SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

**Tim AIR1802**

mRanger

**PROJEKT IZ KOLEGIJA „Analiza i razvoj programa“**

Varaždin, 2018.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

**Fabijan Josip Kraljić**

**Jakov Kristović**

**Karlo Pavleka**

**Kristijan Perković**

**Kristijan Žebčević**

**GitHub repozitorij:** [**github.com/krizebcev/mRanger**](https://github.com/krizebcev/mRanger)

Trello: <https://trello.com/mranger2>

mRanger

**PROJEKT IZ KOLEGIJA „Analiza i razvoj programa“**

Mentor:

Dr. sc. Boris Tomaš

Doc.dr.sc Zlatko Stapić

Varaždin, listopad 2018.

Sadržaj

[1. Uvod 1](#_Toc535590624)

[2. Popis funkcionalnosti 2](#_Toc535590625)

[3. Detaljni opis funkcionalnosti i dijagrami 3](#_Toc535590626)

[3.1. Funkcionalnost povezivanje s bluetoothom 3](#_Toc535590627)

[3.2. Funkcionalnost daljinskog upravljanja robotom 9](#_Toc535590628)

[*3.3. Funkcionalnost bilježenja temperature zraka u bazu podataka* 11](#_Toc535590629)

[4. Korišteni alati i tehnologija 12](#_Toc535590630)

[5. Dizajn arhitekture 13](#_Toc535590631)

[6. Navigacijski dijagram 14](#_Toc535590632)

[7. Baza podataka 15](#_Toc535590633)

[8. Dijagram klasa 15](#_Toc535590634)

# Uvod

Tehnička dokumentacija sa svojim opisima detaljno prikazuje funkcionalnosti android mobilne aplikacije za daljinsko upravljanje mBot:Ranger robota. Ova dokumentacija opisuje arhitekturu sustava pomoću koje se lakše može razumijeti koje su sve komponente obuhvaćene projektom. Također sadrži opće opise funkcionalnosti te njihove detaljnje opise prikazane pomoću dijagrama aktivnosti. Uvidom u dijagrame aktivnosti programeri mogu lako isprogramirati aktivnosti vezane uz pojedinu funkcionalnost jer svojim prikazom definiraju točno koja aktivnost se treba izvršiti prije koje. Osim do sada navedenih djelova tehničke dokumentacije, ona sadržava i popis resursa i alata koji se koriste u projektu. Također sadrži i ERA dijagram baze podataka kako bi se znalo koje točno podatke se treba proslijediti bazi podataka. Uz sve navedno u tehničkoj dokumentaciji se nalazi i dijagram klasa koji prikazuje veze između klasa, potrebne varijable i metode kako bi se izradila navedena aplikacija.

# Popis funkcionalnosti

Ova aplikacija ima 5 funkcionalnosti koje omogućuju pravilan rad aplikacije.

* Povezivanje na mBot Ranger putem Bluetooth-a - potrebno je omogućiti povezivanje na mBot Ranger putem Bluetooth-a kako bi se uopće sam robot mogao povezati s aplikacijom.
* Daljinsko upravljanje robotom - potrebno je napraviti funkcionalnost daljinskog upravljanja robotom gdje se putem korisničkog sučelja u obliku daljinskog upravljača zadaju upravljačke komande samom robotu.
* Mogućnost podešavanja brzine kretanja – ova funkcionalnost omogućava korisniku aplikacije promjenu brzine kretanja robota.
* *Bilježenje temperature zraka u bazu podataka – putem senzora za temperaturu robot će moći zabilježiti temperaturu koju zatim šalje u bazu podataka te ju dodatno obrađuje na serverskoj strani. Nakon čega se obrađeni podaci prikazuju na web stranici.*

Glavna funkcionalnost upravljanja robotom prikazana je na dijagramu aktivnosti prikazanom na slici 1. Korisnik odabire jednu od ponuđenih aktivnosti kretanje unaprijed, kretanje unatrag, skretanje lijevo ili skretanje desno. Nakon jedne od navedenih aktivnosti aplikacija putem bluetootha šalje signal na mBot Ranger koji se pokreće u zadanom smjeru. U slučaju da je odabrana aktivnost kretanje unaprijed mBot Rander izvršava navedenu kretnju te očitava vrijednosti na senzoru koji detektira udaljenost te podatke prosljeđuje aplikaciji. U slučaju da je detektirana prepreka ispred mBot Rangera aplikacija zacrveni zaslon kako bi korisnik imao predoćenje da može doći do oštećenja uređaja.

