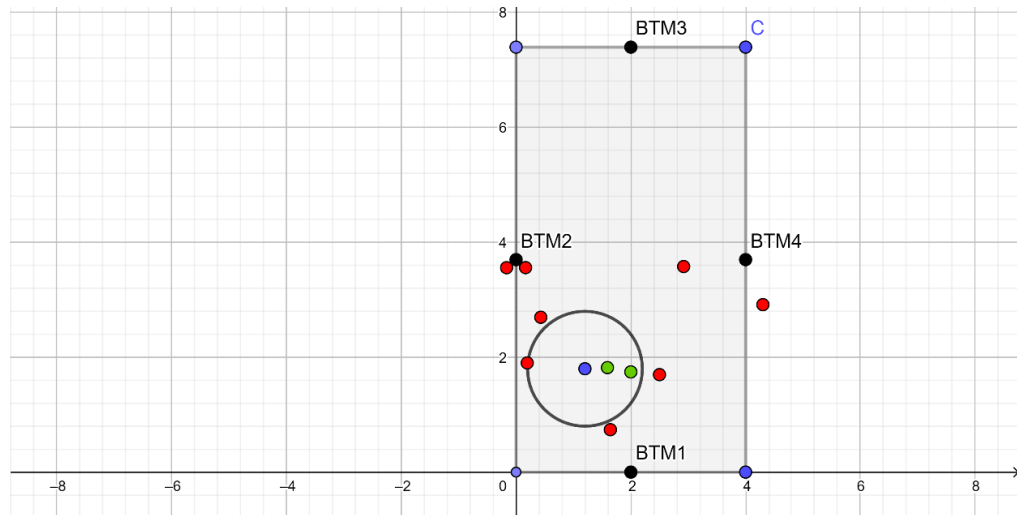
Sl. 4.7. Prikaz rezultata mjerenja za drugu unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate treće unaprijed određene lokacije su (1.2, 1.8). Rezultati mjerenja prikazani su u tablici 4.3.

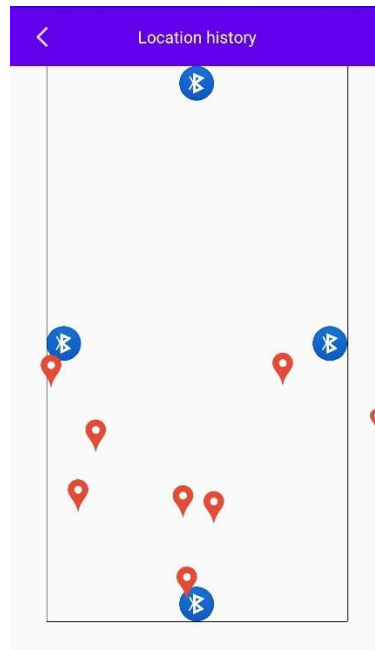
Tab. 4.3. Rezultati mjerenja za treću unaprijed određenu lokaciju

ZADANE KOORDINATE	(1.2, 1.8)
1. MJERENJE	(2.499, 1.698)
2. MJERENJE	(0.1945, 1.901)
3. MJERENJE	(1.5924, 1.82)
4. MJERENJE	(2.0, 1.745)
5. MJERENJE	(0.1654, 3.561)
6. MJERENJE	(-0.165, 3.56)
7. MJERENJE	(2.92, 3.58)
8. MJERENJE	(4.301, 2.916)
9. MJERENJE	(0.43064, 2.6966)
10. MJERENJE	(1.643, 0.739)

U tablici 4.3 vidljivo je kako se osam od deset mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije. Mjerenje koje je dalo najbližu lokaciju zadanoj je treće. mjerenje u kojem su rezultat koordinate (1.59, 1.82). Samo dva mjerenja su dala rezultat u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.8 gdje se dva prihvatljiva mjerenja nalaze unutar kružnice, a na slici 4.9 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



Sl. 4.8. Prikaz rezultata mjerenja treće unaprijed određene lokacije u koordinatnom sustavu



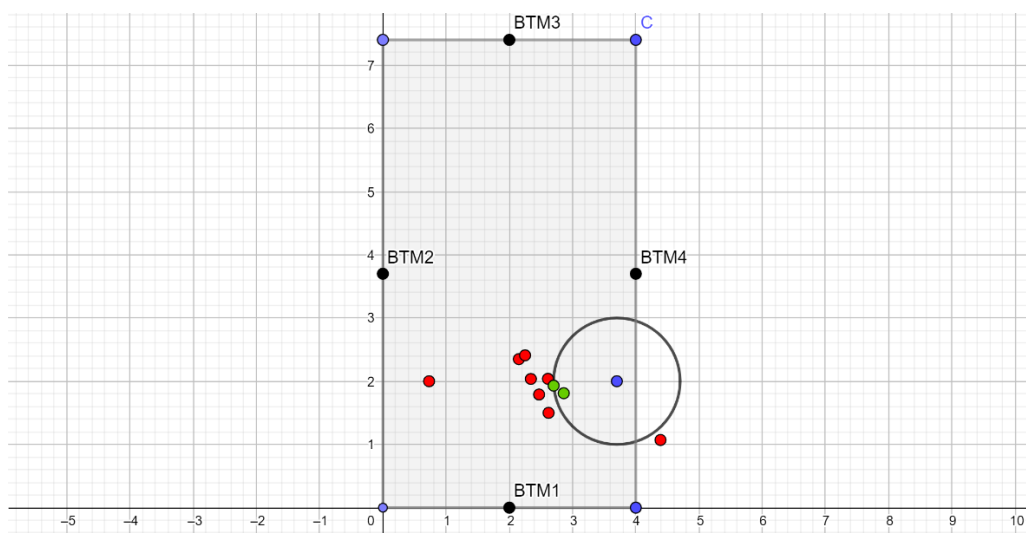
Sl. 4.9. Prikaz rezultata mjerenja za treću unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate četvrte unaprijed određene lokacije su (3.7, 2). Rezultati mjerenja prikazani su u tablici 4.3.

