

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Luka Košenina

**Tehnologija NFC v izvedbi
glasovalnega sistema**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2014

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Najnovejši modeli pametnih telefonov imajo vključeno tehnologijo NFC (Near Field Communication), ki omogoča optimizacijo različnih aktivnosti, kot so nakupi v trgovinah, preverjanje prisotnosti na dogodkih, identifikacija oseb in druge. Postavlja se vprašanje kako bi z njihovo uporabo v povezavi s sodobnimi tehnologijami optimizirali postopek glasovanja večjega števila udeležencev na različnih predavanjih, sestankih ali javnih dogodkih.

Na osnovi pregleda tehnologije NFC in analize zahtev za izvedbo glasovalnega sistema naj kandidat v diplomskem delu zasnuje ustrežno rešitev, ki vključuje spletno in mobilno aplikacijo. Organizatorju glasovanja naj omogoči pripravo vprašalnikov, izvedbo preverjanja prisotnosti, glasovanja ter analizo rezultatov. Udeleženci naj uporabijo svoj telefon za prijavo v glasovanje in za njegovo izvedbo. Delovanje sistema naj preveri na testnem primeru glasovanja na sestanku z uporabo dveh pametnih NFC naprav ter poda prednosti in slabosti predlagane rešitve.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Luka Košenina, z vpisno številko **63110192**, sem avtor diplomskega dela z naslovom:

Tehnologija NFC v izvedbi glasovalnega sistema

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Mire Trebar,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 12. septembra 2014

Podpis avtorja:

Zahvaljujem se doc. dr. Miri Trebar za pomoč, nasvete ter izkazano potrpežljivost pri izdelavi diplomskega dela. Posebna zahvala gre tudi moji družini in vsem ostalim, ki so mi stali ob strani tekom študija.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Teoretični del	3
2.1	Tehnologija NFC	3
2.2	Android in NFC	5
2.3	P2P komunikacija	6
2.4	Spletno ogrodje Django	6
2.5	Django AllAuth	8
2.6	Django Cron	8
2.7	SQLite	9
2.8	Programska orodja in programski jeziki	10
3	Izvedba glasovalnega sistema NFC	13
3.1	Spletna aplikacija	15
3.2	Mobilna aplikacija	33
4	Testiranje	45
5	Sklepne ugotovitve	51

Seznam uporabljenih kratic

kratica	angleško	slovensko
ADT	android developer tools	android orodja za razvijalce
AES	advanced encryption standard	napreden standard kodiranja
AJAX	asynchronous JavaScript and XML	asinhroni JavaScript in XML
API	application programming interface	vmesnik za programiranje aplikacij
APK	android application package	android aplikacijski paket
HTML	hypertext markup language	jezik za označevanje nadbesedila
IDE	integrated development environment	integrirano razvojno okolje
IEC	international electrotechnical commission	mednarodna komisija za elektrotehniko
IrDA	infrared data association	infrardeče podatkovno združenje
ISO	international organization for standardization	mednarodna organizacija za standardizacijo
JSON	JavaScript object notation	JavaScript objektna notacija
MVC	model view controller	model pogled krmilnik
NDEF	NFC Forum data format	NFC Forum format podatkov
NFC	Near Field Communication	komunikacija kratkega dosega
POS	point of sale	prodajni terminal
P2P	peer to peer	omrežje vsak z vsakim
RFID	radio frequency identification	radiofrekvenčna identifikacija
SQL	structured query language	strukturirani povpraševalni jezik za delo s podatkovnimi bazami
WLAN	wireless LAN	brežično lokalno omrežje
WORA	write once, run anywhere	napiši enkrat, izvajaj kjerkoli

Povzetek

Diplomska naloga obravnava uporabo tehnologije NFC (Near Field Communication) in uporabo sodobnih spletnih in mobilnih tehnologij s katerimi lahko izboljšamo izvajanje javnih in zasebnih glasovanj. Problem identificiranja sodelujočih oseb je izveden z uporabo NFC mobilnih naprav. Za izvedbo glasovanj sta bili izdelani spletna in mobilna aplikacija. Spletna aplikacija je zasnovana v ogrodju Django, ki hrani potrebne podatke s pomočjo sistema za upravljanje relacijskih baz SQLite. Mobilna aplikacija je namenjena za naprave NFC, ki jih poganja operacijski sistem Android. Predstavljen je proces kreiranja in urejanja glasovanja, izdaje dovoljenja za pristop h glasovanju ter oddaje glasu. V povezavi z mobilno aplikacijo je predstavljena tehnologija Android Beam, ki je uporabljena pri identifikaciji uporabnikov.

Ključne besede: glasovanje, NFC, spletna in mobilna aplikacija, Android.

Abstract

The thesis discusses the use of NFC (Near Field Communication) technology and presents the use of modern online and mobile technologies which can improve the performance of public and private voting. The problem of identifying the participating people is carried out using NFC mobile devices. To perform the voting we constructed web and mobile application. The web application is designed in the Django Web Framework, which keeps the necessary information by means of a relational database management SQLite. The mobile app is designed for NFC device, powered by the Android operating system. Featured is the process of creating and editing vote, authorization to accede to vote and voice broadcast. In connection with the mobile application, is introduced Android Beam, which is used for the identification of users.

Keywords: voting, NFC, web and mobile application, Android.

Poglavje 1

Uvod

Dandanes si skoraj pri vsaki dejavnosti, delu pomagamo z različnimi tehnologijami, ki poskrbijo za učinkovito in poenostavljeno izkušnjo. Pri iskanju teme za diplomsko nalogo sem si izbral opravilo, ki bi ga lahko izboljšal s pomočjo mobilne in spletne tehnologije. Naletel sem na primer glasovanja, ki se navadno izvaja v manjši skupini ljudi, kjer mora biti preverjena identiteta sodelujoče osebe z osebnim ali katerim drugim dokumentom. Oseba, ki preveri dokument, dovoli ali pa zavrne pristop h glasovanju in hkrati beleži evidenco sodelujočih. Ob vsaki ponovitvi tovrstnih glasovanj je potrebno pred glasovanjem izvesti proces preverjanja oseb.

Glasovanja, ki se izvajajo s pomočjo mobilnih naprav, z ustrezno uporabo tehnologij omogočajo tako preverjanje identitete lastnika naprave kot tudi pravico do sodelovanja pri izvedbi glasovanja. Na takšen način lahko izvajamo glasovanja pri katerih se določen uporabnik identificira samo ob prvem glasovanju posameznega skrbnika. Pri nadaljnjih glasovanjih lahko skrbnik prenese že izdano dovoljenje na to novo glasovanje. Tovrstna izvedba omogoča sodelujoči osebi pristop h glasovanju brez predhodne identifikacije.

Predlagana rešitev vključuje spletno in mobilno aplikacijo, kjer spletna aplikacija skrbi za izvedbo in upravljanje z glasovanji. Mobilna aplikacija za Android pa omogoča preverjanje identitete sodelujočih oseb z uporabo tehnologije NFC. Samo oddajo glasu lahko uporabnik izvede na poljubni napravi, kot je pameten telefon, tablični, prenosni ali namizni računalnik. Potrebuje le povezavo do svetovnega spleta in dovoljenje avtorja, če pristopa k zasebnemu glasovanju.

V nalogi je podrobneje opisana implementacija glasovalnega sistema in nje-

gove funkcionalnosti z uporabo mobilne in spletne aplikacije. V začetnih poglavjih najprej spoznamo vso tehnologijo, ki jo uporablja glasovalni sistem. Naslednje poglavje opisuje načrtovanje spletne in mobilne aplikacije, kjer spoznamo lastnosti in način uporabe posamezne aplikacije. Sledi konkreten primer uporabe sistema za glasovanje, ki vključuje korake od ustvarjanja glasovanja, izdaje dovoljenj, oddaje glasu do zadnjega koraka, kjer zaključimo glasovanje in prikažemo rezultate. Sklepne ugotovitve in možne izboljšave sistema za glasovanje so zbrane v zaključku naloge.

Poglavje 2

Teoretični del

Mobilne naprave so s pomočjo NFC tehnologije pridobile velik prostor za izboljšave že obstoječih aplikacij. Omogočajo preprostejšo izvedbo določenih opravil, kot so na primer plačilo nakupa, plačilo vožnje mestnega prometa z mobilno napravo, itd. Vendar tovrstne aplikacije ne delujejo samostojno in za ustrezno izvedbo potrebujejo strežniški del aplikacije. Tovrstna arhitektura aplikacij omogoča hitrejše delovanje na strani odjemalca, varnejšo izvedbo opravil in boljšo dostopnost končnim uporabnikom. Predlagana rešitev aplikacij, ki so uporabljene v izvedbi glasovanja z mobilnimi NFC napravami, zajema različne računalniške tehnologije.

2.1 Tehnologija NFC

Near Field Communication (NFC) je brezžična tehnologija kratkega dosega, ki je bila razvita pod skupnim okriljem podjetja Philips in Sony. Decembra 2003 je bila tehnologija standardizirana s strani Mednarodne Organizacije za Standardizacijo (ISO) in Mednarodne Elektrotehnične Komisije (IEC). Omogoča enostavno, varno in intuitivno P2P (ang. peer to peer) komunikacijo med napravami, ki podpirajo to tehnologijo. Temelji na radio frekvenčni identifikaciji (RFID) in uporablja magnetno polje za komunikacijo med elektronskimi napravami [1]. Pri javanskih mobilnih aplikacijah je ta tehnologija dostopna preko uporabe Contactless Communication API. Tehnologija je združljiva z obstoječo brezžično infrastrukturo in mobilnimi napravami, ki vsebujejo NFC vmesnik lahko posnemajo pametne kartice (ISO 14443), lahko opravljajo naloge bralnika in lahko celo udeležujejo P2P

komunikacijo. Naštete lastnosti omogočajo mobilnim napravam, ki podpirajo NFC tehnologijo, povezavo do spletne strani preko branja zapisa na znački, plačevanje transakcij namesto uporabe plačilnih kartic, itd.

RFID sistem sestoji iz dveh komponent, odzivnika (ang. transponder) in bralnika oziroma pisalnika (ang. transceiver). Je asimetrična tehnologija, saj lahko le bralno/pisalna naprava sproži prenos podatkov [2]. Nasprotno NFC omogoča dvosmerno komunikacijo med napravama. NFC uporablja isti komunikacijski mehanizem kot RFID tehnologija. Združuje prednosti RFID in brezžične tehnologije, kot so Bluetooth, WLAN, IrDA. Leta 2003 so bili predlagani NFC standardi, ki določajo delovanje na ne licenčnem frekvenčnem pasu 13.56MHz. Hitrost prenosa podatkov z NFC je do 424 kBit/s med 10 cm oddaljenima napravama [3]. Notranji prenosni modul lahko deluje kot bralnik oziroma zapisovalnik za dostop do značke (ang. tag) in brezkontaktna pametne kartice.

NFC tehnologija deluje tako, da prenos podatkov sprožimo z dotikom dveh NFC naprav. Takrat napravi nemudoma začneta komunicirati. Prednost te tehnologije je, nepotrebno predhodno združevanje naprav v komunikaciji, kot ga poznamo pri Bluetooth komunikaciji.

Tehnologija NFC uporablja naslednje pametne naprave [13]:

- **NFC značka:** Tukaj gre pravzaprav za RFID značko, ki zahteva tuj vir napajanja in z nje lahko samo beremo podatke. Posledično dve znački med seboj ne moreta izmenjati podatkov.
- **NFC čitalec:** NFC čitalec omogoča branje podatkov iz poljubne NFC naprave. Najpogostejši primer uporabe je POS terminal, ki omogoča brezkontaktno izvajanje plačil, ko preide NFC naprava v stik z NFC čitalcem.
- **Pametni telefon, ki podpira NFC tehnologijo:** Mobilnim telefonom, ki omogočajo NFC komunikacijo, pogosto pravimo kar NFC telefoni. To so najpomembnejše NFC naprave in so hkrati zaslužne za porast zanimanja za NFC tehnologijo. Omogočajo tako branje kot pisanje z napravami v komunikaciji.

Komunikaciji med dvema NFC napravama pravimo tudi seja (ang. session). Napravi, ki sproži komunikacijo pravimo pobudnik (ang. initiator), medtem ko napravi, ki se odzove pravimo cilj (ang. target). Takšen primer komunikacije nam

je najboljše znan iz arhitekture odjemalec-strežnik, ki jo srečamo pri računalniškem omrežju. Spodnja tabela prikazuje možne kombinacije med pobudniki in cilji.