Napomena: *naznačeni dio će se implementirati u sklopu kolegija Logičko programiranje koji također sudjeluje u programu STEM revolucija u zajednici. (Kristović, Kraljić, Pavleka)*

# Detaljni opis funkcionalnosti i dijagrami

## 3.1. Funkcionalnost povezivanje s bluetoothom

Ova funkcionalnost omogućava mobilnom uređaju da se putem aplikacije poveže s Mbot:Rangerom bluetooth vezom. Za ostvarivanje veze potrebne su sljedeće klase: MainActivity.java, Bluetooth.java, ConnectBT.java, Controls.java, DeviceListAdapter.java. U nastavku se nalaze kodovi navedenih klasa. Kako biste vidjeli kod klasa u potpunosti potrebno je napraviti dvoklik na crni okvir.



Klasa 1. MainActivity.java

Na ugrađenom dokumentu ispod dostupan je prikaz klase Bluetooth. To je klasa koja služi za uključivanje i isključivanje bluetootha kao i za otkrivanje uređaja te stvaranje konekcije.



Klasa 2. Bluetooth.java

Zatim dostupan je prikaz klase ConnectBT isto klikom na dolje prikazani ugrađeni dokument. Ova klasa služi za ostvarivanje i upravljanje bluetooth konekcijom nad uređajem.



Klasa 3. ConnectBT.java

Jedna od vjerojatno najvažnijih klasa za ovaj projekt je klasa Controls dostupna dolje u ugrađenom dokumentu. Ovo je klasa koja služi za upravljanje robotom, odnosno sve kontrole za upravljanje samim robotom su implementirane koristeći ovu klasu.