Tab. 4.4. Rezultati mjerenja za četvrtu unaprijed određenu lokaciju

ZADANE KOORDINATE	(3.7, 2)
1. MJERENJE	(1.47, -0.42)
2. MJERENJE	(2.61, 2.038)
3. MJERENJE	(-0.73, 2.0)
4. MJERENJE	(2.34, 0.036)
5. MJERENJE	(2.7, 1.93)
6. MJERENJE	(4.39, 1.07)
7. MJERENJE	(2.86, 1.81)
8. MJERENJE	(1.15, 0.35)
9. MJERENJE	(2.248, 0.35)
10. MJERENJE	(2.62, 1.5)

U tablici 4.4 vidljivo je kako se sedam od deset mjerenja nalaze unutar prostorijske u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije. Samo dva mjerenja su dala rezultat u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.10, a na slici 4.11 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



Sl. 4.10. Prikaz rezultata mjerenja četvrte unaprijed određene lokacije u koordinatnom sustavu



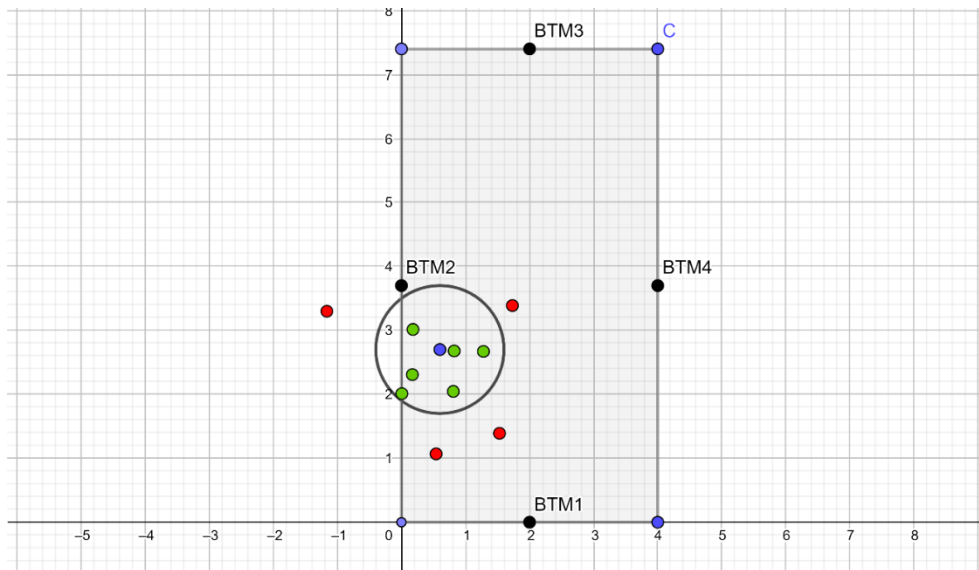
Sl. 4.11. Prikaz rezultata mjerenja za četvrtu unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate pete unaprijed određene lokacije su (0,6, 2,7). Rezultati mjerenja prikazani su u tablici 4.5.

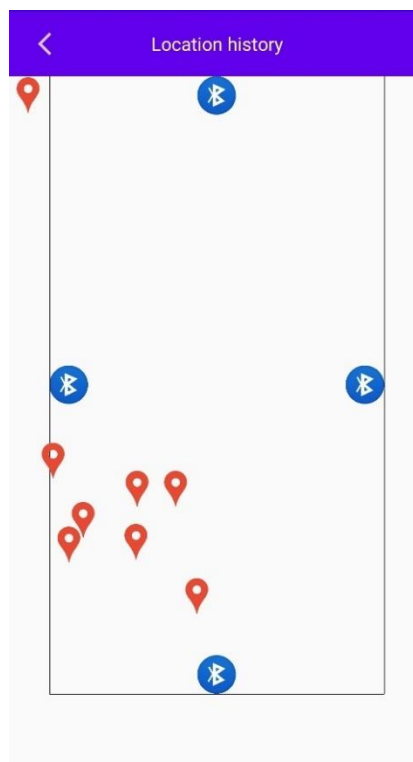
Tab. 4.5. Rezultati mjerenja za petu unaprijed određenu lokaciju

ZADANE KOORDINATE	(0,6, 2,7)
1. MJERENJE	(1.53, 1.39)
2. MJERENJE	(0.006, 2.011)
3. MJERENJE	(-1.1641, 3.299)
4. MJERENJE	(0.823, 2.678)
5. MJERENJE	(0.541, 1.069)
6. MJERENJE	(0.81, 2.045)
7. MJERENJE	(1.732, 3.388)
8. MJERENJE	(0.18, 3.014)
9. MJERENJE	(1.28, 2.67)
10. MJERENJE	(0.17, 2.308)

U tablici 4.5 vidljivo je kako se devet od deset mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije. Čak šest mjerenja su dala rezultat u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.12, a na slici 4.13 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



SI. 4.12. Prikaz rezultata mjerenja pete unaprijed određene lokacije u koordinatnom sustavu



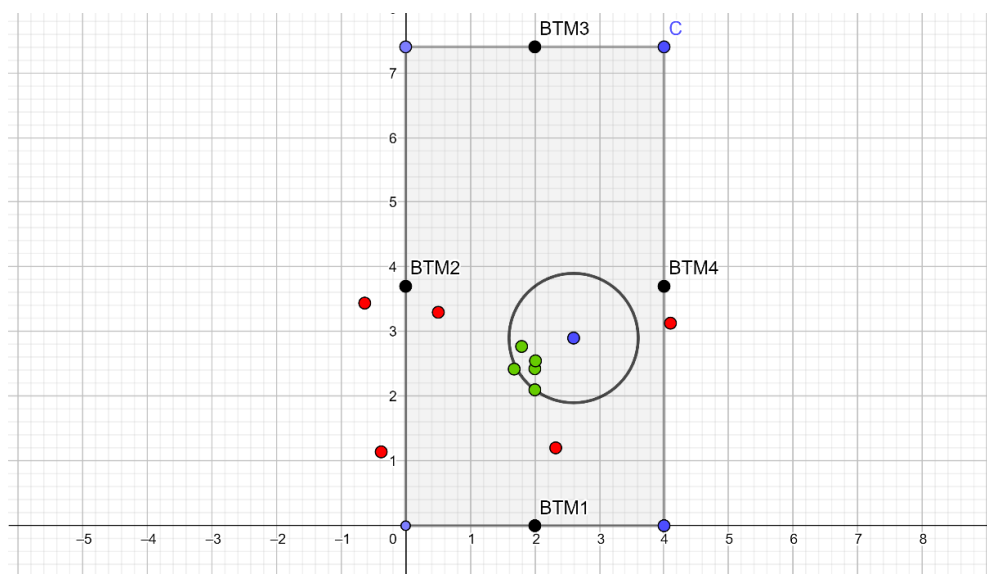
SI. 4.13. Prikaz rezultata mjerenja za petu unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate šeste unaprijed određene lokacije su (2.6, 2.9). Rezultati mjerenja prikazani su u tablici 4.6.