Pobudnik	Cilj
NFC telefon	NFC značka
NFC telefon	NFC telefon
NFC bralnik	NFC telefon

2.2 Android in NFC

Operacijski sistem Android podpira NFC tehnologijo od verzije Android 2.3 (API 9) dalje. Sprva je bilo podprto samo branje NFC značk šele kasnejše verzije so omogočale bolj obsežno podporo [8]. Android mobilne naprave, ki omogočajo NFC komunikacijo, podpirajo tri načine delovanja [7]:

- **Bralno/pisalni način:** NFC napravam omogoča branje in/ali pisanje na pasivne NFC značke ali nalepke.
- **P2P način:** Omogoča izmenjavo podatkov med dvema NFC napravama. Takšen način uporabe zagotavlja funkcija Android Beam.
- **Posnemanje kartic:** Zadnji način omogoča NFC napravam, da se obnašajo kot NFC kartica. Posnemana NFC kartica je dosegljiva zunanjemu NFC čitalcu, kot je na primer NFC POS terminal.

Android Beam je funkcionalnost, implementirana znotraj operacijskega sistema Android in omogoča preprosto P2P komunikacijo. Podpira hitro izmenjavo podatkov, kot so spletni zaznamki, podatki iz imenika, navigacijska navodila, YouTube video posnetki in še bi lahko naštevali. Pri prenosu večje količine podatkov priskoči na pomoč tehnologija Bluetooth. V takšnem primeru Android Beam poskrbi za vklop povezave in identifikacijo obeh naprav. Ob končanem prenosu Beam samodejno prekine in izklopi povezavo. Ta lastnost se izkaže za ključno, saj Bluetooth lahko veliko hitreje prenese podatke in deluje tudi na daljši razdalji, kar pride še kako prav pri daljšem času prenosa podatkov.

2.3 P2P komunikacija

P2P omogoča NFC napravama ob dotiku izmenjavo podatkov. Uspeh komunikacije je odvisen od izpolnitve naslednjih pogojev:

- Obe napravi morata imeti vključen Android Beam.
- Aplikacija, ki želi prožiti komunikacijo mora teči v ospredju.
- Na obeh napravah se ob dotiku prikaže Touch to Beam uporabniški vmesnik. Uporabnik, ki želi poslati podatke se mora dotakniti zaslona in s tem sprožiti proces komunikacije.

Podatki za izmenjavo morajo biti zajeti v ustreznem NDEF formatu [8]. NFC Data Exchange Format (NDEF) specifikacija določa obliko zapisa podatkov za izmenjavo med dvema NFC napravama ali med NFC napravo in NFC značko. Je okrnjen binarni format sporočila, ki se uporablja za enkapsulacijo aplikacijsko koristnih podatkov poljubnega tipa in velikosti v eno samo sporočilo. Sporočilo sestavlja eno ali več NDEF zapisov. Iz splošnega vidika lahko rečemo, da je NDEF sporočilo nabor zapisov. Število le teh znotraj sporočila je odvisno od aplikacije in tipa NFC značke. Med tem, ko je velikost zapisa po specifikaciji omejena na velikost $2^{32} - 1$ oktet. Vsak zapis nosi tri parametre s katerimi opisuje svoje podatke: dolžina podatkov, tip podatkov in identifikator podatkov [4].

2.4 Spletno ogrodje Django

Django je bil prvič predstavljen leta 2003 v Kansasu. To je spletno ogrodje, ki temelji na programskem jeziku Python in njegov cilj je hitro izdelovanje dinamičnih in odzivnih spletnih aplikacij. Ogrodje temelji na vzorcu MVC, kar je kratica za Model View Controller, in nam omogoča, skladno in kompaktno arhitekturo projekta [6]. MVC vzorec je bil izdelan z namenom ločitve logičnega in prikazovalnega dela aplikacije ter predstavlja naslednje tri plasti aplikacije:

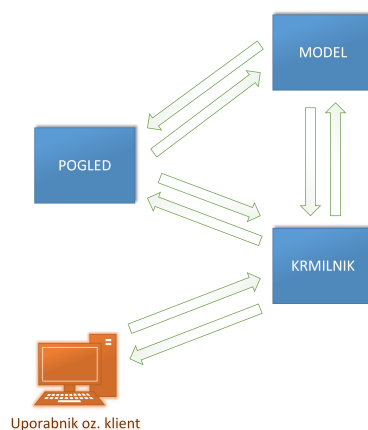
- **Modeli (ang. models):** Tukaj je predstavljena podatkovna struktura aplikacije. Na posplošenem primeru lahko rečemo, da vsak model definira tabelo v podatkovni bazi in relacije med različnimi modeli.

- **Pogledi (ang. views):** Vsebuje vse potrebne informacije iz modelov, katere se uporabijo za generiranje izhodne oblike, ki jo vidijo uporabniki.
- **Krmilnik (ang. controller):** Predstavlja logično plat aplikacije, ki se izvaja na strežnikih in ni na voljo klientom. Tukaj se na primer izvaja identifikacija uporabnikov, poizvedbe po podatkovni bazi in tako dalje.

Tovrstni vzorec omogoča učinkovit način razvoja spletne programske opreme. Ločitev nalog v diskretne skupine služi v različne namene. Programsko kodo, izdelano za reševanje določenega sklopa opravil je veliko lažje vzdrževati, saj ni potrebno predpostavljati o popolnoma ne povezanih delih aplikacije. V splošnem je ta zasnova imenovana ločitev skrbi (ang. separation of concerns) in se uporablja v celotnem postopku razvoja programske opreme v spletnem ogrodju Django. Prav tako razvoj aplikacij postane bolj prilagodljiv, saj lahko več izrazito drugačnih pogledov in krmilnikov povežemo z enim samim modelom. To omogoča različnim aplikacijam deljenje iste poslovne logike in podatkov za več pogledov in različne načine interakcije za različne ciljne skupine.

Slika 2.1 prikazuje posamezne korake izvajanja aplikacije po vzorcu MVC. V prvem koraku klient pošlje zahtevek na stran strežnika za izris strani. V naslednjem koraku krmilnik uporabi podatkovno bazo s pomočjo modelov. Na voljo ima ustvarjanje, branje, spreminjanje in brisanje podatkov. V istem koraku se izvedejo še vsa preostala potrebna opravila, da pridobimo končen skupek informacij, ki jih bomo prikazali uporabniku. Prikazani podatki se generirajo v razdelku pogled, ki tudi vrne HTML vsebino, ki jo krmilnik posreduje odjemalcu.

Resnična prednost ogrodja Django se prikaže ob uporabi različnih brezplačnih vtičnikov. Za vse manj večje uporabnike ogrodja Django je tu potrebno navesti, da je posamezen projekt znotraj ogrodja sestavljen iz vsaj ene aplikacije in poljubno mnogo vtičnikov. Ta nujno potrebna aplikacija je navadno naša glavna aplikacija, ki opravlja glavnino dela. Poleg imamo lahko nameščene vtičnike, ki opravljajo posamezna specifična opravila. Na takšen način lahko ločimo specifična opravila, kot so identifikacija uporabnikov, deljenje podatkov z družabnimi omrežji, komentiranje člankov in tako dalje.



Slika 2.1: MVC vzorec.

2.5 Django AllAuth

Django AllAuth je združen sklop vtičnikov, ki skrbijo za identifikacijo, registracijo, vodenje računa, kot tudi za overjanje zunanjih socialnih uporabniških računov [9]. Vsebuje tudi že izdelane poteke potrjevanja E-mail naslova, pozabljenega gesla, itd. Podpira vrsto zunanjih ponudnikov uporabniških računov s pomočjo katerih se lahko uporabnik prijavi v spletno aplikacijo. Sem spadajo uporabniški računi priljubljenih družabnih omrežij, kot so Facebook, Twitter, Google, Instagram, Amazon in tako dalje. Vtičnik je mogoče tudi prilagajati lastnim zahtevam. Izvorna različica modela uporabnika hrani le njegove osnovne podatke in v primeru, da želimo dodati neki atribut je potrebno samo napisati lasten model, ki razširja standardnega. Prav tako so standardni pogledi za prijavo, registracijo in drugo, dokaj skopi in brez vsakršnega oblikovanja. Lastno oblikovanje vpeljemo z ustvarjanjem lastnih predlog za zelene poglede.

2.6 Django Cron

Linux ali UNIX podobni operacijski sistemi poznajo tako imenovana 'cron' opravila. Uporabljajo se za časovno določeno izvajanje opravil [10]. Izvajajo lahko posamezen ukaz, kot tudi celotne skripte ukazov. Opravila se hranijo v tako imenovani CRON TABLE spremenljivki, ki je pravzaprav polje definiranih opravil.

Vsako minuto sistem preveri, ali je treba v polju opravil izvesti kakšno opravilo. Preverja se vrednost trenutnega časa z vrednostjo pravila izvajanja posameznega opravila. Definicija strukture zapisa o intervalu ponovitev se razlikuje glede na uporabljen sistem in program.

Vtičnik Django Cron omogoča izvajanje periodičnih opravil na strani strežnika. V poljubnih časovnih intervalih izvaja programsko kodo, ki je napisana v programskem jeziku Python ali Django. Ob izvedbi posameznega opravila beleži uspeh in se po potrebi tudi odzove glede na končno stanje opravila. Vtičnik se navadno uporablja za preverjanje določenih podatkov v modelih aplikacije, ustvarjanje varnostnih kopij podatkov ali brisanje zastarelih podatkov. Časovni interval je lahko poljuben in lahko nosi vrednost nekaj minut lahko pa tudi samo enkrat na leto izvede določeno opravilo. Določamo lahko tudi dodatna pravila izvajanja, kot je na primer: opravilo izvajaj trikrat dnevno in vsako prvo nedeljo v mesecu opravi še dodatno opravilo ob točno določeni uri.

2.7 SQLite

SQLite je sistem za upravljanje relacijskih podatkovnih baz [11]. V nasprotju z drugimi sistemi za upravljanje podatkovnih baz je SQLite knjižnica procesov, ki izvajajo samostojen, brez strežniški transakcijski SQL podatkovni motor. Za razliko od večine drugih SQL podatkovnih baz, nima ločenega strežniškega procesa. Branje in pisanje se izvajata neposredno v običajne datoteke v pomnilniku. Celotna podatkovna baza z več tabelami, indeksi, sprožilci je vsebovana v eni sami datoteki na podatkovnem disku. Format datoteke podatkovne baze je podprt s strani različnih platform, kar omogoča preprost prenos podatkovne baze med 32 bitnimi in 64 bitnimi sistemi ali med sistemi, ki uporabljajo bodisi 'big-endian' ali 'little-endian' zapis bitov v pomnilnik.

Django v novejših verzijah podpira tudi zelo uporaben dodatek, ki se imenuje South. Uporablja se za izvajanje migracij nad podatkovno bazo. Razvijalcem olajša delo ob vsaki spremembi poljubnega polja v modelu aplikacije. Prav tako ohranja zgodovino sprememb v strukturi podatkovne baze, tako lahko kadarkoli ponastavimo polja modela na eno izmed prejšnjih stanj migracij.

2.8 Programska orodja in programski jeziki

Razvoj programskih rešitev poteka najhitreje s pomočjo uporabe integriranih programskih okolij. Integrirano razvojno okolje (krajše IDE) je programska aplikacija, ki programskim razvijalcem ponuja celovito infrastrukturo za razvoj programske opreme. Ta navadno sestoji iz urejevalnika izvorne kode, orodja za izgradnjo programa in iskalnika napak. Bolj napredna razvojna okolja nudijo tudi prevajalnik ali/in tolmač izvorne kode.

Predlagana rešitev je napisana v dveh različnih programskih okoljih proizvajalca JetBrains. Spletna aplikacija je izdelana v programskem okolju PyCharm, ki se uporablja za programiranje v programskem jeziku Python [12]. Orodje nudi analizo programske kode, grafični razhroščevalnik, integrirano testiranje enote in razvoj Django spletnih aplikacij. Deluje na različnih platformah operacijskega sistema in je izdan pod dvojno licenco. Pycharm Community Edition je izdan pod Apache License in je brezplačen v bolj okrnjeni obliki od plačljive različice.

Programski jezik Python je splošno namenski, visoko nivojski programski jezik. Zasnovan je z namenom dobre berljivosti programske kode in njegova sintaksa razvijalcem omogoča izražanje zasnov z manj vrstic kode, kot bi bilo mogoče v programskih jezikih, kot je C. Jezik določa konstrukte, namenjene za pisanje čistih programov v malem kot tudi velikem obsegu. Podpira različne vzorce programiranja, vključno objektno orientirano, funkcionalno programiranje in postopkovni slog programiranja. Vključuje dinamičen tip sistema in samodejno upravljanje pomnilnika ter zagotavlja obsežno standardno knjižnico.