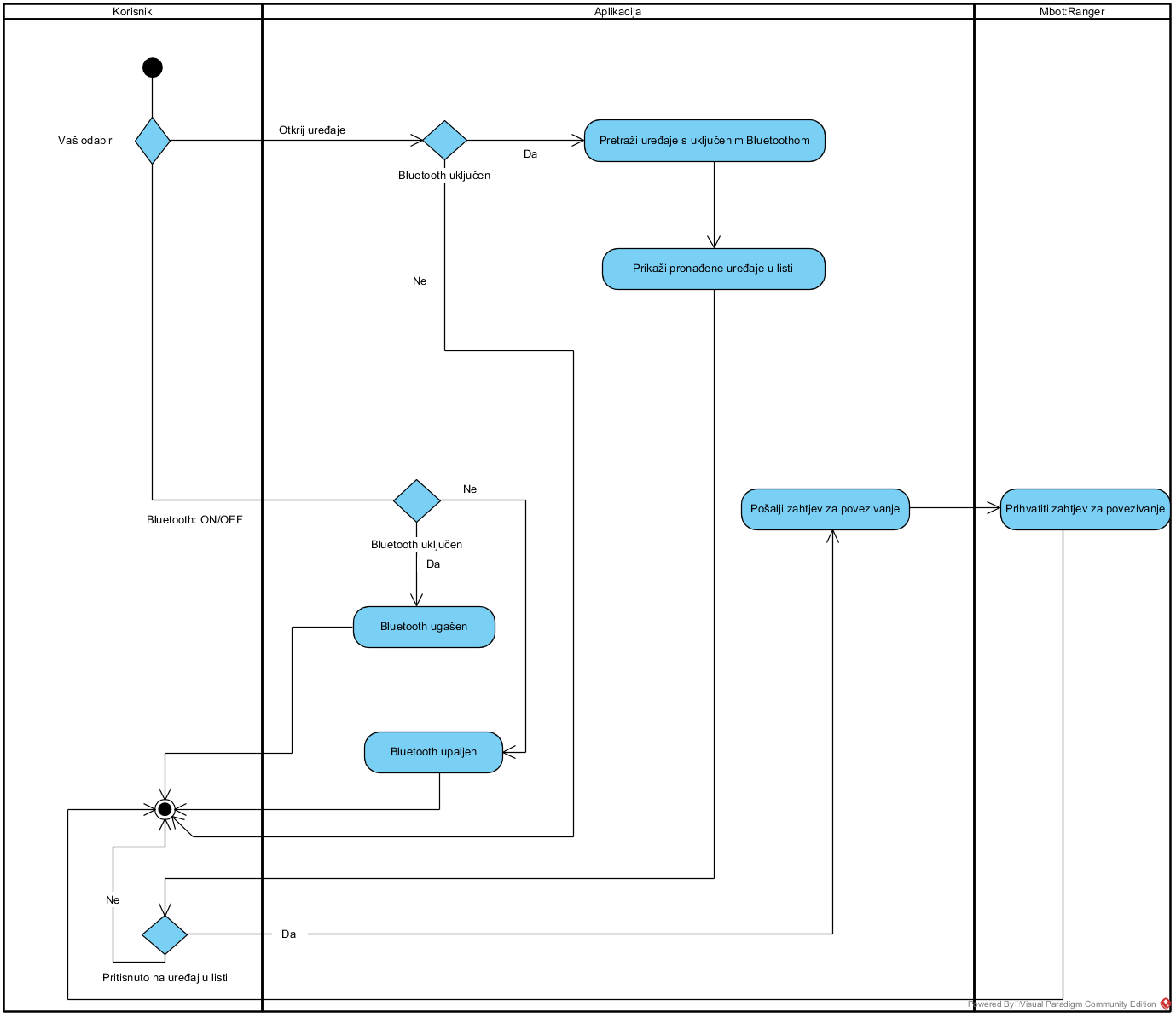


Klasa 4. Controls.java

Dolje prikazana klasa DeviceListAdapter je klasa koja se koristi za prikupljanje svih pronađenih bluetooth uređaja te prikaz istih u listi. Ta lista se zatim prikazuje na početnom zaslonu, a puni se pritiskom na gumb za otkrivanje uređaja.