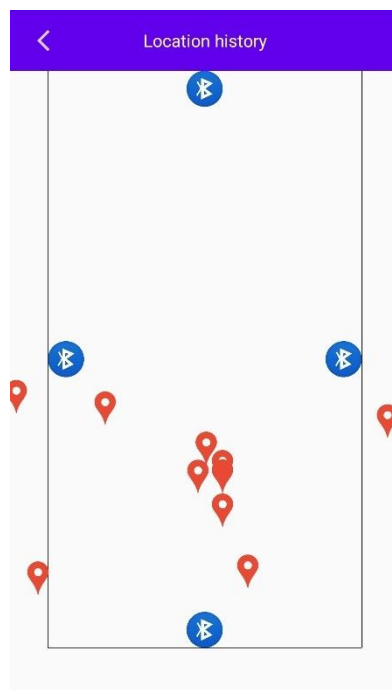
Tab. 4.6. Rezultati mjerenja za šestu unaprijed određenu lokaciju

ZADANE KOORDINATE	(2.6, 2.9)
1. MJERENJE	(1.795, 2.77)
2. MJERENJE	(-0.35, 1.119)
3. MJERENJE	(2.323, 1.203)
4. MJERENJE	(2.0, 1.987)
5. MJERENJE	(4.1, 3.13)
6. MJERENJE	(-0.6339, 3.44)
7. MJERENJE	(0.505, 3.298)
8. MJERENJE	(2.0, 2.425)
9. MJERENJE	(1.68, 2.42)
10. MJERENJE	(2.01, 2.548)

U tablici 4.6 je vidljivo kako se sedam mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije, dok se pet mjerenja nalaze u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.14, a na slici 4.15 je vidljiv prikaz mjerenja na zaslonu pametnog telefona.



Sl. 4.14. Prikaz rezultata mjerenja šeste unaprijed određene lokacije u koordinatnom sustavu



Sl. 4.15. Prikaz rezultata mjerenja za šestu unaprijed određenu lokaciju na zaslonu pametnog telefona

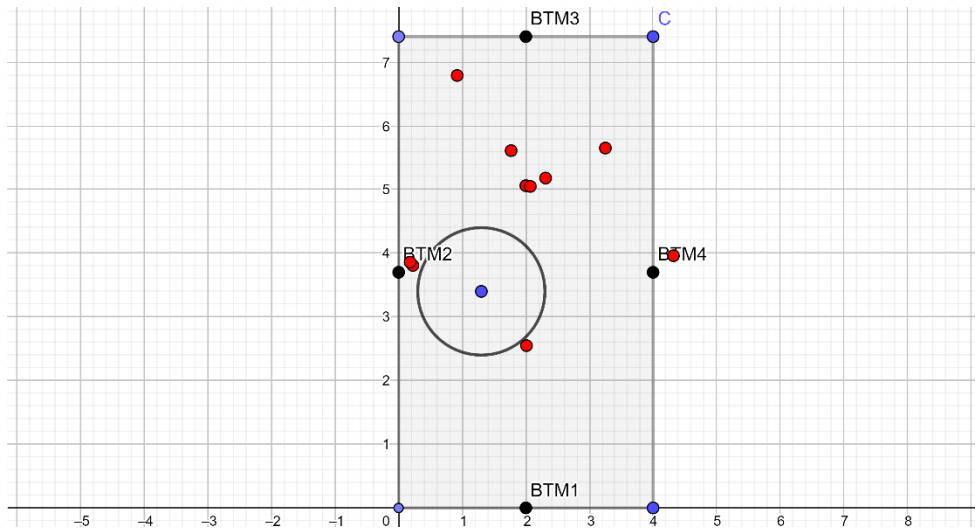
Koordinate sedme unaprijed određene lokacije su (1.3, 3.4). Rezultati mjerenja prikazani su u tablici 4.7.

Tab. 4.7. Rezultati mjerenja za sedmu unaprijed određenu lokaciju

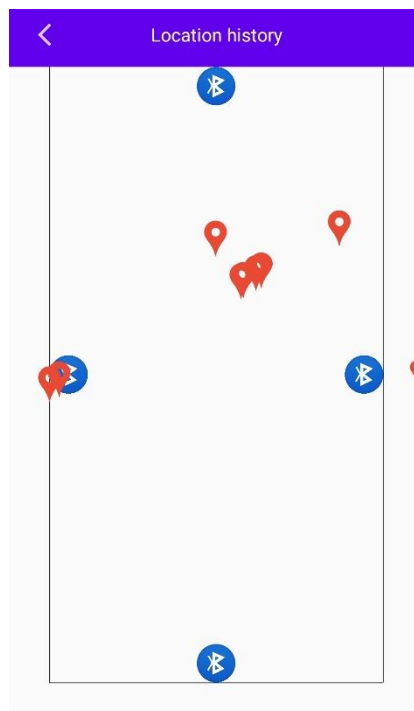
ZADANE KOORDINATE	(1.3, 3.4)
1. MJERENJE	(1.768, 5.61)
2. MJERENJE	(4.32, 3.96)
3. MJERENJE	(0.92, 6.79)
4. MJERENJE	(2, 5.06)
5. MJERENJE	(2.31, 5.179)
6. MJERENJE	(3.25, 5.65)
7. MJERENJE	(2.07, 5.048)
8. MJERENJE	(0.224, 3.807)
9. MJERENJE	(0.18, 3.86)
10. MJERENJE	(2.099, 5.06)

U tablici 4.7 je vidljivo kako se devet mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije. Ovo je

prvo mjerenje u kojem se niti jedan rezultat mjerenja ne nalazi u radijusu jedan metar od zadane lokacije. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.16, a na slici 4.17 je vidljiv prikaz mjerenja na zaslonu pametnog telefona



SI. 4.16. Prikaz rezultata mjerenja sedme unaprijed određene lokacije u koordinatnom sustavu