Mobilna aplikacija je napisana v programskem jeziku Java in izdelana s pomočjo programskega okolja Android Studio. Android Studio je novo Android razvojno okolje, ki temelji na IntelliJ IDEA. Razvojno okolje prinaša nove funkcionalnosti in izboljšave v primerjavi z Eclipse ADT. Med njimi so bolj pomembne:

- Prilagodljiv izgradnji sistem Gradle.
- Izgradnja različice in večkratno APK generiranje.
- Podpora razširjenih predlog za Google storitve in različne vrste naprav.
- Bogat urejevalnik zгледа aplikacije s podporo urejanja teme.

- Orodja za merjenje zmogljivosti, uporabnosti, združljivost različice in drugih težav.
- Vgrajena podpora za Google Cloud Platform.

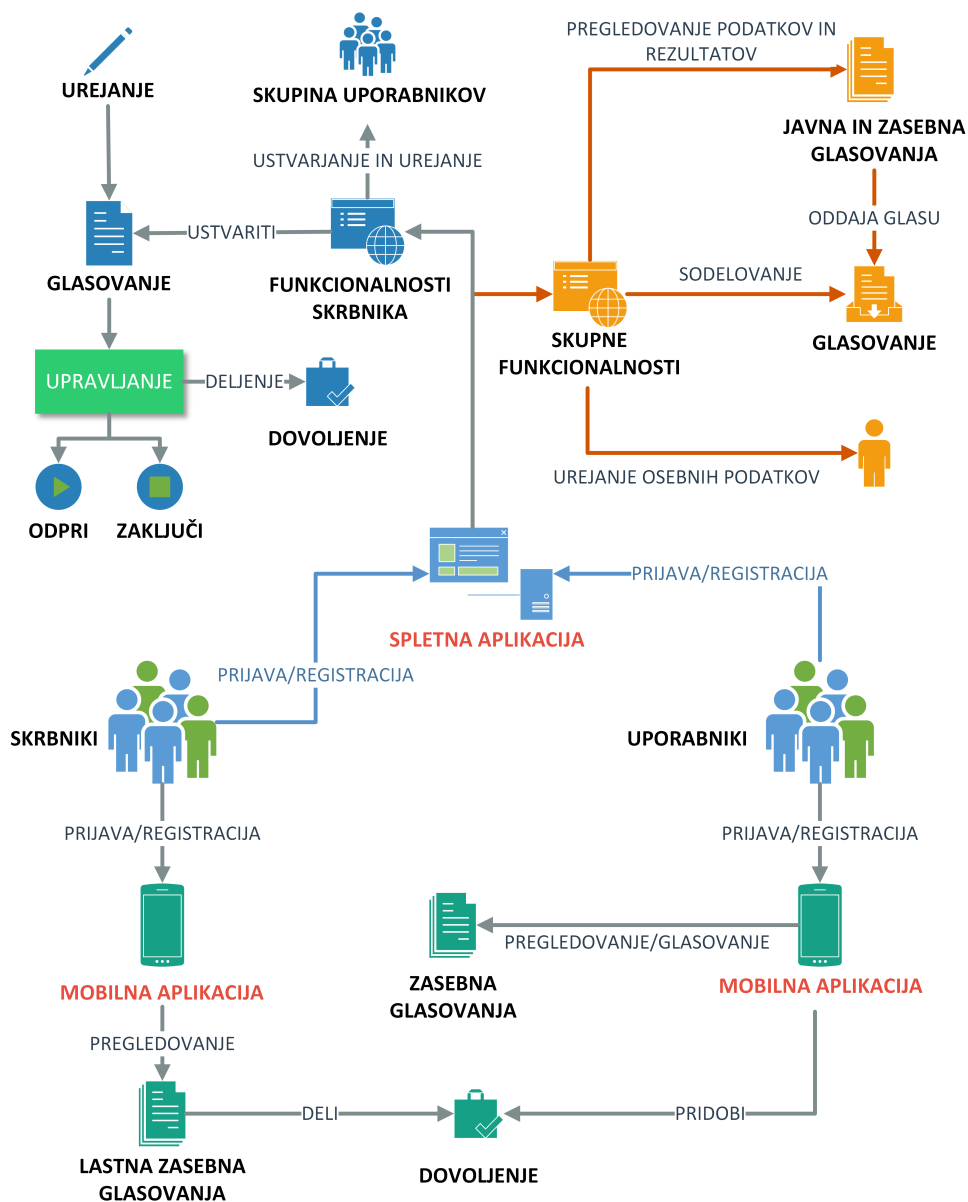
Java je programski jezik, ki temelji na razredih, je objektno usmerjen in stremi k čim manj implementacijskih odvisnosti, kot je le mogoče [5]. Razvijalcem omogoča “write once, run anywhere” (WORA), kar z drugimi besedami pomeni, da kodo spisano za poljubno platformo ni potrebno ponovno prevesti, da bo le ta delovala tudi na drugi platformi. Java je trenutno eden izmed najbolj priljubljenih programskih jezikov v uporabi, predvsem na področju odjemalec-strežnik spletnih aplikacij.

Poglavje 3

Izvedba glasovalnega sistema NFC

Spletno glasovanje je prisotno že dlje časa in ponuja hitro in preprosto glasovanje, kjer koli se že nahajamo. Skupno vsem obstoječim tovrstnim glasovanjem je, da so javna in so na voljo vsem uporabnikom teh storitev. Kot cilj diplomske naloge smo si zadali implementirati spletno glasovanje, ki bo poleg javnih glasovanj podpiralo tudi zasebna. Izvajanje zasebnih glasovanj zahteva preverjanje identitete sodelujočih oseb. Navadno oseba v tem koraku potrdi svojo identiteto z osebnim dokumentom. Na tem mestu si lahko pomagamo z mobilnimi napravami, ki podpirajo tehnologijo NFC, ter izvedemo preverjanje identitete hitreje in učinkovitejše. Prednost tovrstnega preverjanja je, da lahko ob uspešnem preverjanju identitete samodejno izvedemo še dodatna opravila, kot je na primer beleženje evidence identificiranih oseb. V razdelku izdaje dovoljenj bomo spoznali še preostale prednosti tovrstnega pristopa preverjanja osebnih podatkov.

Izvedba glasovanj zahteva delitev glasovalnega sistema na spletno in mobilno aplikacijo. Slika 3.1 prikazuje shemo funkcionalnosti spletne in mobilne aplikacije. Mobilna aplikacija poskrbi za identificiranje sodelujočih oseb, med tem, ko spletna aplikacija skrbi za vsa preostala opravila glasovanja. Mobilno aplikacijo si lahko predstavljamo kot dodatek, ki omogoča izvajanje zasebnih glasovanj.



Slika 3.1: Pregled funkcionalnosti glasovalnega sistema NFC.

Glasovalni sistem mora hraniti podatke o vseh uporabnikih, saj drugače ni mogoče preveriti identitete posamezne osebe oziroma uporabnika. Tukaj je potrebno navesti, da preverjanje identitete sodelujočih oseb izvaja oseba, ki je avtor glasovanja. Iz varnostnih razlogov nobena druga oseba nima pravice izdajati dovoljenj za posamezno glasovanje.

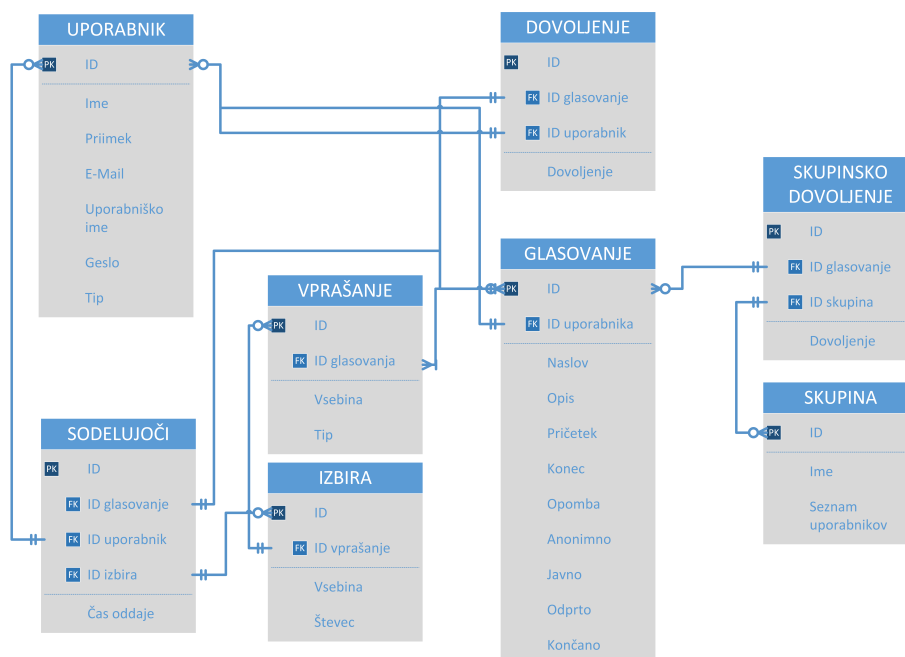
3.1 Spletna aplikacija

Spletna aplikacija omogoča upravljanje z glasovanji, sem spada ustvarjanje, urejanje in brisanje glasovanj ter upravljanje z dovoljenji pristopa h glasovanju. Kot najpomembneje pa omogoča tudi samo glasovanje oziroma oddajo glasu in pregled rezultatov glasovanja. Dostop do funkcionalnosti spletne aplikacije imajo le registrirani in prijavljeni uporabniki. Vsak uporabnik med registracijo lahko izbere, ali želi postati skrbnik. Aplikacija loči dvoje uporabniških vlog, uporabniško in skrbniško. Uporabniki lahko pregledujejo podrobnejše podatke, rezultate in sodelujejo pri javnih in glasovanjih za katera so že prejeli dovoljenja. Ureajo lahko podatke uporabniškega računa, kjer imajo na voljo tudi postati skrbnik. Vloga skrbnika izhaja iz vloge uporabnika, kar pomeni, da ima na voljo funkcionalnosti uporabnika in dodatne za ustvarjanje novega glasovanja, urejanje in brisanje lastnih glasovanj, ustvarjanje in urejanje skupin uporabnikov, deljenje posameznih ali skupinskih dovoljenj.

3.1.1 Podatkovni model

Podatkovni model hrani podatke o glasovanjih, uporabnikih spletne in mobilne aplikacije in uporabnikovih dovoljenj za posamezno glasovanje. Hranjenje podatkov o glasovanju je razdeljeno na več entitet. Glavna entiteta je Glasovanje z vsemi vprašanji, izbirami in dovoljenji. Za ustrezno hranjenje podatkov mora tvoriti relacije z entitetami Uporabnik, Vprašanje, Dovoljenje, Skupinsko dovoljenje in Sodelujoči. Shema podatkovnega modela je podana na sliki 3.2 z vsemi entitetami in relacijami med njimi.

Entiteta Glasovanje potrebuje atribut za enolično identifikacijo, na katerega se sklicujejo entitete v relaciji nanjo. Prav tako potrebuje atribut, ki enolično identificira avtorja glasovanja. Ker so podatki o avtorju predstavljeni z entiteto



Slika 3.2: Podatkovni model glasovalnega sistema NFC.

Uporabniki je atribut poimenovan ID uporabnika. Poleg navedenih dveh atributov potrebujemo še preostale attribute, ki nosijo informacije o glasovanju in njegovih lastnostih. Celoten seznam atributov relacije Glasovanje je sestavljen iz:

1. **ID** – Enolično identificira glasovanje.
2. **ID uporabnika** – Enolično identificira avtorja glasovanja.
3. **Naslov** – Naslov glasovanja.
4. **Opis** – Opis glasovanja.
5. **Pričetek** – Čas pričetka glasovanja.
6. **Konec** – Čas konca glasovanja.
7. **Opomba** – Opomba glasovanja.
8. **Anonimno** – Identifikacija oddanega glasu.
9. **Javno** – Informacija ali je glasovanje javno ali zasebno.

10. **Odprto** – Informacija ali je glasovanje v poteku.

11. **Končano** – Informacija ali je glasovanje končano.

Glede na vrednosti atributov Pričetek, Konec, Odprto in Končano je lahko posamezno glasovanje vedno v enem izmed treh stanj:

- **Pred pričetkom:** To je začetno stanje vsakega ustvarjenega glasovanja. To stanje traja dokler bodisi skrbnik sam ne odpre glasovanja bodisi se atribut pričetek izenači z trenutnim časom. V tem času ima skrbnik na voljo urejanje glasovanja, kasnejših sprememb ne mora vnašati.
- **V poteku:** To je naslednje stanje glasovanja v katerem lahko uporabniki glasujejo. Stanje se zaključi, ko trenutni čas prekorači vrednost atributa konec oziroma v primeru, da skrbnik predčasno zapre glasovanje. V tem stanju in v stanju pred tem lahko skrbnik, v primeru zasebnega glasovanja, deli dovoljenja uporabnikom.
- **Zaključeno:** V zadnjem in hkrati končnem stanju se onemogoči deljenje dovoljenj in oddajo glasov. Prav tako se izračunajo rezultati glasovanja. Ko glasovanje doseže to stanje ga ni moč ponovno aktivirati in odpreti.