Klasa 5. DeviceListAdapter.java

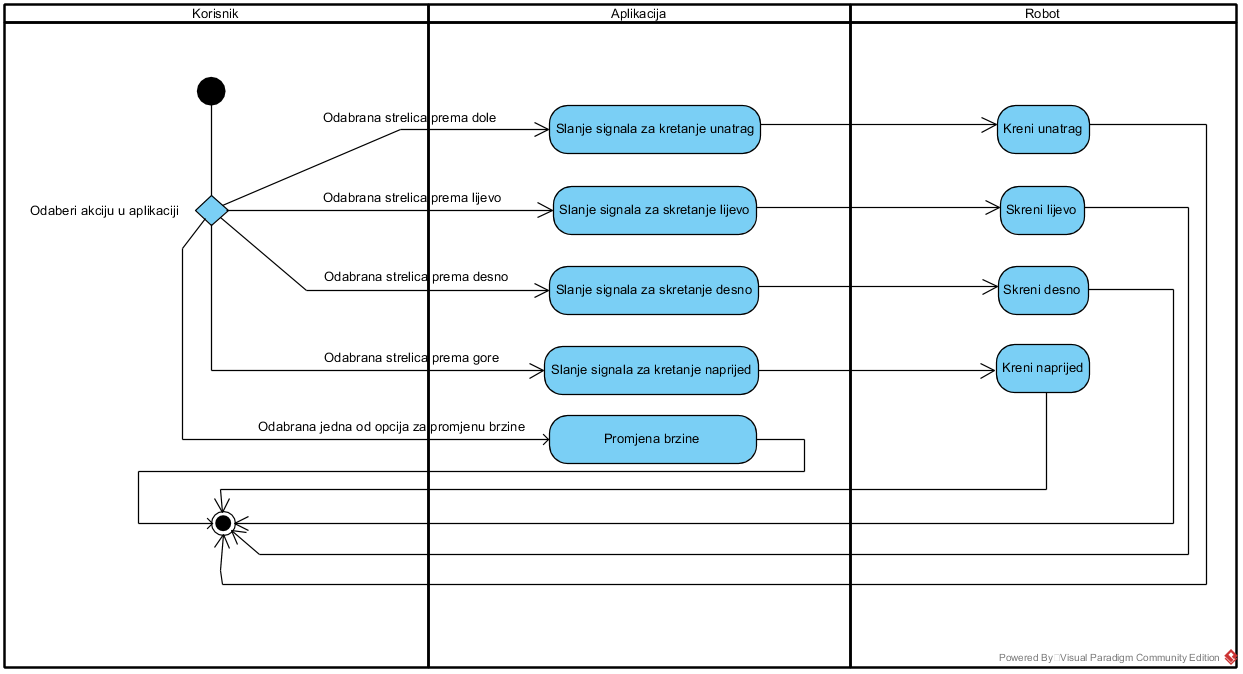


Slika 1. Dijagram aktivnosti povezivanje s bluetoothom

Ulaskom u aplikaciju mRanger korisniku su ponuđene dvije aktinosti. Jedna od aktivnosti je Bluetooth ON/OFF odnosno uključiti ili isključiti bluetooth. Ovisno o stanju bluetooth-a, bluetooth će se uključiti odnosno isključiti. Druga aktivnost je Otkrij uređaje. Odabirom ove aktivnosti, ako je uključen Bluetooth aplikacija pronalazi sve uređaje koji imaju uključeni Bluetooth te ih ispisuje u listi. U slučaju da korisnik pritisne na jedan od uređaja u listi aplikacija šalje zahtijev za povezivanje Mbot:Rangeru koji on prihvaća. Nakon na navedenog niza aktivnosti aplikacija i Mbot:Ranger su uspješno upareni.

## 3.2. Funkcionalnost daljinskog upravljanja robotom

Nakon uspješnog uparivanja aplikacije s Mbot:Rangerom korisnicima se nudi mogućnost upravljanja Mbot:Rangerom. Ponuđene su im četiri mogućnosti: kretanje unaprijed, kretanje unatrag, kretanje lijevo i kretanje desno. Ovisno o odabranoj akciji Mbot:Ranger će izvršiti zadanu naredbu. Uz ovu funkcionalnost dolazi i mogućnost promijene brzine u jednu od preddefiniranih.



Slika 2. Dijagram aktivnosti daljinsko upravljanje

Navedene mogućnosti ostvarili smo putem klasa Controls.java, BaseActivity.java, FragmentBasicKontrole.java, IControls.java i FragmentFancyKontrole.java. Prikaz koda klase Contols.java nalazi se u poglavlju Funkcionalnost povezivanja s bluetoothom. Kodovi ostalih navedenih klasa nalaze se ispod ovog teksta te se njihov cijeli prikaz može dobiti dvoklikom na crni okvir.

Na dolje dostupnom ugrađenom dokumentu prikazana je BaseActivity aktivnost. To je aktivnost unutar koje se pojedini dizajni kontrola, odnosno fragmenti prikazuju pomoću kojih je moguće upravljati robotom.



Klasa 6. BaseActivity.java

Kao osnovni fragment postoji FragmentBasicKontrole također dostupan putem dolje prikazanog ugrađenog dokumenta. Ovo je fragment za kontrolu robota putem jednostavnog izgleda te se koristi kod BaseActivity-a. Dakle, ovaj fragment predstavlja početni izgled aplikacije bez posebno rađenog dizajna. Također, ovakav dizajn aplikacije se dobije sljedeći STEM dokumentaciju.



Klasa 7.FragmentBasicKontrole.java

Kako bismo implementirali modularnost unutar naše aplikacije koristili smo sučelje IControls čiji je prikaz dostupan dolje putem ugrađenog dokumenta. Ovo sučelje koristili smo kako bi povezali dva modula, odnosno kako bi mogli pravilno implementirati fragmente. Oba dizajna koriste jednake metode za upravljanje nad robotom pa je zato kreirano ovo sučelje čije su metode implementirane u klasi Controls.