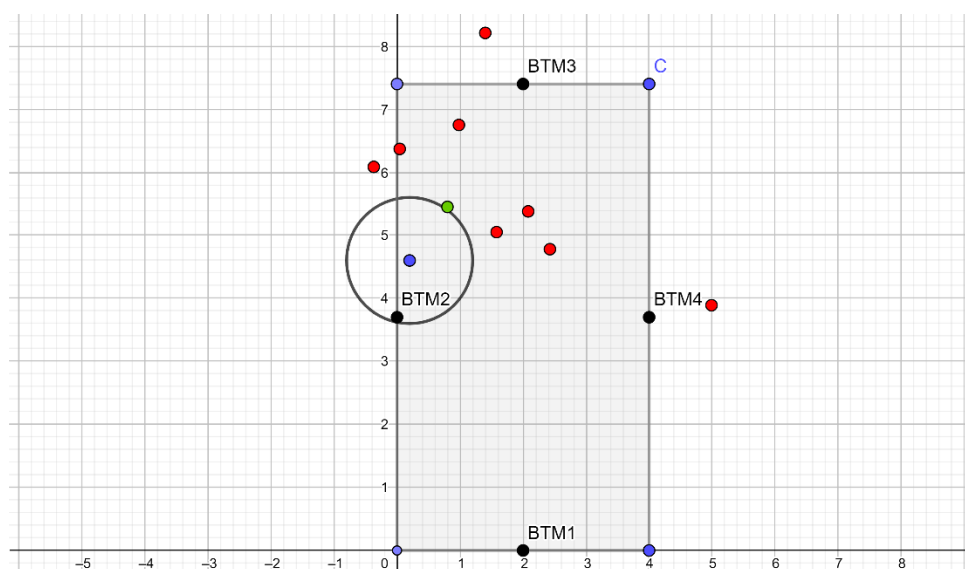
SI. 4.17. Prikaz rezultata mjerenja za sedmu unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate osme unaprijed određene lokacije su (0.2, 4.6). Rezultati mjerenja prikazani su u tablici 4.8.

Tab. 4.8. Rezultati mjerenja za osmu unaprijed određenu lokaciju

ZADANE KOORDINATE	(0.2, 4.6)
1. MJERENJE	(1.58, 5.05)
2. MJERENJE	(NaN, NaN)
3. MJERENJE	(2.426, 4.779)
4. MJERENJE	(2.081, 5.379)
5. MJERENJE	(4.99, 3.89)
6. MJERENJE	(0.0427, 6.37)
7. MJERENJE	(0.9824, 6.75)
8. MJERENJE	(1.4, 8.21)
9. MJERENJE	(-0.369, 6.085)
10. MJERENJE	(0.948, 5.45)

U tablici 4.8 je vidljivo kako se šest mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je kako nijedno mjerenje nije dalo točan rezultat lokacije, dok je samo jedno mjerenje prihvatljivo. Ovo je prvo mjerenje u kojemu su rezultat koordinate (NaN, NaN). NaN (skraćeno od engl. *Not a Number*) se u matematici tumači kao nedefinirana vrijednost. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.18, a na slici 4.19 je vidljiv prikaz mjerenja na zaslonu pametnog telefona



Sl. 4.18. Prikaz rezultata mjerenja osme unaprijed određene lokacije u koordinatnom sustavu



Sl. 4.19. Prikaz rezultata mjerenja za osmu unaprijed određenu lokaciju na zaslonu pametnog telefona

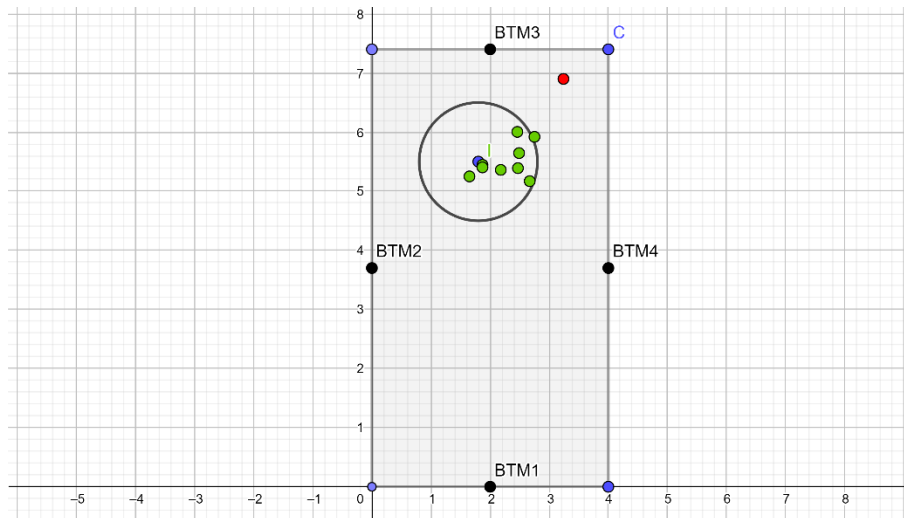
Koordinate devete unaprijed određene lokacije su (1.8, 5.5). Rezultati mjerenja prikazani su u tablici 4.9.

Tab. 4.9. Rezultati mjerenja za devetu unaprijed određenu lokaciju

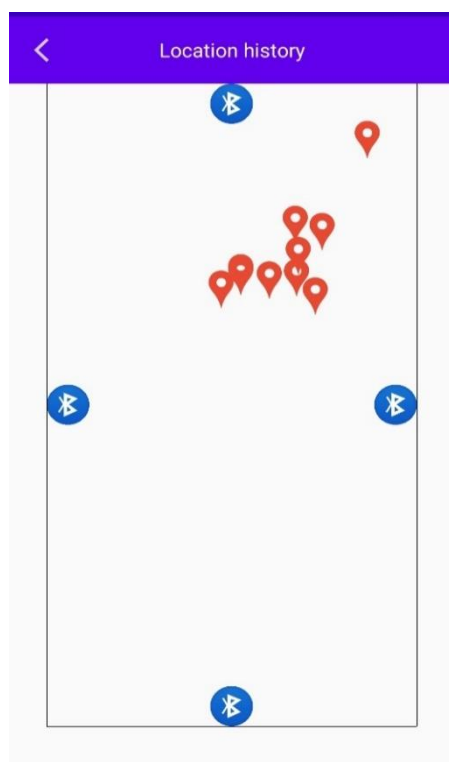
ZADANE KOORDINATE	(1.8, 5.5)
1. MJERENJE	(3.24, 6.9)
2. MJERENJE	(2.67, 5.17)
3. MJERENJE	(2.46, 6.004)
4. MJERENJE	(1.87, 5.4029)
5. MJERENJE	(2.18, 5.36)
6. MJERENJE	(2.49, 5.645)
7. MJERENJE	(1.65, 5.25)
8. MJERENJE	(2.75, 5.92)
9. MJERENJE	(2.47, 5.39)
10. MJERENJE	(1.87, 5.446)

U tablici 4.9 je vidljivo kako se sva mjerenja nalaze unutar prostorije u kojoj je aplikacija testirana. Također, vidljivo je da je čak devet mjerenja dalo prihvatljiv rezultat, dok su četvrto i deseto

mjerenje dali približan rezultat zadanoj vrijednosti. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.20, a na slici 4.21 je vidljiv prikaz mjerenja na zaslonu pametnog telefona



Sl. 4.20. Prikaz rezultata mjerenja devete unaprijed određene lokacije u koordinatnom sustavu