Entiteta Uporabnik predstavlja posameznega uporabnika aplikacije. Uporabnik je lahko skrbnik ali pa samo običajen uporabnik. Entiteti Uporabnik in Glasovanje tvorita relacijo ena proti mnogo, kar omogoča, da ima en uporabnik lahko poljubno število glasovanj. O uporabniku beležimo le nujno potrebne podatke, ki jih predstavljajo spodnji atributi:

1. **Ime** – Ime uporabnika.
2. **Priimek** – Priimek uporabnika.
3. **E-Mail** – E-Mail naslov uporabnika.
4. **Uporabniško ime** – Uporabniško ime uporabnika.
5. **Geslo** – Geslo uporabnika.
6. **Tip** – Tip uporabniškega računa.

Vsako glasovanje mora imeti vsaj eno vprašanje, vsako vprašanje pa vsebuje vsaj dve izbiri. Takšno pravilo definiramo s pripravo relacije ena proti mnogo med entitetama Glasovanje in Vprašanje ter med entitetama Vprašanje in Izbira. Posamezno vprašanje je lahko poljubnega tipa. Skrbnik glasovanja lahko izbira med tipom vprašanja, ki dovoljuje le eno izbrano izbiro ali takšnim, ki dovoljuje več izbranih izbir. Atributi entitete Vprašanje so naslednji:

1. **ID glasovanja** – Enolično identificira glasovanje.
2. **Vsebina** – Besedilo vprašanja.
3. **Tip** – Tip vprašanja.

Prikaz rezultatov glasovanja zahteva štetje, kolikokrat je bila posamezna izbira izbrana tekom glasovanja. V ta namen entiteta Izbira vsebuje poleg ostalih atributov tudi atribut števec:

1. **ID vprašanja** – Enolično identificira vprašanje.
2. **Vsebina** – Besedilo izbire.
3. **Števec** – Kolikokrat je bila izbira izbrana.

Ob oddaji glasu je treba shraniti, kateri uporabnik je izbral katero izbiro pri katerem glasovanju. Entiteta Sodelujoči vsebuje spodnje atribute:

1. **ID glasovanja** – Enolično identificira glasovanje.
2. **ID uporabnika** – Enolično identificira uporabnika.
3. **ID izbire** – Enolično identificira izbiro.
4. **Čas** – Čas oddaje glasu.

Pristop k zasebnim glasovanjem zahteva dovoljenje, ki ga predstavlja entiteta Dovoljenje. Posamezno dovoljenje je v relaciji mnogo proti ena z entiteto Glasovanje in Uporabnik. Entiteto predstavljajo atributi:

1. **ID glasovanje** – Enolično identificira glasovanje.
2. **ID uporabnik** – Enolično identificira uporabnika.

3. Dovoljenje – Vrednost dovoljenja [dovoli, zavrne].

Dovoljenja lahko delimo tudi skupinam uporabnikov. Da lahko to storimo, moramo najprej ustvariti skupino uporabnikov, nato pa pri zasebnem glasovanju poljubni skupini uporabnikov izdamo dovoljenje. Takšna funkcionalnost zahteva dve novi entiteti. Prva entiteta se imenuje Skupina in jo definirajo naslednji atributi:

1. **Ime** – Ime skupine.
2. **Uporabniki** – Seznam uporabnikov.

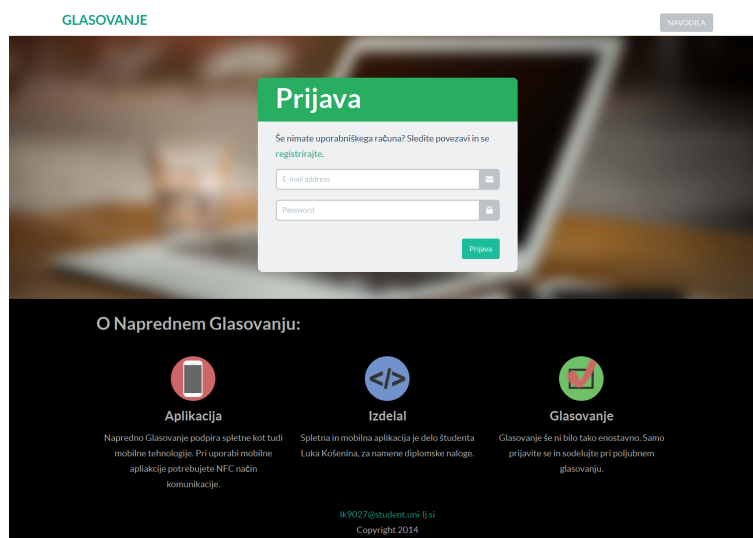
Druga entiteta je Skupinsko dovoljenje, ki tvori relacijo mnogo proti ena z entiteto Skupina in Glasovanje. Poleg relacij vsebuje še dodaten atribut dovoljenje:

1. **ID glasovanje** – Enolično identificira glasovanje.
2. **ID skupina** – Enolično identificira skupino uporabnikov.
3. **Dovoljenje** – Vrednost dovoljenja [dovoli, zavrne].

3.1.2 Funkcionalnosti

Uporabniki imajo na voljo veliko funkcionalnosti s katerimi lahko pregledujejo glasovanja, jih ustvarjajo, urejajo, ter kot najpomembneje glasujejo. Ob uporabi spletne aplikacije se prikaže uporabniški vmesnik, kjer se uporabnik lahko prijavi. Prijavi se lahko vsak registriran uporabnik, ne registrirani uporabniki se lahko registrirajo z ustrezno izpolnitvijo podanega obrazca. Na tem delu aplikacije so vsem obiskovalcem na voljo tudi informacije in navodila o uporabi spletne kot tudi mobilne aplikacije. Uporabniki imajo na voljo tudi prenos mobilne aplikacije, katero lahko namestijo na Android mobilno napravo.

Spletna aplikacija je prilagojena za uporabo na zaslonih vseh ločljivosti, vsebina prikaza se tako prilagaja glede na širino zaslona naprave. Ob uporabi smo pričra odlični izkušnji, neglede na to ali uporabljamo mobilni telefon, tablični ali namizni računalnik. Odziven dizajn spletne aplikacije je bil nujno potreben, saj jo dopolnjuje mobilna, kjer uporabnik določena opravila lahko opravi le preko spletne aplikacije.

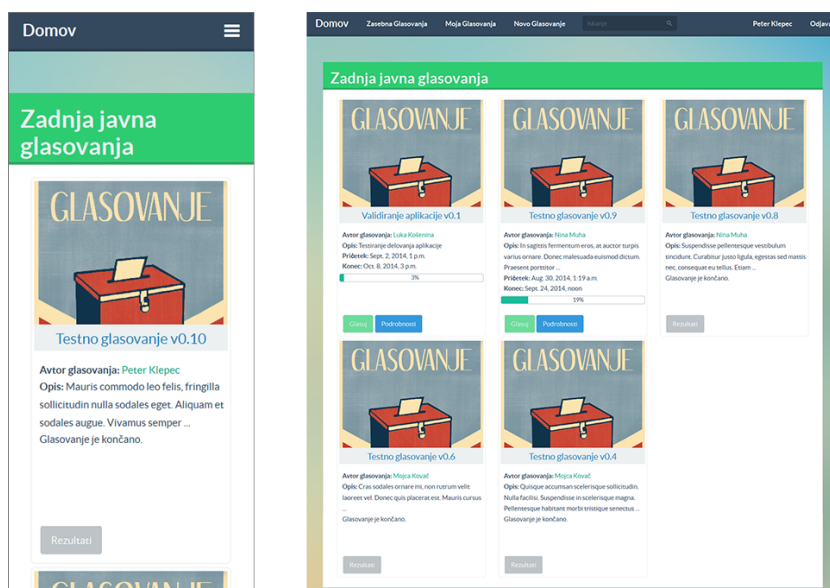


Slika 3.3: Uporabniški vmesnik za prijavo.

Slika 3.3 prikazuje uporabniški vmesnik, ki se pojavi ob prihodu na spletno aplikacijo. Iz slike je razvidna uporaba preprostega in čistega dizajna. Ob prijavi oziroma registraciji je uporabnik nemudoma preusmerjen na domačo stran aplikacije.

3.1.3 Prikaz glasovanj

Domača stran spletne aplikacije, prijavljenemu uporabniku ponudi seznam vseh javnih glasovanj. Podatki o posameznem glasovanju so prikazani v pravokotniku, ki vsebuje le najpomembnejše informacije. Obiskovalec ima pri vsakem glasovanju na voljo povezave do podrobnosti glasovanja, oddaje glasu ter pregledovanja rezultatov glasovanja. Glasovanja v poteku imajo na vseh straneh pregledovanja glasovanj podano vrstico napredka. Le ta prikazuje preostanek časa, dokler je še dovoljeno glasovanje. Slika 3.4 prikazuje prilagodljiv prikaz seznama glasovanj na mobilni napravi in osebem računalniku. Na sliki lahko opazimo, da stran ne vsebuje oštevilčenja. Le to tudi ni potrebno zaradi dinamičnega dodajanja preostalih glasovanj, ko uporabnik preide na konec seznama. Takšen način pridobivanja podatkov o glasovanjih razbremenjuje podatkovno bazo in omogoča hitrejše in bolj



Slika 3.4: Prikaz seznama glasovanj na mobilnem telefonu in prenosnem računalniku.

gladko delovanje na strani odjemalca.

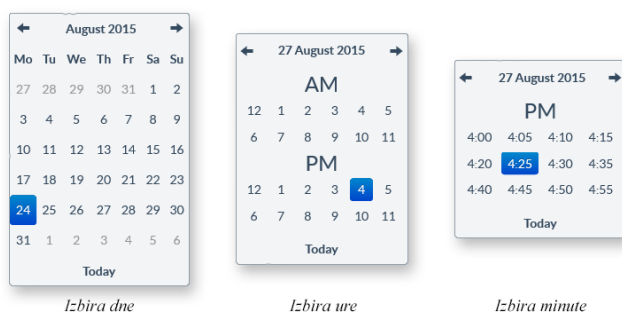
Uporabniki lahko poleg javnih glasovanj pregledujejo tudi zasebna glasovanja za katera so prejeli dovoljenja. Za prikaz navedenih glasovanj je potrebno v menijski vrstici izbrati gumb Zasebna glasovanja. Ob izbiri se prikaže seznam, ki je enake oblike kot seznam javnih glasovanj, le vsebina je druga. Skrbniki imajo v menijski vrstici poleg gumbov Domov in Zasebna glasovanja na voljo še dve dodatni izbiri. Prva izbira, Moja glasovanja, prikaže vsa skrbnikova glasovanja. Ta pregled je razdeljen na tri razdelke glede na stanja glasovanj. Ker smo že spoznali ta tri stanja, potem vemo, da se pregled deli na glasovanja ki se še niso začela, tekoča in zaključena glasovanja. Za lažjo predstavbo zgleđa si oglejmo spodnjo sliko 3.5.

Razdelek, ki ne vsebuje nobenega glasovanja prikaže obvestilo o prazni listi glasovanj. Posamezen razdelek lahko na tej strani prikaže največ šest glasovanj, v primeru večjega števila se prikaže dodaten gumb za preusmeritev na stran, ki prikazuje vsa tovrstna glasovanja.

The screenshot displays a web interface for a voting system, organized into three main sections:

- Glasovanja, ki se še niso pričela** (Voting events that have not yet started): A blue header with a light blue message box stating "Trenutno nimate nimate nobenega glasovanja, ki se še ni pričelo." (You currently have no voting events that have not yet started).
- Glasovanja v poteku** (Voting events in progress): A green header containing two active voting event cards:
 - Testno glasovanje v0.13**: Includes a description, end time (Oct. 23, 2014, noon), public status (Ne), and anonymity status (Ne). A progress bar shows 3% completion. A "Možnosti" button is visible.
 - Testno glasovanje v0.11**: Includes a description, end time (Sept. 10, 2014, 1 p.m.), public status (Ne), and anonymity status (Da). A progress bar shows 63% completion. A "Možnosti" button is visible.
- Zaključena glasovanja** (Completed voting events): A red header containing one completed voting event card:
 - Testno glasovanje v0.10**: Includes a description, public status (Da), and anonymity status (Ne). A "Rezultati" button is visible.

Slika 3.5: Prikaz seznama glasovanj skrbnika.



Slika 3.6: Prikaz korakov izbire datuma.