Klasa 8. IControls.java

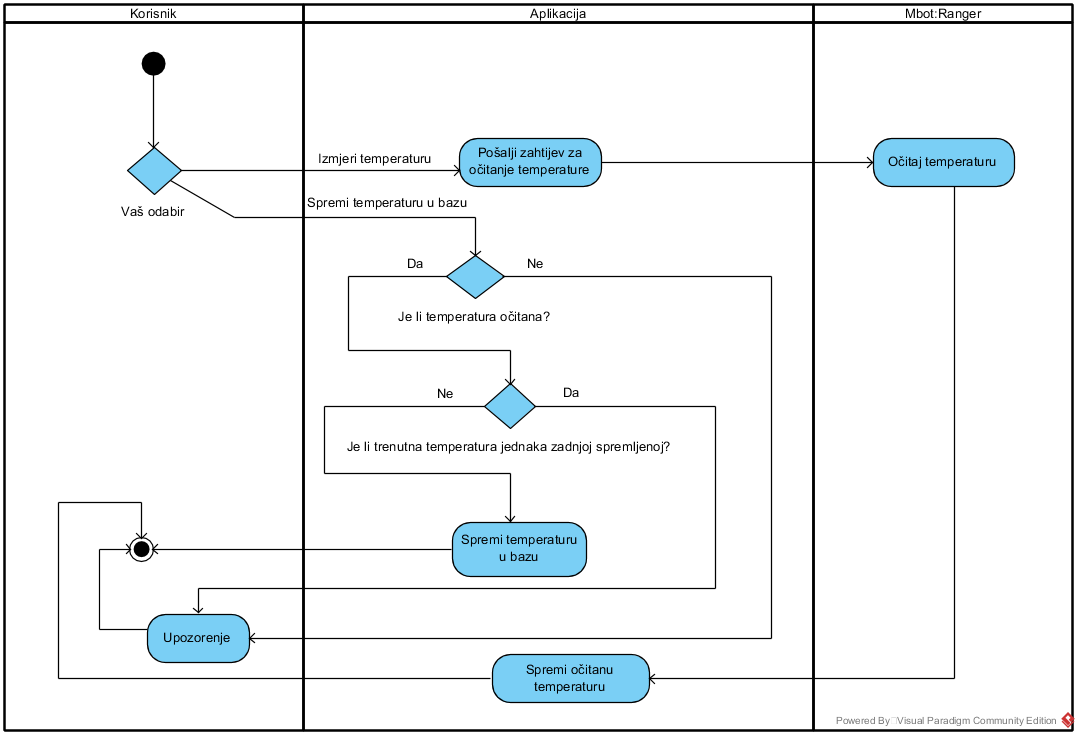
Na kraju dolazi FragmentFancyKontrole koji je isto dostupan putem ugrađenog dokumenta. Ovo je fragment koji omogućava kontrolu robota putem unaprijeđenog dizajna te se također koristi kod BaseActivity-a. Ovaj dizajn je u potpunosti izrađen posebno za ovu aplikaciju koristeći alat Adobe Photoshop te je korišten za kreiranje svih gumba, animacija pa i ikone od mRanger robota.



Klasa 9. FragmentFancyKontrole.java

## *3.3. Funkcionalnost bilježenja temperature zraka u bazu podataka*

*Putem senzora za temperaturu robot ima mogućnost bilježenja temperature zraka koju zatim šalje u bazu podataka te ju dodatno obrađuje na serverskoj strani. Nakon čega se obrađeni podaci prikazuju na web stranici. Dakle, robot na sebi ima ugrađen senzor za mjerenje temperature zraka u okolini te ovisno o okolini u kojoj se nalazi dobiva različitu temperaturu.*



Slika 3. Dijagram aktivnosti mjerenje temperature

Navedena funkcionalnost je prikazana dijagramom aktivnosti. Korisnik na zaslonu na kojemu se nalaze aktivnosti za upravljanje kretanjem robota također ima aktivnosti za očitavanje temperature i spremanje iste u bazu. Korisnik odabirom opcije „Izmjeri temepraturu“ putem aplikacije šalje zahtijev Mbot:Rangeru za očitanje temperature. Ukoliko korisnik odabere opciju „Spremi temperaturu u bazu“ aplikacija prvo provjerava je li temperatura očitana, ako je provjerava se je li očitana temperatura jednaka posljednjoj spremljenoj temperaturi te nakon što su zadovoljene sve provjere temperatura se prema u bazu. Ukoliko jedan od navedena dva uvijeta nije zadovoljen korisnik dobiva upozorenje. Opisana funkcionalnost ostvarena je pomoću istih klasa kao i funkcionalnost daljinskog upravljanja robotom te su navedene u ranijem poglavlju.

# 4. Korišteni alati i tehnologija

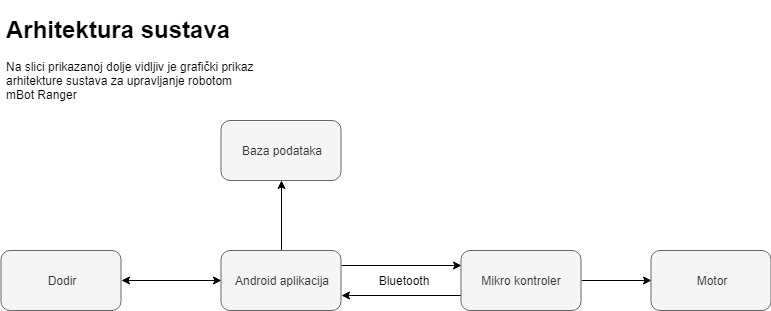
Ovo poglavlje sadrži popis svih korištenih tehnologija i resursa te kratki opis za koje područje koristimo koju tehnologiju odnosno resurs.