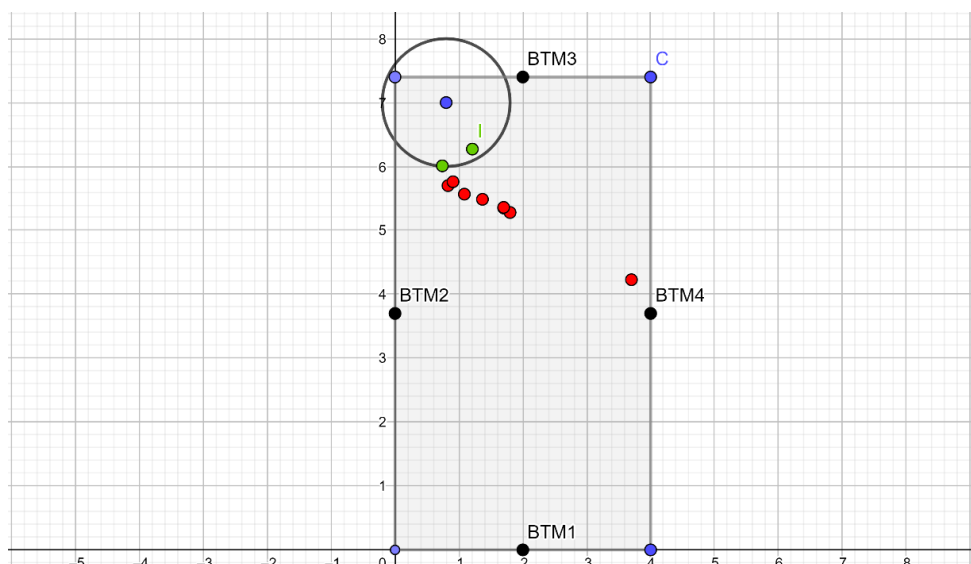
Sl. 4.21. Prikaz rezultata mjerenja za devetu unaprijed određenu lokaciju na zaslonu pametnog telefona

Koordinate desete unaprijed određene lokacije su (0,8, 7). Rezultati mjerenja prikazani su u tablici 4.10.

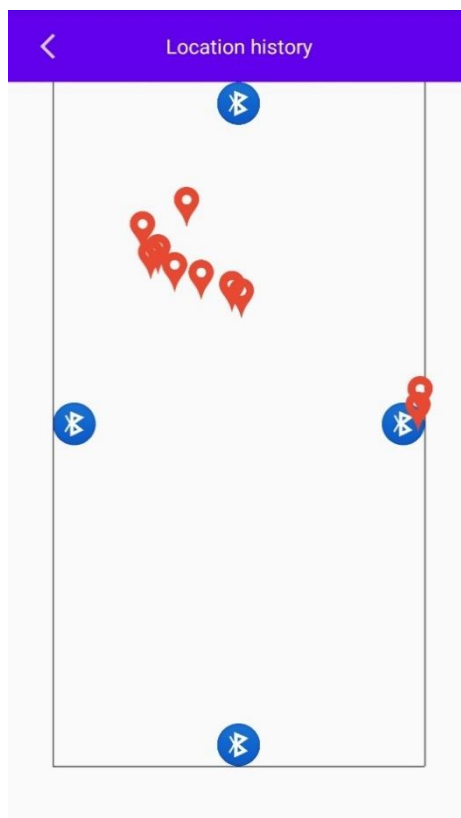
Tab. 4.10. Rezultati mjerenja za desetu unaprijed određenu lokaciju

ZADANE KOORDINATE	(0,8, 7)
1. MJERENJE	(1.085, 5.567)
2. MJERENJE	(1.7, 5.35)
3. MJERENJE	(0.74, 6.01)
4. MJERENJE	(1.8, 5.28)
5. MJERENJE	(3.7, 4.228)
6. MJERENJE	(1.368, 5.486)
7. MJERENJE	(0.8288, 5.7)
8. MJERENJE	(0.907, 5.761)
9. MJERENJE	(1.7, 5.36)
10. MJERENJE	(1.21, 6.272)

U tablici 4.10 je vidljivo kako se sva mjerenja nalaze unutar prostoriije u kojoj je aplikacija testirana. Također, vidljivo je su dva mjerenja dala prihvatljiv rezultat. Prikaz mjerenja u dvodimenzionalnom koordinatnom sustavu vidljiv je na slici 4.22, a na slici 4.23 je vidljiv prikaz mjerenja na zaslonu pametnog telefona



Sl. 4.22. Prikaz rezultata mjerenja desete unaprijed određene lokacije u koordinatnom sustavu



Sl. 4.23. Prikaz rezultata mjerenja za desetu unaprijed određenu lokaciju na zaslonu pametnog telefona

4.3. Analiza rezultata za provedena mjerenja

Realizirani sustav za pozicioniranje unutar prostora testiran je deset puta za svakih deset unaprijed određenih lokacija unutar prostorije za koju je sustav izrađen, što znači da je sveukupno provedeno 100 mjerenja. Od sveukupno 100 mjerenja, 36 mjerenja je dalo prihvatljiv rezultat, odnosno rezultat unutar radijusa od jedan metar od unaprijed određene lokacije. Najviše prihvatljivih rezultata dalo je testiranje devete unaprijed određene lokacije s devet prihvatljivih mjerenja. Testiranje šeste unaprijed određene lokacije dalo je šest prihvatljivih mjerenja, dok su testiranja prve i šeste unaprijed određene lokacije dala po pet prihvatljivih mjerenja. Slijedi testiranje druge unaprijed određene lokacije s četiri prihvatljiva mjerenja. Testiranja treće, četvrte i desete unaprijed određene lokacije dala su rezultat od dva prihvatljiva mjerenja, dok je rezultat testiranja osme unaprijed određene lokacije jedno prihvatljivo mjerenje. Jedino testiranje bez ijednog prihvatljivog mjerenja je testiranje sedme unaprijed određene lokacije. Također, jedno mjerenje unutar testiranja osme unaprijed određene lokacije dalo je nedefiniran rezultat. Budući da je samo 36 mjerenja dalo prihvatljiv rezultat, izrađeni sustav nije pouzdan za određivanje

precizne lokacije unutar prostora. U tablici 4.11 prikazan je broj prihvatljivih rezultata po mjerenjima.