3.1.4 Ustvarjanje glasovanj

Ustvarjanje glasovanj je za skrbnika zelo preprost postopek. Novo glasovanje je mogoče ustvariti v najmanj dveh korakih, kjer lahko drugi korak ponovimo večkrat. V drugem koraku namreč skrbnik dodaja novo vprašanje in njegove izbire.

Postopek ustvarjanja novega glasovanja pričnemo s pritiskom na gumb Novo Glasovanje v navigacijski vrstici. Ob tem dogodku uporabnik prejme obrazec, ki ga mora izpolniti za kreiranje novega glasovanja. Podati mora podatke, kot so naslov, opis, pričetek, konec, opomba glasovanja ter označiti zelene lastnosti. Med lastnostmi glasovanja je ponujena izbira med javnim in zasebnim glasovanjem, anonimnim in osebnim ter med možnostjo oddaje glasu avtorja glasovanja. Pri vnosu časa pričetka in konca glasovanja moramo biti pozorni, da ima čas pričetka manjšo vrednost kot čas konca. V ta namen je na voljo poseben pripomoček, kjer v treh korakih pripravimo zeleno vrednost. V prvem koraku izberemo datum, tukaj lahko izbiramo med današnjim datumom in med tistimi v prihodnosti. Drugi korak nam ponudi možnost izbire ure, tukaj lahko izberemo poljubno vrednost. V zaključku pa le še določimo minute, na voljo imamo izbire s pet minutnim presledkom. Slika 3.6 prikazuje uporabniški vmesnik posameznega koraka.

Ob pravilno izpolnjenem obrazcu lahko nadaljujemo ustvarjanje novega glasovanja s pritiskom na gumb Ustvari in dodaj vprašanja. S tem manevrom preidemo na drugi korak, kjer ustvarjamo poljubno število vprašanj in njegovih izbir. Vprašanje in izbire ustvarimo s pravilno izpolnjenim obrazcem, ki ga prikazuje slika 3.7.

The image shows a web form for creating a new question. It has a green header 'Novo vprašanje'. Below it is a text input field for the question, a dropdown menu for the question type (currently 'Možen odgovor'), and a checkbox for 'Neobvezno'. The second section is 'Izbire', which has two text input fields for options, each with a yellow 'Izbriši' button below it. A blue 'Dodaj novo izbiro' button is at the bottom of the options section. At the very bottom of the form, there are two green buttons: 'Kreiraj' and 'Kreiraj in ustvari novo'.

Slika 3.7: Uporabniški vmesnik ustvarjanja novega glasovanja.

Obrazec je razdeljen na dva dela, v prvem podamo podatke o vprašanju, v drugem pa dodajamo izbire. Največje število dovoljenih izbir je dvanajst, če želimo dodati novo izbiro to storimo s pritiskom na gumb Dodaj novo izbiro. Posamezno izbiro lahko tudi odstranimo s pritiskom na gumb Izbriši. Ob koncu kreiranja vprašanja in izbir imamo dve možnosti za nadaljevanje ustvarjanja. S pritiskom na gumb Kreiraj se bo ustvarilo vprašanje z zgornjimi podatki in strežnik nas bo preusmeril na seznam mojih vprašanj. Alternativna možnost je ustvariti vprašanje in ponovno izpolniti obrazec za ustvarjanje vprašanja. To pot uberemo s pritiskom na gumb Kreiraj in ustvari novo. Kreiranje glasovanja zaključimo s pritiskom na gumb Kreiraj oziroma če smo ustvarili že vsaj eno vprašanje lahko to storimo tudi s pritiskom na gumb Prekliči.

3.1.5 Urejanje glasovanj

V razdelku glasovanj, ki se še niso pričela skrbnik lahko posamezno glasovanje ureja in ga predčasno odpre. Ob kliku na gumb Uredi ureja podatke glasovanja. Na voljo ima urejanje lastnosti glasovanja, kot so naslov, opis, čas pričetka itd.

Pri urejanju naštetih lastnosti moramo biti pozorni, da njihovo vrednost pravilno urejamo. Čas pričetka mora biti manjši od časa konca, vsa ostala vnosna polja morajo biti izpolnjena razen polja opomba, ki je opsijsko. Če želimo lahko urejamo tudi posamezno vprašanje in njegove izbire. Dokler glasovanje vsebuje najmanj eno vprašanje lahko skrbnik vprašanja tudi briše. Poleg brisanja vprašanj lahko tudi dodajamo nova vprašanja s preprostim pritiskom na gumb Dodaj novo vprašanje, ki nam odpre ustrezen obrazec. Slika 3.8 prikazuje zgled urejanja glasovanja.

Pri vsakem vprašanju lahko urejamo tudi njegove izbire. Pri tem moramo biti pozorni le, da vsako vprašanje vsebuje najmanj dve izbiri. Izbire lahko tako kot vprašanja brišemo, dodajamo in jim urejamo vsebino. Za lažjo predstavo urejanja je podana spodnja slika 3.9.

Vsi dogodki se na tej strani izvedejo s pomočjo tehnologije AJAX. Podatki se prenašajo v obliki objektov tipa JSON. Posledica tovrstne tehnologije je prijetnejše izvajanje opravil na strani klienta, saj uporabnik ni deležen nepotrebnih osveževanj strani.

3.1.6 Iskanje

Dandanes je že skoraj privzeto, da vsaka podatkovno obsežnejša spletna aplikacija nudi iskanje. Glasovalni sistem NFC ni izjema in vsebuje preprosto iskanje po glasovanjih in uporabnikih. Polje kjer uporabnik vnese iskalni niz se nahaja v izbirni vrstici. Zadetki iskanja so prikazani v minimalistični obliki in so postavljeni pod vnosno polje. Ob kliku na zadetek iskanja je uporabnik preusmerjen na izbrano vsebino, v našem primeru je to lahko glasovanje ali pa osebni podatki uporabnika in njegova glasovanja v primeru statusa skrbnika.

Iskalni algoritem je implementiran na preprost način. Za vneseno besedilo na strani strežnika preverimo ali je prisotno v katerem izmed naslednjih polj: naslov glasovanja, opis glasovanja, ime uporabnika, priimek uporabnika in E-mail naslov uporabnika. Strežnik ob uspešnem iskanju ustvari JSON objekt in ga posreduje na stran odjemalca. Posredovano sporočilo je zapisano v naslednjem formatu:

```
1 [{  
2   "data": [{  
3     "naslov": "Testno glasovanje v0.8",  
4     "opis": "Suspendisse pellentesque vesti..."
```

Podatki Glasovanja

Naslov

Opis

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Aenean mattis mauris sed odio eleifend convallis. Curabitur condimentum bibendum efficitur. Suspendisse euismod luctus cursus. Nam sit amet varius tortor. Donec sed sagittis ante, vel tempor urna.

Pričetek

 ✕ ⋮

Konec

 ✕ ⋮

Opomba

Udeležba skrbnika glasovanja pri glasovanju

Anonimno

Javno

[Uredi](#) [Prekliči](#) [Izbriši](#)

Vprašanja

Vprašanje	Akcija
Class aptent taciti sociosqu ad litora torquent per conubia nostra?	Uredi Izbriši
Class aptent taciti sociosqu ad litora torquent?	Uredi Izbriši

[Dodaj novo vprašanje](#)

Slika 3.8: Prikaz urejanja glasovanja.

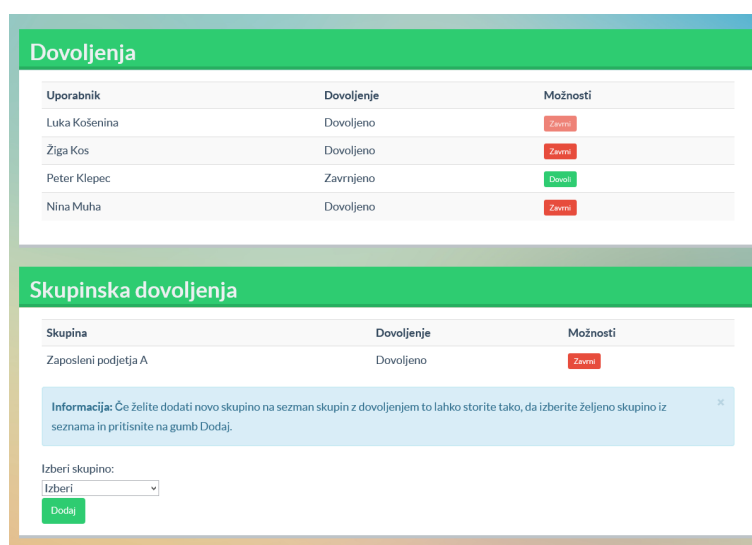
The screenshot displays a web application interface with three main sections:

- Vprašanje:** A red header bar with the title "Vprašanje". Below it, the text "Vsebina: Class aptent taciti sociosqu ad litora torquent per conubia nostra?" is shown, followed by a yellow "Uredi" button.
- Izbire:** A red header bar with the title "Izbire". Below it, there are two columns: "Izbira" and "Akcija". The "Izbira" column contains two text inputs with placeholder text: "Vestibulum molestie dapibus iaculis." and "Nulla vel porttitor ligula, nec blandit turpis. Sed sodales id nulla vitae semper." The "Akcija" column contains three buttons: "Uredi", "Izbiši", and "Prekliči", "Posodobiti", and "Izbiši".
- Dodaj novo izbiro:** A red header bar with the title "Dodaj novo izbiro". Below it, the text "Vsebina:" is followed by a text input field and a green "Dodaj" button.

Slika 3.9: Urejanja vprašanja in izbir.

```
5     "id": 9
6   }],
7   "title": "votes"
8 },
9 {
10  "data": [{
11    "first_name": "Spela",
12    "last_name": "Novak",
13    "email": "spela@gmail.com",
14    "id": 2
15  }],
16  "title": "users"
17 }]
```

JSON objekt hrani dvoje vrst podatkov, v prvem zapisu so shranjeni podatki o glasovanjih, v drugem pa podatki o uporabnikih.



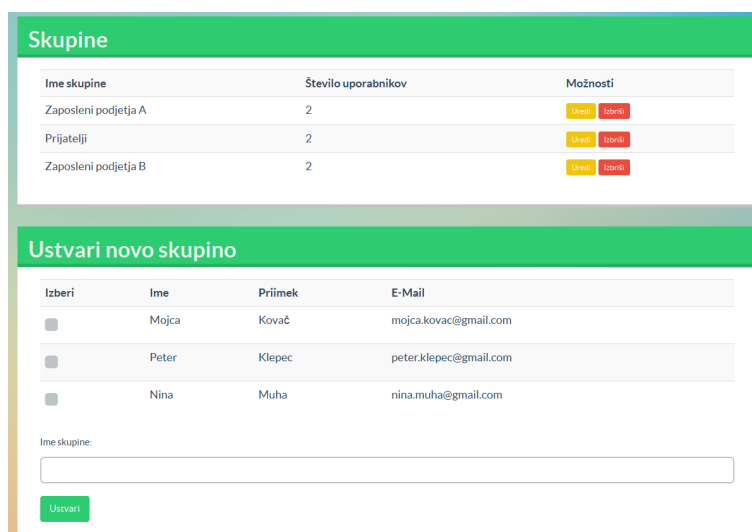
Slika 3.10: Opravljanje z dovoljenji zasebnega glasovanja.

3.1.7 Izdaja dovoljenj

Izdaja dovoljenj sicer spada pod funkcionalnosti mobilne aplikacije. Spletna aplikacija podpira upravljanje z dovoljenji in dodajanje skupinskih dovoljenj. Skrbnik zasebnega glasovanja ima tako možnost pregledovanja izdanih dovoljenj posameznega uporabnika ali celotne skupine uporabnikov in urejanja njihovih vrednosti. Posamezno dovoljenje je lahko bodisi zavrjeno ali dovoljeno. Zgled uporabniškega vmesnika za opravljanje z dovoljenji prikazuje slika 3.10.

Skrbnik ima možnost ustvarjanja skupine uporabnikov, ki jim je že izdal dovoljenje za neko lastno zasebno glasovanje. Novo skupino ustvari tako, da iz seznama uporabnikov z že izdanim dovoljenjem izbere ustrezne uporabnike in poda ime skupine. Ob uspešnem dodajanju se ravnokar ustvarjena skupina vključi v seznam definiranih skupin skrbnika. Posamezno skupino lahko skrbnik ureja in po potrebi tudi izbriše. Urejanje skupine podpira dodajanje novih uporabnikov na seznam kot tudi brisanje obstoječih uporabnikov iz seznama. Po želji lahko spremenimo tudi ime skupine. Zgled ustvarjanja skupine ponazarja slika 3.11.

Izdaja skupinskih dovoljenj pride prav ob izvajanju več zaporednih zasebnih glasovanj z istimi sodelujočimi uporabniki. Najlažje si to lahko predstavljamo na konkretnem primeru: ustvarimo zasebno glasovanje, za katerega izdamo dovolje-



Slika 3.11: Uporabniški vmesnik kreiranja skupine uporabnikov.