|  |  |
| --- | --- |
| Resurs | Količina |
| Prijenosna računala | 5 |
| mBot Ranger | 1 |

|  |  |
| --- | --- |
| Tehnologija | Alat |
| Android IDE | Android studio |
| Adobe Photoshop | Adobe Photoshop CC |
| Visual Paradigm | Visual Paradigm 15.0 |
| Balsamiq | Balsamiq Mockups 3 |
| GitHub | GitHub for Desktop |
| PHP | Sublime Text 3 |
| MySQL | MySQL Workbench 6.3 CE / DbForge Studio Express for MySQL 8.0 |

Za izradu projekta mRanger koristit će se pet prijenosnih računala na kojima su instalirani Android studio, Visual Paradigm, GitHub for Desktop, ostali navedeni alati se nalaze instalirani po pojedinim prijenosnim računalima. Za programiranje same aplikacije koristit će se Android Studio te programski jezik Java. Kako bi se kreirao logo aplikacije te njezin dizajn koristit će se Adobe Photoshop CC. Za izradu tehničke dokumentacije koristit će se alate Balsamiq Mockups 3 i Visual Paradigm 15.0 u kojima se redom izrađuju navigacijski dijagram te dijagrami aktivnosti i dijagram klasa. Aplikacija omogućava očitanje temperature te njezino spremanje u bazu. Iz tog razloga koristi se MySQL tehnologiju pomoću koje se izgrađuje baza podataka u koju se pohranjuje temperatura, za ovu aktivnost je potreba također i PHP tehnologija u kojoj pomoću programskog jezika PHP programiramo spremanje temperature u bazu podataka. Jedini resurs koji nije spomenut je mBot:Ranger pomoću kojega testiramo rad aplikacije.