Tab. 4.11. Prikaz prihvatljivih mjerenja za svaku unaprijed određenu lokaciju

Unaprijed određena lokacija	1.	2.	3.	4.	5.	6.	7	8.	9.	10.
Broj prihvatljivih rezultata	5	4	2	2	6	5	0	1	9	2

Od sveukupno 100 mjerenja, čak 84 mjerenja su pokazala rezultat koji se nalazi unutar prostorije u kojoj je sustav testiran. Sva mjerenja unutar testiranja prve, devete i desete unaprijed određene lokacije nalaze se unutar prostorije u kojoj se sustav testiran, dok su testiranja pete i šeste unaprijed određene lokacije dala po jedan rezultat koji se ne nalazi unutar prostorije. Testiranja druge i treće unaprijed određene lokacije dala su po dva rezultata koja se ne nalaze unutar prostorije, a testiranja četvrte i šeste unaprijed određene lokacije dala su po tri rezultata koja se ne nalaze unutar prostorije. Testiranje osme unaprijed određene lokacije dalo je četiri mjerenja koja se ne nalaze unutar prostorije. Analizom ovih rezultata dolazi se do zaključka da iako sustav nije pouzdan za određivanje precizne lokacije unutar prostora, pouzdan je za određivanje unutar koje prostorije se korisnik nalazi. U tablici 4.12 prikazan je broj rezultata koji se nalaze unutar prostorije u kojoj je sustav testiran.

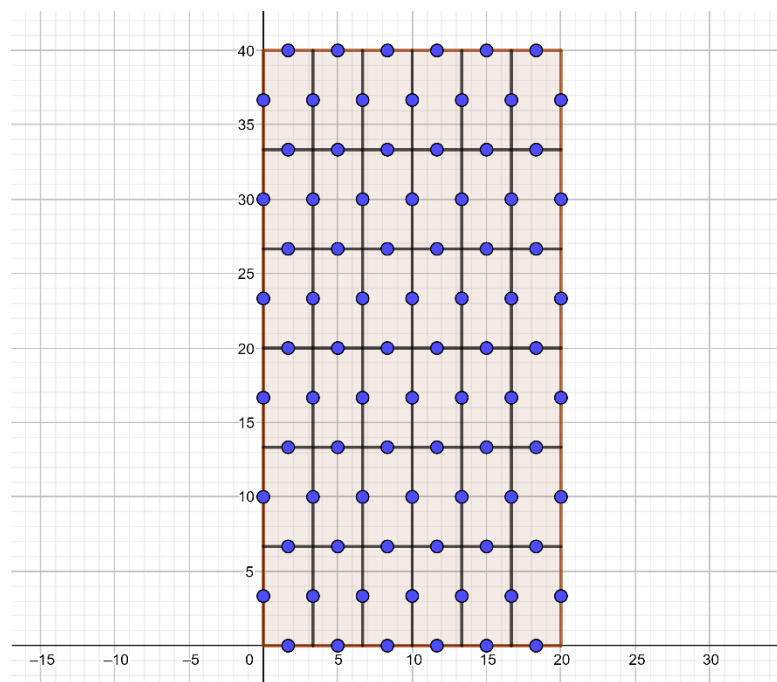
Tab. 4.12. Prikaz rezultata koji se nalaze unutar prostorije u kojoj je sustav testiran

Unaprijed određena lokacija	1.	2.	3.	4.	5.	6.	7	8.	9.	10.
Broj rezultata unutar prostorije	10	8	8	7	9	7	9	6	10	10

4.4. Primjena sustava u analizi rukometne utakmice

Jedan od načina primjene realiziranog sustava za pozicioniranje unutar prostora je analiza rukometne utakmice tako da se rukometni teren podijeli na više dijelova te da se prati u kojem dijelu terena se određeni igrač najviše kretao. Dužina rukometnog terena je 40 metara, a širina 20 metara. Realizirani sustav za pozicioniranje unutar prostora je prilagođen za prostoriju dužine 7.4 metara te širine 4 metra. Omjer dužine rukometnog terena i dužine prostorije za koju je sustav izrađen je 5.4054, što bi značilo da za pokrivanje dužine rukometnog terena treba 5.4054 izrađenih sustava. Budući da rezultat omjera nije cijeli broj, gleda se prvi sljedeći cijeli broj, odnosno broj šest. Ako se gleda manji broj od rezultata omjera, odnosno broj pet, postavljeni senzori bi bili više

udaljeni jedni od drugih nego u sustavu u kojemu su testirani. Veća je vjerojatnost da će rezultat biti točan ako su senzori gušće raspoređeni. Ako je šest izrađenih sustava dovoljno da pokrije dužinu terena, znači da se jedan sustav postavlja na svakih 6.6667 metara dužine rukometnog terena te na svaka 3.3333 metra širine rukometnog terena. To znači da je za pokrivanje rukometnog terena potrebno 36 sustava koji se sastoje od 84 bazne stanice postavljene na sredinu dužine i širine svakog sustava. Na slici 4.24 je u dvodimenzionalnom koordinatnom sustavu prikazan rukometni teren podijeljen na 36 sustava za pozicioniranje unutar prostora. Točke plave boje predstavljaju Bluetooth senzore potrebne za određivanje lokacije igrača.



Sl. 4.24. Prikaz položaja baznih stanica na rukometnom terenu

Sustav predložen za pokrivanje rukometnog terena ne odgovara konfiguraciji testnog sustava zbog toga što bi senzori morali biti instalirani na samom terenu. Budući da je ispitivanje rezultata iz ovog rada provedeno na visini od 120 cm, takva mreža senzora ne bi bila prikladna za kontaktni sport poput rukometa zbog fizičkih barijera. Mogućnost postavljanja ovakve konfiguracije senzora na višim visinama mogla bi donijeti dodatnu nepreciznost u mjerenju zbog toga što se uređaj ne nalazi u horizontalnom polju unutar senzorske mreže. Bez detaljnijeg ispitivanja svaki teorijski zaključak može biti pogrešan, kao i pretpostavka da bi ovakav sustav predložen u radu mogao vrlo precizno locirati igrače unutar rukometnog terena i pratiti njihovu analizu, iako matematički može dati vrlo precizne rezultate. Postavljanje mreže s manje Bluetooth senzora ili postavljanje senzora samo uz rub rukometnog terena bi vrlo vjerojatno rezultiralo još

većom nepreciznošću zbog ograničenog doseg Bluetooth signala. Time bi preklapanja signala bila manja, zbog čega bi izračun lokacije metodom trilateracije vrlo vjerojatno bio nemoguć.