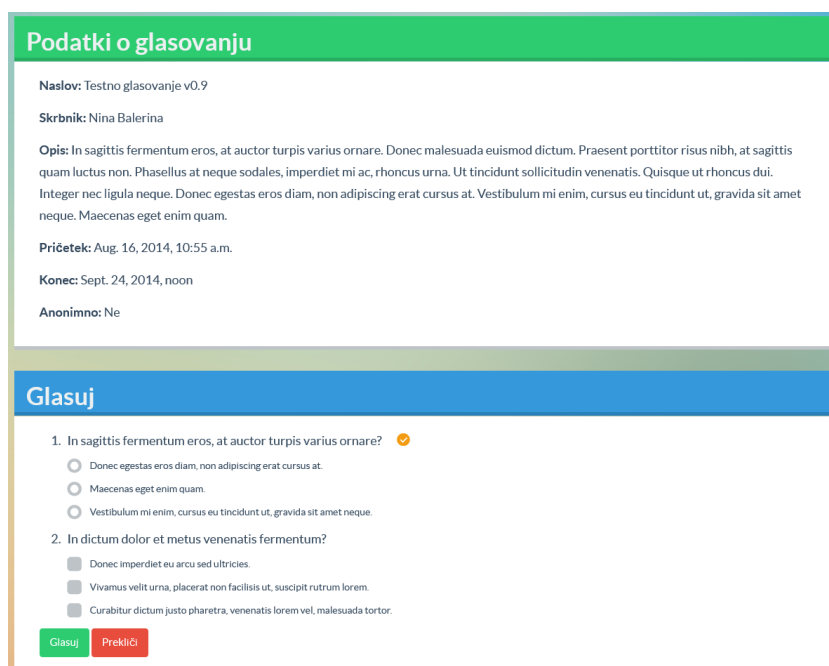
nja desetim osebam. Kasneje želimo izvesti ponovno zasebno glasovanje z istimi uporabniki. Da se izognemo ponovnemu identificiranju in dodeljevanju dovoljenj vsakemu sodelujočemu uporabniku posebej, enostavno izdamo skupinsko dovoljenje, kjer skupino sestavljajo uporabniki prejšnjega glasovanja.

3.1.8 Glasovanje

Glavni namen aplikacije je glasovanje in temu ustrezno je v tem razdelku podrobneje opisan potek oddaje glasu. H glasovanju lahko pristopimo po dveh različnih poteh. K javnim glasovanjem lahko pristopijo vsi prijavljeni uporabniki, k zasebnim pa le tisti, ki so pred tem prejeli dovoljenje.

Ob pristopu h glasovanju so na vrhu strani napisana navodila in obrazložitev označbe obveznih glasovanj. Vsako glasovanje vsebuje najmanj eno obvezno vprašanje na katerega mora uporabnik nujno odgovoriti če želi uspešno oddati svoj glas. Vsa ostala vprašanja so bodisi obvezna ali neobvezna. Obvezna vprašanja so označena z rumeno kljukico, ki je na koncu vsebine vprašanja. Slika 3.12 prikazuje neizpolnjen obrazec glasovanja.

Trenutna verzija aplikacije podpira dva tipa vprašanj, takšna kjer je možno izbrati le eno izbiro in takšna, kjer lahko označimo več izbir. Posamezno izbiro



Podatki o glasovanju

Naslov: Testno glasovanje v0.9

Skrbnik: Nina Balerina

Opis: In sagittis fermentum eros, at auctor turpis varius ornare. Donec malesuada euismod dictum. Praesent porttitor risus nibh, at sagittis quam luctus non. Phasellus at neque sodales, imperdiet mi ac, rhoncus urna. Ut tincidunt sollicitudin venenatis. Quisque ut rhoncus dui. Integer nec ligula neque. Donec egestas eros diam, non adipiscing erat cursus at. Vestibulum mi enim, cursus eu tincidunt ut, gravida sit amet neque. Maecenas eget enim quam.

Pričetek: Aug. 16, 2014, 10:55 a.m.

Konec: Sept. 24, 2014, noon

Anonimno: Ne

Glasuj

1. In sagittis fermentum eros, at auctor turpis varius ornare? 👉

- Donec egestas eros diam, non adipiscing erat cursus at.
- Maecenas eget enim quam.
- Vestibulum mi enim, cursus eu tincidunt ut, gravida sit amet neque.

2. In dictum dolor et metus venenatis fermentum?

- Donec imperdiet eu arcu sed ultricies.
- Vivamus velit urna, placerat non facilisis ut, suscipit rutrum lorem.
- Curabitur dictum justo pharetra, venenatis lorem vel, malesuada tortor.

Glasuj Prekliči

Slika 3.12: Obrazec glasovanja.

določimo s klikom na njeno besedilo oziroma ikono postavljeno pred vsebino izbire. Izbrana izbira se za namen preglednosti obarva v zeleno barvo. Ob ustrezno izpolnjenem glasovanju je potrebno za oddajo glasu pritisniti gumb Glasuj. Takrat se v ozadju preveri pravilnost izpolnjenosti glasovanja. Glasovanje je pravilno izpolnjeno če vsa obvezna vprašanja vsebujejo izbrano izbiro. V primeru nepravilno izpolnjenega glasovanja se pojavi obvestilo, ki uporabnika opozori o napaki. Prav tako se ob uspešno izpolnjenem glasovanju pri oddaji glasu prikaže pojavno okno, ki uporabnika opozori, da je oddaja glasu enkratna in je ni mogoče kasneje spremenjati. V primeru, da uporabnik želi zapustiti glasovanje lahko to stori s pritiskom na gumb Prekliči. Takrat se izbire ne shranijo in ko uporabnik naslednjič pristopi h glasovanju prejme prazno, neizpolnjeno glasovanje.

Programerska plat poteka glasovanja je nekoliko bolj zanimiva zaradi izvedbe raznih preverjanj in izmenjave podatkov. Ob prejtem zahtevku za prikaz obrazca glasovanja se najprej preveri ali je uporabnik, ki zahteva pristop h glasovanju že oddal svoj glas. Tovrstno preverjanje vrne rezultat tipa boolean, če je njegova vrednost False potem ima uporabnik dovoljen prikaz glasovanja. V nasprotnem pri-

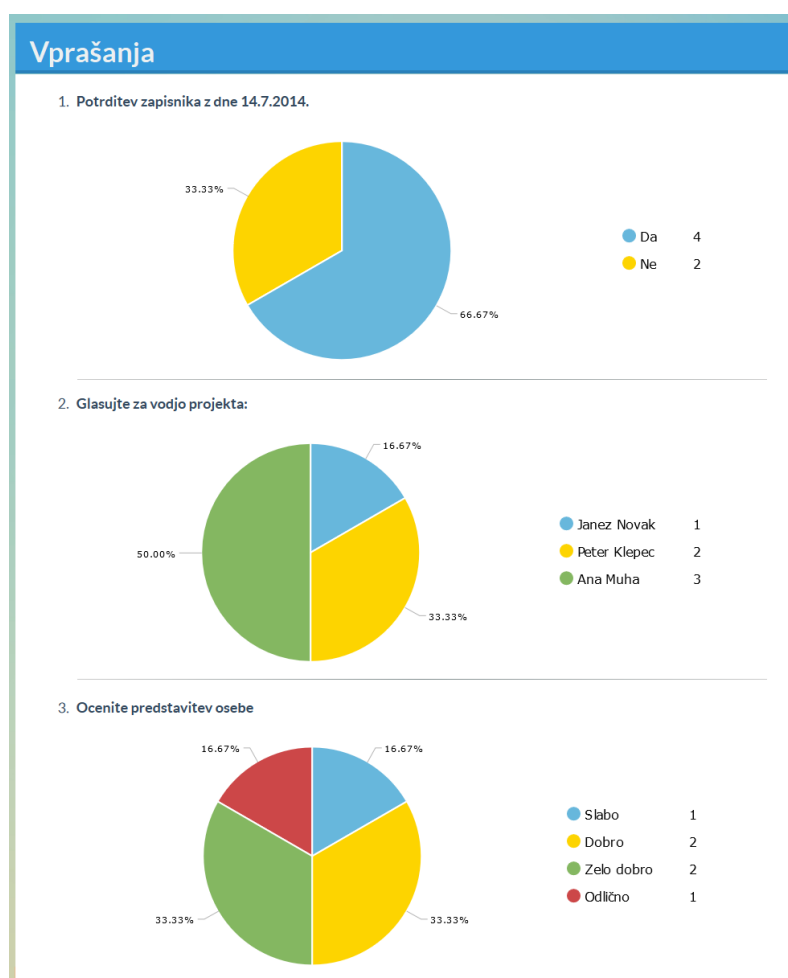
meru uporabniku ne smemo prikazati obrazca za glasovanje in ga ob tej priložnosti tudi ustrezno obvestimo. Zasebna glasovanja ob zahtevku potrebujejo še dodatno preverjanje ali ima uporabnik ustrezno dovoljenje za pristop h glasovanju. Rezultat preverjanja je izdelan po podobnem kopitu kot prej opisano preverjanje. V naslednjem koraku pripravimo potrebne podatke za pripravo glasovalnega obrazca. Podatki se s strani strežnika na stran odjemalca pošljejo v posebnem HTTP odgovoru. Iz prejetih podatkov se nato dinamično pripravi glasovalni obrazec. Vprašanja in izbire morajo biti enolično označene znotraj obrazca, da lahko kasneje pravilno beležimo izbrane izbire posameznega vprašanja. Ob oddaji glasovanja se ponovno prenesejo podatki s strani klienta na strežnik. Podatki so pripravljene v formatu JSON in vsebujejo ključne `vprasanje_id`, `izbira_id`. Pri vprašanjih, ki dovoljujejo več možnih izbranih izbir se temu ustrezno pošljejo podatki, ki imajo enako vrednost ključa `glasovanje_id` a različno vrednost ključa `izbira_id`. Na strani strežnika iteriramo čez prejete podatke in zapisujemo v model sodelujočih uporabnikov ustrezne podatke. Prav tako pri vsaki izbrani izbiri povečamo števec, kateri beleži, kolikokrat je bila izbrana. V zadnjem koraku zaključimo proces glasovanja in uporabnika preusmerimo na domačo stran.

3.1.9 Prikaz rezultatov

Prikaz rezultatov se razlikuje glede na to ali do njih dostopa avtor glasovanja ali ostali uporabniki. Avtor glasovanja ima na voljo vpogled rezultatov tekom izvajanja glasovanja, med tem, ko ostali uporabniki lahko pregledujejo le rezultate zaključenega glasovanja.

Neuradni rezultati, to so rezultati glasovanja v poteku, so dostopni takoj, ko glasovanje preide v stanje poteka. Prikaz rezultatov je odvisen tudi ali je glasovanje anonimno ali ne. Pri anonimnem glasovanju skrbnik ne vidi rezultata prvega sodelujočega, saj bi v tem primeru lahko razbral, kdo je oddal prvi glas in bi posledično izvedel njegovo izbiro. V takšnem primeru in v primeru še nobenega oddanega glasu skrbnik prejme obvestilo, zakaj rezultati niso prikazani.

Rezultati so predstavljeni v obliki tortnih diagramov in legende. Vsaka izbira je predstavljena s posamezno barvo in z odstotnim deležem. Ob kliku na posamezno izbiro se prikaže njena celotna vsebina, če je le ta predolga za celoten prikaz v legendi, in število, kolikokrat je bila izbrana. Za boljšo predstavbo prikaza rezultatov



Slika 3.13: Tortni prikaz rezultatov glasovanja.

je podana slika 3.13.

Pri glasovanjih, ki niso anonimna, ima skrbnik na voljo pregled oddaje glasov za vsakega uporabnika. Ti podatki so prikazani v obliki razpredelbnice in so ločeni glede na uporabnika, na ravni nižje so pa ločeni glede na vprašanje (slika 3.14).

3.1.10 Urejanje osebnih podatkov

Registrirani uporabnik lahko ureja podatke uporabniškega računa ob pritisku na gumb v menijski vrstici, na katerem je izpisano uporabnikovo ime in priimek.