# 5. Dizajn arhitekture

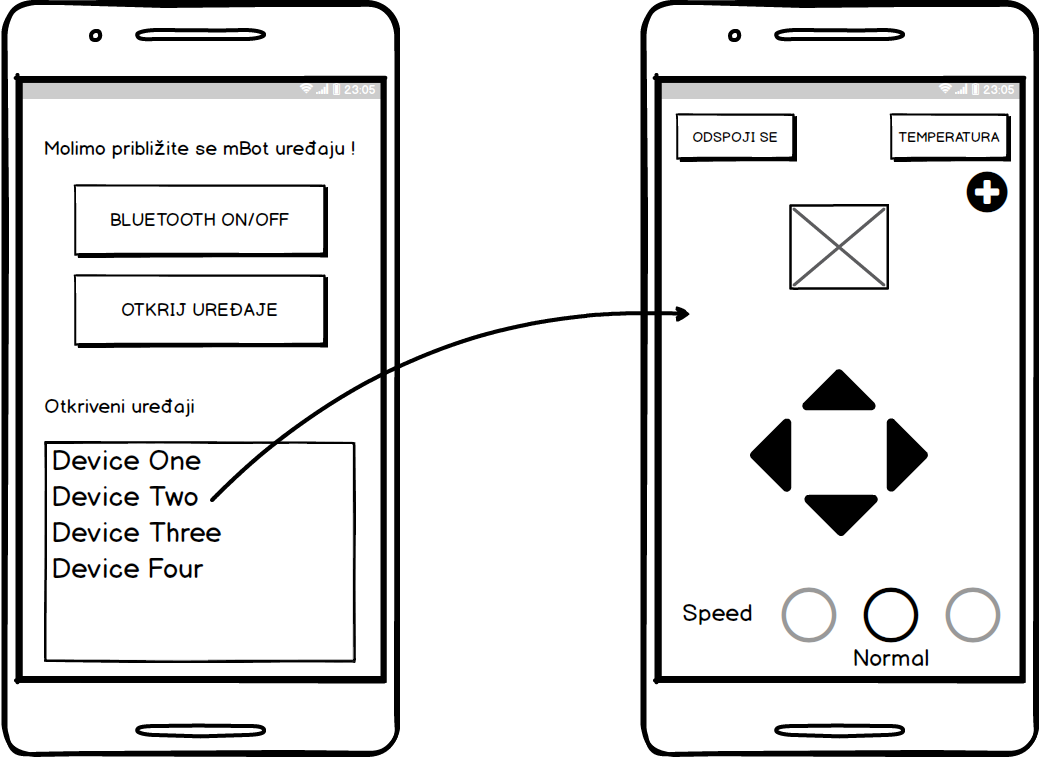


Slika 4. Dizajn arhitekture

Slika 2. prikazuje dizajn arhitekture aplikacije mRanger. Arhitektura nam prikazuje da će se pomoću aplikacije slati signal putem bluetootha na Arduino mikroprocesor koji upravlja komponentama na mBot Rangeru. Aplikacija služi za upravljanje robotom, odnosno njegovim kretanjem te je iz toga razloga na slici prikazan motor kao jedna od ključnih komponenti. Također, aplikacija šalje različite podatke robotu prema kojima robot zna koju funkciju treba pozvati, koju brzinu treba uzeti kao i smjer u kojem se treba kretati.

# 6. Navigacijski dijagram

Pomoću navigacijskog dijagrama definiraju se veze između zaslon, odnosno pritiskom na koji gumb će se otvoriti koji zaslon. Naša aplikacija sadrži dva zaslona, početni zaslon koji prikazuje mogučnosti uključivanja Bluetooth-a, otkrivanja uređaja koji imaju uključeni bluetooth te popis otkrivenih uređaja. Drugi zaslon prikazuje koje sve kontrole možemo izvršavati na Mbot:Rangeru.

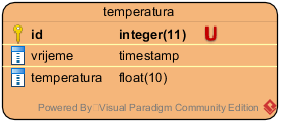


Slika 5. Navigacijski dijagram

Slika 5. prikazuje navigacijski dijagram aplikacije s već opisanim zaslonima. Ukoliko korisnik želi prijeći s prvog zaslona na drugi treba pratiti sljedeće korake: uključiti Bluetooth, otkriti uređaje te nakon što se u popisu otkrivenih uređaja ispišu uređaji, odabere uređaj Makeblock koji predstavlja Mbot:Ranger-a.

# 7. Baza podataka

*Na slici 6. je prikazan model baze podataka korišten u ovom projektu, gdje je model vrlo jednostavno izveden, odnosno imamo samo jednu tablicu u koju se upisuju podaci. Tablica temperatura prima podatke izmjerene temperature s robota i vrijeme kada se to dogodilo. Pohranjeni podaci se tada mogu učitat i obraditi te prikazati korisniku u web pregledniku. Baza podataka je Slika 6. prikazuje ERA dijagram aplikacije mRanger.*