Na temelju testiranja pozicioniranja svakog uređaja i dobivenih rezultata, nije moguće ni približno točno odrediti poziciju određenog igrača te bi svaka daljnja analiza dovela do značajnijih odstupanja i sve netočnijih rezultata. Svaki uređaj radi kao samostalni detektor Bluetooth senzora koji su u doseg te simultana detekcija više igrača nema primjenu u samom uređaju, odnosno aplikaciji. Podaci se mogu sinkronizirati putem podatkovne veze s bazom podataka te tako agregirati u novi skup zajedničkih mjerenja svih sudionika. Zbog nepreciznosti ove konfiguracije Bluetooth uređaja i mobilnog telefona, bilo koja daljnja zasebna ili skupna analiza može služiti samo na razini raspoznavanja prisutnosti ili detekcije, no ne i pouzdanog pozicioniranja. Također, u razmatranje se može uzeti i obrnut postupak detekcije gdje bi svaki Bluetooth senzor, odnosno bazna stanica, bio primatelj signala te očitavao RSSI svake pokretne stanice. Međutim, za obradu takvih podataka u stvarnom vremenu potrebna je značajna procesorska snaga koju ovi jednostavni uređaji nisu u mogućnosti isporučiti. Ograničenje realiziranog sustava za pozicioniranje unutar prostora je i sam razred korištenih komponenti. Radi se o najjeftinijim sklopovima koji omogućuju Bluetooth komunikaciju zbog čega je njihova pouzdanost i snaga upitne preciznosti te je moguća veća tolerancija na odstupanja nego kod sklopova višeg razreda. Moguće je da bi se preciznije pozicioniranje moglo postići kvalitetnijim i skupljim sklopovima.

5. ZAKLJUČAK

Zadatak ovog diplomskog rada bio je izraditi sustav pozicioniranja unutar prostora koristeći bežičnu tehnologiju Bluetooth. Za realizaciju zadatka izrađena je Android aplikacija, koja pomoću prikupljenih podataka o jačini primljenog signala (RSSI) koji odašilju bazne stanice (Bluetooth moduli) izračunava lokaciju pametnog telefona. U izrađenoj aplikaciji jačina primljenog signala se pretvara u metre, zatim se odabiru tri najbliže bazne stanice te se konačni položaj pametnog telefona računa metodom trilateracije. Također, primljeni podaci se spremaju u bazu podataka te ih je moguće prikazati na zaslonu pametnog telefona.

Glavna problematika izrade ovog sustava je to što vrijednost jačine primljenog signala nije konstantna. Testiranjem sustava i analizom rezultata došlo je do zaključka kako sustav nije pouzdan za precizno određivanje lokacije unutar prostora. Sustav nije pouzdan jer jačina primljenog signala varira, iako se svako mjerenje provodi u istim uvjetima. Također, analizom rezultata testiranja preciznosti lokacije došlo je do zaključka da je sustav pouzdan za određivanje unutar koje prostorije se korisnik nalazi. U praksi, ovaj sustav se može koristiti u trgovačkim centrima za lociranje unutar koje trgovine se korisnik nalazi, u poslovnim zgradama za lociranje unutar kojeg ureda se korisnik nalazi, u zračnim lukama za lociranje na kojem terminalu se korisnik nalazi, u bolnicama za lociranje unutar koje sobe se korisnik nalazi, itd. Osim provjere prostorije u kojoj se korisnik nalazi, sustav se može koristiti i za praćenje broja korisnika unutar određene prostorije. Iako je ovaj sustav prvotno zamišljen za korištenje u analizi rukometne utakmice, zaključak je da sustav nije prikladan za korištenje u bilo kojem kontaktnom sportu zbog fizičkih barijera, odnosno visine na kojoj su Bluetooth senzori postavljeni. Također, postoji vjerojatnost da je sustav neprecizan jer su u realizaciji korišteni najjeftiniji sklopovi koji omogućuju Bluetooth komunikaciju te da bi određivanje lokacije bilo preciznije ako bi se koristili kvalitetniji i skuplji sklopovi.

LITERATURA

- [1] M. Kaluža , K. Beg, B. Vukelić, Analysis of an Indoor Positioning Systems, pregledni rad, Zbornik Veleučilišta u Rijeci, 2017.
- [2] D. Namiot, On Indoor Positioning, Internacional Journal of Open Information Technologies, ISSN: 2307-8162 vol.3, no. 3, 2015.
- [3] H. Park , J. Noh ,S. Cho, Three-dimensional Positioning Using Bluetooth Low-energy Beacons, International Journal of Distributed Sensor Networks, vol.12, listopad 2016.
- [4] V. Valjetić , Ponuda usluga temeljenih na lokaciji putem mobilnih mreža, završni rad, Zagreb, 2016., dostupno na: <https://repozitorij.fpz.unizg.hr/islandora/object/fpz:666/preview>, pristup ostvaren 18.9.2021.
- [5] M. Simić-Pejović., Principi pozicioniranja u ćelijskim radio sistemima, elektronski udžbenik, Beograd, 2016.
- [6] J. Larson, Distance Estimation and Positioning Based on Bluetooth Low Energy Technology, KTH Royal Institute of Technology, 2015.
- [7] I. Kraljević, Utvrđivanje mikrolokacije mobilnog uređaja u zatvorenom prostoru, diplomski rad, dostupno na: http://www.zemris.fer.hr/~golub/ga/studenti/2014_kraljevic/index.html , pristup ostvaren 11.9.2021.
- [8] Cell Phone Trilateration Algorithm, dostupno na: <https://www.101computing.net/cell-phone-trilateration-algorithm/>, pristup ostvaren 18.9.2021.
- [9] L. Frank, N. Nikic , Indoor navigation with Bluetooth Low Energy in crowded places, diplomski rad, Wien, 2018., dostupno na: https://hildatellioglu.com/download/Bachelorarbeit_Lukas_Frank_Natasa_Nikic.pdf , pristup ostvaren 11.9.2021.
- [10] Q. Dong, W. Dargie, Evaluation of the Reliability of RSSI for Indoor Localization, Technical University of Dresden, Germany, 2012., dostupno na: https://www.researchgate.net/publication/261447858_Evaluation_of_the_reliability_of_RSSI_for_indoor_localization , pristup ostvaren 11.9.2021.