Uporabnik	Vprašanje	Izbira
Peter Klepec	Potrditev zapisnika z dne 14.7.2014.	Da
	Glasujte za vodjo projekta	Ana Muha
	Ocenite predstavitev osebe A	Odlično
Nina Muha	Potrditev zapisnika z dne 14.7.2014.	Da
	Glasujte za vodjo projekta	Janez Novak
	Ocenite predstavitev osebe A	Slabo

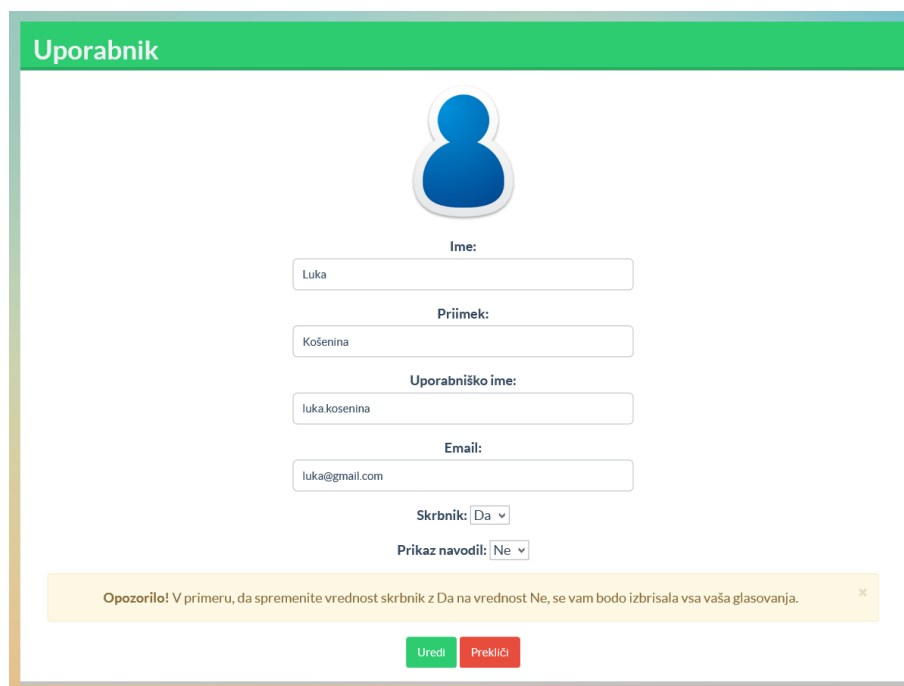
Slika 3.14: Prikaz seznama sodelujočih.

Prikaže se pogled na katerem so prikazani podatki uporabnika. Če želi urejati prikazane podatke, to lahko stori s pritiskom na gumb Urejaj podatke. Za vnos spremembe podatkov uporabniškega računa je treba spremeniti poljubno vrednost in pritisniti na gumb Uredi. Pri tem je pomembno upoštevati, da sprememba vrednosti polja skrbnik z Da na Ne, izbriše vsa glasovanja, ki jih je uporabnik predtem ustvaril. Navedeno spremembo je potrebno posebej potrditi, da lahko preprečimo pomotoma spremenjeno vrednost. Uporabnik lahko izbira tudi med prikazom navodil v spletni aplikaciji. Če je nastavljena vrednost Da, uporabnik na straneh, kjer je potrebno videti razna navodila ali obvestila. Uporabniški vmesnik urejanja podatkov ponazarja spodnja slika 3.15.

3.2 Mobilna aplikacija

Mobilna aplikacija je izdelana za mobilne naprave, ki jih poganja operacijski sistem Android. Nameščen mora imeti operacijski sistem Android verzije Jelly Bean 4.1.X ali novejšo, ter z vidika strojne opreme, podporo tehnologiji NFC. Naprave, ki ustrezajo navedenim zahtevam lahko namestijo mobilno aplikacijo. Po uspešni namestitvi se mora uporabnik prijaviti ali registrirati, če še nima uporabniškega računa. Zaželeno je uporaba identičnega uporabniškega računa tako za mobilno kot tudi za spletno aplikacijo. Prijava se izvede z ustreznim uporabniškim imenom in geslom. Uporabniški vmesnik registracije in prijave uporabnika sta prikazana na sliki 3.16.

Ob uspešni prijavi ali registraciji, aplikacija preveri ali ima naprava vključeno



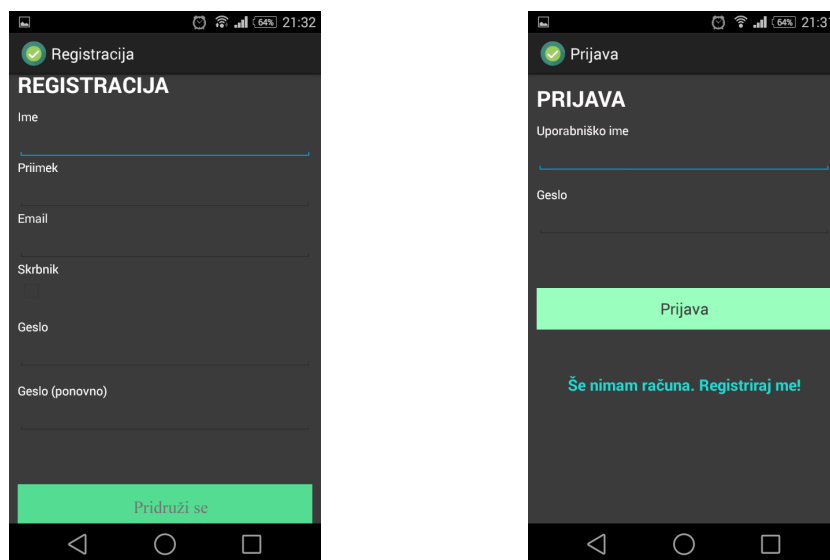
The screenshot shows a web page titled "Uporabnik" (User) with a green header. Below the header is a blue profile icon. The form contains the following fields and controls:

- Ime: Luka
- Priimek: Košenina
- Uporabniško ime: luka.kosenina
- Email: luka@gmail.com
- Skrbnik: Da (dropdown menu)
- Prikaz navodil: Ne (dropdown menu)

A yellow warning box at the bottom states: "Opozorilo! V primeru, da spremenite vrednost skrbnik z Da na vrednost Ne, se vam bodo izbrisala vsa vaša glasovanja." (Warning! In the event that you change the value of the administrator from Yes to No, all your votes will be deleted.)

At the bottom of the form are two buttons: "Uredi" (Edit) in green and "Prekliči" (Cancel) in red.

Slika 3.15: Urejanja podatkov uporabniškega računa.



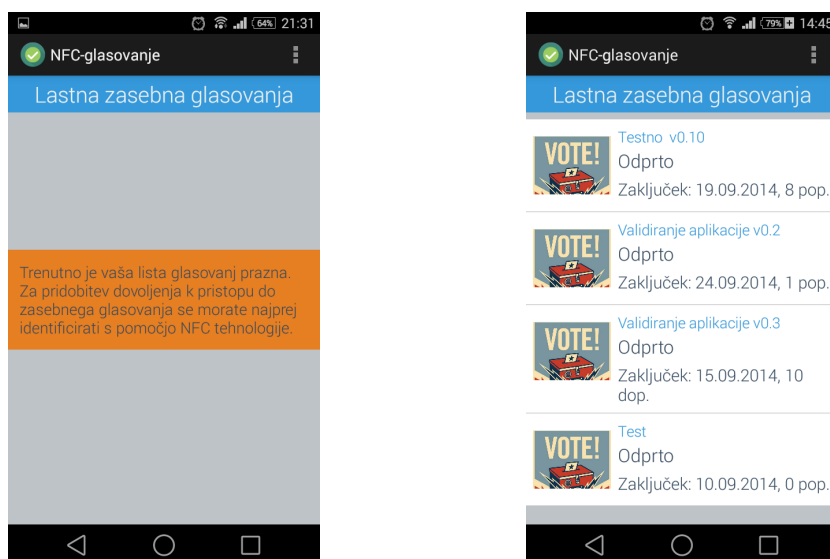
The image shows two screenshots of a mobile application interface. The left screenshot is the registration screen, titled "Registracija" (Registration), with a green checkmark icon. It contains the following fields and controls:

- Ime (Name)
- Priimek (Surname)
- Email
- Skrbnik (Administrator)
- Geslo (Password)
- Geslo (ponovno) (Password (repeat))
- A green "Pridruži se" (Join) button at the bottom.

The right screenshot is the login screen, titled "Prijava" (Login), with a green checkmark icon. It contains the following fields and controls:

- Uporabniško ime (Username)
- Geslo (Password)
- A green "Prijava" (Login) button.
- A link "Še nimam računa. Registriraj me!" (I don't have an account. Register me!) in blue text.

Slika 3.16: Uporabniški vmesnik registracije in prijave uporabnika.

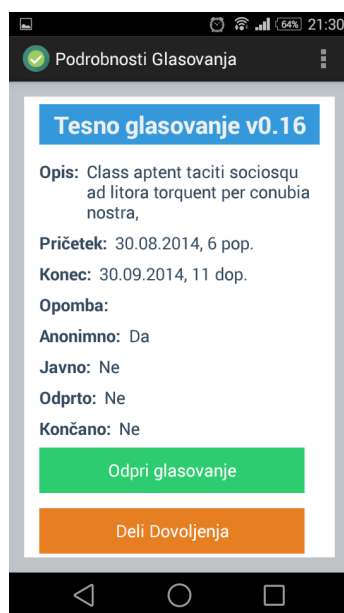


Slika 3.17: Prikaz praznega in zapolnjenega seznama glasovanj.

internetno povezavo in NFC nastavitve. Ta dva pogoja morata biti izpolnjena, saj drugače aplikacija ne deluje pravilno. V primeru neizpolnjenih zahtev uporabnik prejme opozorilo. Opozorilo uporabniku ponudi izhod iz aplikacije oziroma bližnjico do nastavitve, kjer lahko ustrezno nastavi potrebne nastavitve. Vstopna točka aplikacije je seznam zasebnih glasovanj. Seznam je bodisi prazen bodisi vsebuje glasovanja, za katere smo prejeli dovoljenja za oddajo glasu. Slika 3.17 prikazuje zgled seznama v primeru praznega in zapolnjenega seznama.

Izbira glasovanja s seznama odpre nov pogled z vsemi opisi in možnostmi glasovanja. Glede na lastnosti glasovanja imamo na voljo tudi gumb s povezavo na spletno aplikacijo, kjer lahko oddamo svoj glas. Skrbniki imajo na voljo še dodatno izbiro, s katero omogočijo oziroma onemogočijo deljenje dovoljenj. S pritiskom na gumb Odpri glasovanje pa lahko odprejo še zaprta glasovanja. Zgled pogleda podrobnosti glasovanja prikazuje spodnja slika 3.18.

Proženje gumba Deli Dovoljenja omogoča ostalim uporabnikom prejemanje dovoljenj. Uporabnik, ki želi prejeti dovoljenje od skrbnika oziroma avtorja glasovanja mora le prisloniti svojo napravo ob napravo skrbnika. Takrat se pojavi uporabniški vmesnik tehnologije Android Beam. S preprostim pritiskom na zaslon naprave uporabnik pošlje zahtevek na stran skrbnika. Ob uspešni izvedbi zahtevka



Slika 3.18: Prikaz podrobnosti glasovanja.

uporabnik prejme novo dovoljenje na listi Zasebnih glasovanj, skrbnik pa obvestilo o uspehu. V nasprotnem primeru uporabnik ne prejme novega dovoljenja, skrbnik pa prejme obvestilo o vzroku za neuspešno izvedbo zahteve.

Mobilna aplikacija ne implementira vseh funkcionalnosti spletne aplikacije ampak jo le dopolnjuje. A to še ne pomeni, da ne moremo uporabljati Glasovalnega sistema NFC v celoti na mobilnih napravah. Nepokrite funkcionalnosti s strani mobilne aplikacije lahko izvajamo s pomočjo spletnega brskalnika. Spletno aplikacijo lahko odpremo preko vnosa URL naslova v spletni brskalnik ali še lažje, v meni vrstici pritisnemo na gumb Odpri v brskalniku. Na ta način se izognemo prijavi saj to namesto nas stori že mobilna aplikacija.

3.2.1 Tehnični opis funkcionalnosti

Do sedaj smo izvedeli kaj vse lahko počnemo s pomočjo mobilne aplikacije in česar ne moremo. V tem razdelku se bomo osredotočili na delovanje zgoraj opisanih funkcionalnosti. Celotno mobilno aplikacijo sestavlja programska koda napisana v programskem jeziku Java. Izbran programski jezik podpira vse potrebne funkcio-

nalnosti, ki jih zahteva uporaba mobilne aplikacije. Mobilna in spletna aplikacija med seboj izmenjujeta veliko sporočil in podatkov. Vsi izmenjujoči se podatki so zapisani v lahkem izmenjevalnem formatu JSON. Ob vsakem prejetem odgovoru s strani strežnika je zapisano ali se je zahtevek uspešno obdelal. Neuspešno izveden zahtevek vrne odgovor, kjer je podano obvestilo o vzroku napake. Zahtevki se s strani mobilne aplikacije na stran strežnika pošiljajo v obliki GET in POST metod. Strežniški API implementira funkcionalnosti za prijavo oziroma registracijo uporabnika, pridobitev seznama zasebnih glasovanj, lastnih zasebnih glasovanj, podatkov o glasovanju, odprtje glasovanja in dodajanje dovoljenj.