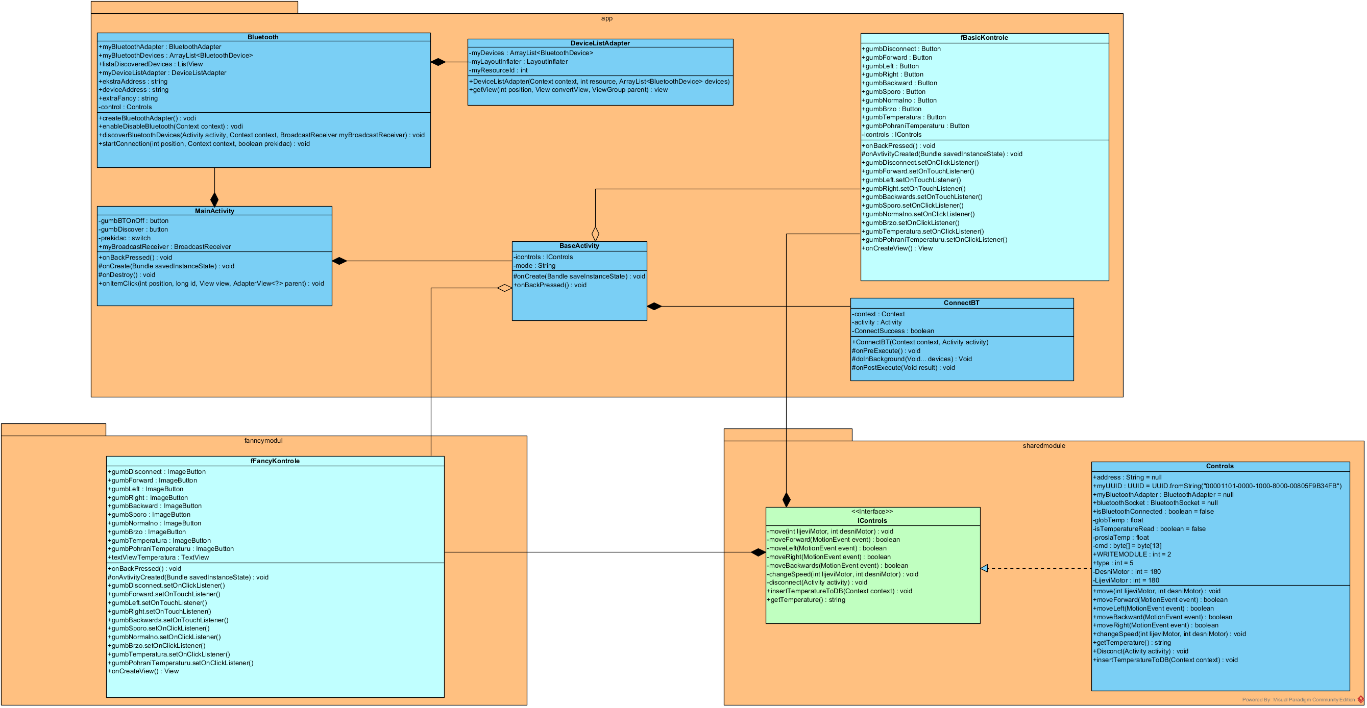


Slika 6.ERA dijagram

Napomena: *naznačeni dio će se implementirati u sklopu kolegija Logičko programiranje koji također sudjeluje u programu STEM revolucija u zajednici. (Kristović, Kraljić, Pavleka)*

# 8. Dijagram klasa

Dijagramom klasa se prikazuju veze između klasa te struktura svake klase. Dijagram klasa aplikacije mRanger se sastoji od 9 klasa: MainActivity, Bluetooth, DeviceListAdapter, ConnectBT, FragmentFancyKontrole, Controls, IControls, FragmentBasicKontrole te BaseActivity. Klasa Bluetooth sadrži sve varijable i metode vezane uz Bluetooth čija je svrha omogučiti uključivanje Bluetootha, pretraživanje uređaja kojima je uključen bluetooth te povezivanje na odabrani pronađeni uređaj. Klasa Controls sadrži sve kontrole nad Mbot:Rangerom kao što su očitavanje temperature, kretanja u svim smjerovima te kontrola nevezana uz Mbot:Rangera spremanje temperature u bazu podataka. U klasi MainActivity se pozivaju metode iz klase Bluetooth, odnosno klasa MainActivity služi za uspostavljanje veze između mobilnog uređaja i Mbot:Rangera. Klasa BaseActivity je klasa koja služi za prikazivanje određenog fragmenta (FragmentBasicKontrole, FragmentFancyKontrole) u ovisnosti o uključenosti fancy moda ili ne. Pomoću navedenih fragmenata možemo updavljati Mbot:Rangerom, odnosno pristupiti metodama klase Controls.



Slika 7. Dijagram klasa

# 9. Modularnost

Aplikacija mRanger je ostvarena pomoću 3 modula (app, fanncymodule, sharemodule). Gdje je app osnovni modul. Grafički prikaz modularnost prikazan je na slici ispod.

U app nalaze se klase:

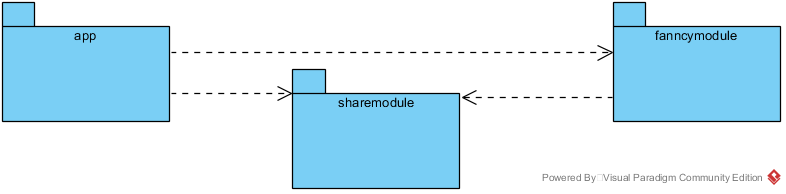
1. MainActivity
2. BaseActivity - activitiy koji prihvaća fragmente za prikaz
3. fBasicKontrole - prvi fragment, odnosno dizajn osnovnog prikaza kontrola
4. DeviceListAdapter – klasa za organizaciju dohvaćenjih bluetooth uređaja u listu koja se prikazuje u MainActivity
5. Bluetooth – klasa s funkcijama za povezivanje na bluetooth
6. ConnectBT – klasa s kojimo se ostvaruje bluetooth konekcija

U fanncymodule nalaze se klase (fragmenti):

1. fFancyKontrole – fragment za prikaz naprednog dizajna kontrola (prikazuje se u BaseActivity)

U sharedmodule nalaze se klase (interface):

1. Controls – klasa u kojoj se nalaze sve metode za upravljanje robotom
2. IControls – interface pomoću kojeg je moguće iz različitih modula vršiti kontrolu robota, korištenje klase Controls



Slika 8. Dijagram modula