- [11] HERE, Indoor Positioning Installation Guide, dostupno na: https://indoor.here.com/docs/Indoor_Positioning_Installation_Guide.pdf , pristup ostvaren 19.9.2021.
- [12] H. K. Almathami, M. A. Alrafiee, E. Vlahu- Gjorgievska, K. T. Win, An Analytical Approach to Using and Implementing Beacons: Opportunities and Challenges, Journal of Wireless Mobile Networks, str. 58-74, ožujak 2019.
- [13] Jeren, P. Pale, Sustavi za vođenje i praćenje procesa, dostupno na: <http://spvp.zesoi.fer.hr/predavanja%202008/BT-skripta.pdf>, pristup ostvaren 30.6.2019.
- [14] Nacionalni Cert u suradnji s LSS, Ranjivosti Bluetooth tehnologije NCERT-PUBDOC-2009-11-281, dostupno na: <https://www.cis.hr/www.edicija/LinkedDocuments/NCERT-PUBDOC-2009-11-281.pdf> , pristup ostvaren 30.6.2019.
- [15] D. Žagar, Predavanja iz kolegija Komunikacijske mreže, Osijek, 2013.
- [16] STEM Centar MŽ, Arduino 01: Uvod u Arduino programiranje, dostupno na: <https://stem.mik.hr/arduino-01-uvod-u-arduino-programiranje/>, pristup ostvaren 12.9.2021.
- [17] JavaTpoint, Arduino Syntax and Program Flow, dostupno na: <https://www.javatpoint.com/arduino-syntax-and-program-flow>, pristup ostvaren 12.9.2021.
- [18] Britannica, Android Operating System, dostupno na: <https://www.britannica.com/technology/Android-operating-system>, pristup ostvaren 20.9.2021.
- [19] S. Haberl, Razvoj aplikacija za operacijski sustav Android, diplomski rad, Zagreb, 2017., dostupno na: <http://docplayer.rs/194192023-Razvoj-aplikacija-za-operacijski-sustav-android.html>, pristup ostvaren 12.9.2021.
- [20] E-radionica, Bluetooth modul HC-06, dostupno na: <https://e-radionica.com/hr/bluetooth-modul-hc06.html>, pristup ostvaren 30.6.2019
- [21] E-radionica, Novi Croduino je tu – Croduino Basic 2, dostupno na: <https://e-radionica.com/hr/blog/2015/10/11/novi-croduino-je-tu-croduino-basic2> , pristup ostvaren 12.9.2021.
- [22] E-radionica, Croduino Basic2, dostupno na: <https://e-radionica.com/hr/croduino-basic2.html> , pristup ostvaren 12.9.2021.

- [23] E-radionica, KKM Bluetooth modul HC-06, dostupno na: <https://e-radionica.com/hr/blog/2016/04/06/bluetooth/>, pristup ostvaren 12.9.2021.
- [24] Instructables Circuits, How to Set AT Command Mode for HC-05 Bluetooth Module, dostupno na: <https://www.instructables.com/How-to-Set-AT-Command-Mode-for-HC-05-Bluetooth-Mod/>, pristup ostvaren 13.09.2021.
- [25] Techopedia, Graphical User Interface (GUI), dostupno na: <https://www.techopedia.com/definition/5435/graphical-user-interface-gui>, pristup ostvaren 13.9.2021.
- [26] S. Lukac, Android aplikacija s uputama za pružanje prve pomoći, diplomski rad, FERIT Osijek, 2016., dostupno na: <https://repositorij.etfos.hr/islandora/object/etfos%3A832/datastream/PDF/view>, pristup ostvaren 13.9.2021.
- [27] Android Developer, Manifest.permission, dostupno na: <https://developer.android.com/reference/android/Manifest.permission>, pristup ostvaren 13.9.2021.
- [28] B. Kokot, Razvoj Android aplikacije, završni rad, Sveučilište Jurja Dobrile u Puli, 2015., dostupno na: <https://repositorij.unipu.hr/islandora/object/unipu%3A534/datastream/PDF/view>, pristup ostvaren 13.9.2021.
- [29] Android Developer, Bluetooth Adapter, dostupno na: <https://developer.android.com/reference/android/bluetooth/BluetoothAdapter>, pristup ostvaren 13.9.2021.
- [30] Android Developer, Bluetooth Device, dostupno na: <https://developer.android.com/reference/android/bluetooth/BluetoothDevice>, pristup ostvaren 13.9.2021.
- [31] Database of Databases, Realm, dostupno na: <https://dbdb.io/db/realm>, pristup ostvaren 19.9.2021.
- [32] F. Floreani, Ručno određivanje lažnih preklapanja koja nastaju pri sastavljanju genoma, završni rad, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, 2017., dostupno na: <https://www.bib.irb.hr/884141>, pristup ostvaren 13.9.2021.

SAŽETAK

Glavni zadatak ovog rada bio je izrada sustava pozicioniranja unutar prostora koristeći bežičnu tehnologiju Bluetooth. Objasnjene su metode koje se koriste u sustavima pozicioniranja unutar prostora te izračun udaljenosti pomoću vrijednosti RSSI. Za realizaciju sustava korištene su bežična tehnologija Bluetooth, razvojna platforma Arduino te razvojna okolina Android. Izrađena je Android aplikacija koja pomoću prikupljenih podataka s Bluetooth senzora računa lokaciju pametnog telefona. Aplikacija može prikazivati trenutnu lokaciju pametnog telefona ili sve prethodne lokacije koje su spremljene u bazu podataka. Aplikacija je testirana na 10 unaprijed određenih lokacija te su prikazani rezultati mjerenja. Rezultati testiranja pokazuju da se izrađeni sustav može koristiti za lociranje unutar koje prostorijske se pametni telefon nalazi.

Ključne riječi: Android, Bluetooth, pozicioniranje, RSSI, udaljenost.

ABSTRACT

Development of Indoor Positioning System Using Bluetooth Technology

The main task of this final paper was development of indoor positioning system using Bluetooth technology. Methods used in indoor positioning systems and calculation of distance using RSSI value are explained. To realize this system, Bluetooth technology, Arduino development environment and Android development environment are used. An Android application that collects data from Bluetooth sensors to calculate the distance of smartphone is made. Application can show current location of smartphone or all previous locations of smartphone that are saved in the database. Application is tested on ten predefined locations and experimental results are shown. Experimental results show that developed system can be used to locate in which room the smartphone is.

Key words: Android, Bluetooth, positioning, RSSI, distance.

ŽIVOTOPIS

Ivana Puljić rođena je 8. lipnja 1995. godine u gradu Našice. Osnovnu školu upisuje 2002. godine te pohađa do 2010. godine u Osnovnoj školi August Harambašić Donji Miholjac. Nakon završene osnovne škole, 2010. godine upisuje Opću gimnaziju u Srednjoj školi Donji Miholjac. Zbog odličnog uspjeha tijekom četiri godine srednjoškolskog obrazovanja, 2014. godine dobiva izravan upis na Elektrotehnički fakultet u Osijeku (kasnije preimenovan u Fakultet elektrotehnike, računarstva i informacijskih tehnologija) gdje iste godine upisuje Preddiplomski studij računarstva. 2017. godine završava Preddiplomski studij računarstva te upisuje diplomski studij Informacijske i podatkovne znanosti na istom fakultetu.

PRILOZI

- Word dokument rada
- PDF dokument rada
- Programski kod aplikacije