3.2.2 Prijava in registracija

Prijava je nujno potrebna za uporabo mobilne aplikacije, med tem, ko je registracija le dodana vrednost aplikacije. Prijavimo se enostavno z vnosom uporabniškega imena in gesla. Zaradi varnostnih razlogov se vrednost gesla pred pošiljanjem zahtevka zakodira. Aplikaciji uporabljata AES kodiranje in dekodiranje s 128 bitnim ključem. Pripravljeni podatki se nato pošljejo preko POST zahtevka na strežnik. Strežnik preveri ustreznost prejetih podatkov ter poskusi prijaviti uporabnika. Ob uspešni prijavi strežnik odgovori z vsemi podatki o uporabniku. Spodnji odsek prikazuje obliko prejetega sporočila.

```
1  [{
2    "tag": "login"
3  }, {
4    "success": 1
5  }, {
6    "error": 0
7  }, {
8    "user": [{
9      "username": "luka"
10     }, {
11      "email": "luka@gmail.com"
12     }, {
13      "skrbnik": true
14     }, {
```

```
15     "first_name": "Luka "
16     }, {
17     "last_name": "Kosenina"
18     }, {
19     "password": "D4SDA1euUkaEK+XwyrCjQw=="
20   }
21 ]]
```

JSON format definirajo ključi in vrednosti. V zgornjem primeru ima ključ “tag” vrednost “login” in določa, da se prejet odgovor nanaša na zahtevek za prijavo uporabnika. Ključ “success” nosi vrednost ali je bil poslan zahtevek izveden uspešno ali ne. Uspešno izveden zahtevek nosi vrednost 1, neuspešen pa 0. Naslednji ključ “error” predstavlja podatek o tem ali je prišlo do kakršne koli napake pri obdelavi zahtevka, če je njegova vrednost različna od nič je prišlo do napake. Zgornji trije ključi so prisotni v vsakem poslanem odgovoru, v primeru napake je dodan še ključ “error_msg” z vrednostjo opisa napake. Odgovor na prijavo nam tako vrne poleg omenjenih vrednosti še podatke v uporabniku, ki so pripeti na ključ “user”.

Identičen format odgovora prejmemo tudi pri obdelavi registracije uporabnika. Prejete podatke nato shranimo v podatkovno bazo znotraj mobilne aplikacije. Hranjenje podatkov o uporabniku je nujno potrebno za ustrezno delovanje aplikacije in omogoča samodejno prijavo ob zagonu aplikacije.

3.2.3 Pridobivanje liste glasovanj

Vstopna točka po prijavi, je prikaz zasebnih glasovanj. Listo glasovanj ustvarimo iz prejetih podatkov glede na poslan zahtevek. Zahtevek se izvede z uporabo metode GET, ki nosi atributa uporabniško ime in geslo uporabnika. Navedena atributa je potrebno posredovati ob vsaki zahtevi, saj uporaba strežniškega API zahteva identifikacijo uporabnika. Če identifikacija ne uspe, se zahteva konča z napako, saj je uporaba funkcionalnosti API neidentificiranim uporabnikom prepovedana.

Odgovor na zahtevek za listo glasovanj je posredovan v naslednji obliki:

```
1  [{
2    "tag": "private_votes"
3  }, {
4    "success": 1
5  }, {
6    "error": 0
7  }, {
8    "votes": [{
9      "opomba": "",
10     "odprto": true,
11     "id": 11,
12     "javno": true,
13     "opis": "Mauris commodo leo felis, fringilla
14           sollicitudin nulla sodales eget. Aliquam...",
15     "avtor": [
16       {
17         "first_name": "Janez",
18         "last_name": "Novak",
19         "id": 6
20       }
21     ],
22     "naslov": "Testno glasovanje v0.10",
23     "timedate_start": "2014-08-20 08:00",
24     "timedate_end": "2014-08-30 10:00",
25     "koncano": false,
26     "anonimno": false
27   }, {
28     ...podatki o preostalih glasovanjih...
29   }
30 ]
```

Po istem principu se izvede zahteva za pridobitev lastnih zasebnih glasovanj. Razlika teh dveh zahtevkov je samo v obdelavi zahtevka na strani strežnika. Pri prvem se izvede pridobitev glasovanj, za katere imamo dovoljenje, pri drugem pa

vrnemo zasebna glasovanja, katerih avtor je uporabnik poslanega zahtevka.

3.2.4 Podatki posameznega glasovanja

Izbira glasovanja na eni izmed list glasovanj odpre nov pogled, kjer so podani vsi podatki o glasovanju. Za pridobitev podatkov o glasovanju je treba ponovno poslati zahtevek strežniku. Zahtevku podamo dodaten atribut, ki enolično določa glasovanje. Strežniki API obdela zahtevo in vrne podatke v obliki JSON. Zgled vrnutjenih podatkov si oglejmo v spodnjem odseku.

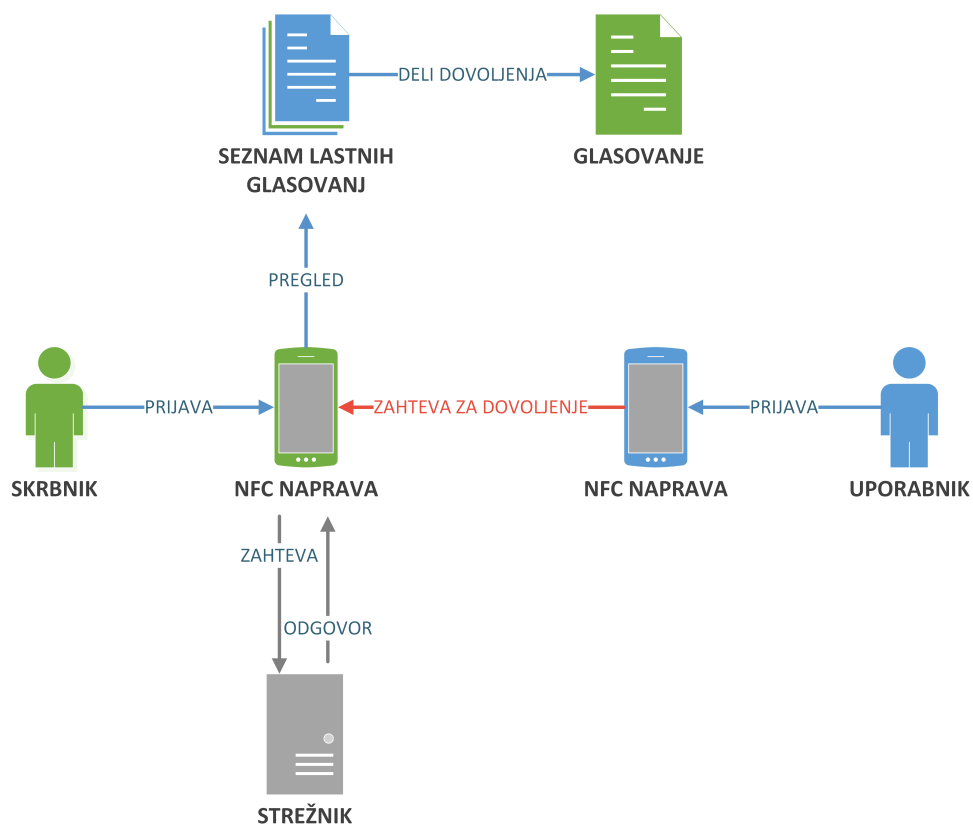
```
1  [{
2    "tag": "vote_details"
3  }, {
4    "success": 1
5  }, {
6    "error": 0
7  }, {
8    "vote": [{
9      "naslov": "Validiranje aplikacije v0.3",
10     "opis": "In ac ipsum quis ante accumsan
11         condimentum. Sed sit amet",
12     "timedate_start": "2014-08-30 13:49:46.316",
13     "timedate_end": "2014-09-15 10:00",
14     "anonimno": false,
15     "id": 20,
16     "odprto": true,
17     "avtor": [{
18       "first_name": "Luka ",
19       "last_name": "Kosenina",
20       "id": 3
21     }]
22   }]
23 }
```

Podan primer prikazuje zelo podobno strukturo podatkov, kot pri odgovoru na zahtevo po pridobitvi liste glasovanj. Razlika je le v tem, da so tukaj navedeni podatki samo za eno glasovanje. Bolj zanimivo pri tej aktivnosti je preverjanje lastnosti glasovanja in ustrezno nastavljanje gumbov. Na voljo imamo naslednje gumbе: Glasuj, Odpri Glasovanje, Deli Dovoljenja in Zaključi Deljenje. Zadnja dva gumba sta na voljo le glasovanjem na seznamu lastnih zasebnih glasovanj.

Gumb Glasuj je prikazan samo takrat, ko ima atribut odprto vrednost true. Prikaz gumba Odpri Glasovanje zahteva preverjanje več vrednosti. Preveriti moramo ali je glasovanje zaprto in še ne zaključeno. Najpomembnejše pa je, da gumb prikažemo samo avtorjem glasovanja. Deli Dovoljenja prikazujemo samo takrat, ko še ne delimo nobenega drugega glasovanja. Prav tako moramo preveriti, da glasovanje še ni zaključeno ter, kot pri prejšnjem gumbu, to smemo prikazati samo avtorjem glasovanja. Gumb Zaključi Deljenje prikažemo samo ob pregledovanju deljenega glasovanja.

3.2.5 Deljenje dovoljenj

V prejšnji točki smo izvedeli, kdaj ima skrbnik na voljo izbiro za deljenje dovoljenj. Tukaj bomo pa podrobneje opisali potek deljenja dovoljenj. Ob pritisku na gumb shranimo v spremenljivko izbrano glasovanje ter omogočimo NFC zahteve. Za izvedbo deljenja dovoljenja potrebujemo skrbnika in hkrati avtorja glasovanja (v nadaljevanju avtor), ki deli dovoljenja in običajnega uporabnika lahko tudi skrbnika (v nadaljevanju uporabnik), ki želi prejeti dovoljenje. Ob stiku mobilnih naprav, uporabnikova naprava pošlje avtorju sporočilo, ki vsebuje njegove uporabniške podatke. Na strani avtorja se nato pripravi zahteva z naslednjimi atributi: uporabnikovo uporabniško ime, uporabnikovo geslo, avtorjevo uporabniško ime, avtorjevo geslo in id glasovanja. Pripravljeno zahtevo pošljemo na stran strežnika, kjer se ustrezno preverijo podatki in poskusi dodati dovoljenje. Dovoljenje se uspešno doda izključno v primeru, ko sta oba uporabnika uspešno identificirana, glasovanje s podanim avtorjem je v stanju deljenja dovoljenj in uporabnik še ni prejel dovoljenja za izbrano glasovanje. Izvedba delitve dovoljenj je podana na sliki 3.19, ki prikazuje izvedbo testiranja mobilne aplikacije.



Slika 3.19: Diagram poteka izdaje dovoljenja.

Spodnji prikaz odgovorov predstavlja eno uspešno izvedbo zahtevo in eno ne-uspešno. Vzrok neuspeha je moč razbrati iz vrednosti ključa "error_msg".

```
1  [{
2    "tag": "add_dovoljenje"
3  }, {
4    "success": 1
5  }, {
6    "error": 0
7  }, {
8    "success_msg": "Dovoljenje uspesno dodano!"
9  }]
10
11  -----
12  [{
13    "tag": "add_dovoljenje"
14  }, {
15    "success": 0
16  }, {
17    "error": 1
18  }, {
19    "error_msg": "Uporabnik ni veljaven!"
20  }]
```

3.2.6 Strežniški API

Strežniški API je bil že velikokrat omenjen, tukaj bo pa tudi podrobneje predstavljen. Njegov namen je komunicirati z zunanji storitvi. S splošnega vidika bi se lahko reklo, da se uporablja za pošiljanje podatkov na zahtevo zunanjih storitev. Strežniški API je v našem primeru na voljo izključno uporabnikom glasovalnega sistema NFC. Edina zunanja storitev, ki izkorišča funkcionalnosti API je pa mobilna aplikacija.

Ob prejemu zahtevku se najprej izvede preverjanje identitete uporabnika oziroma pošiljatelja zahteve. S podanim uporabniškim imenom in geslom poskušamo uporabnika prijaviti. Zavrnitev prijave onemogoči nadaljnje izvajanje zahteve in

storitvi v odgovoru posredujemo vzrok neuspešne prijave. Uspešna prijava oziroma identifikacija uporabnika dovoli nadaljnjo izvedbo zahtevka in ob koncu odgovori z ustreznim sporočilom.

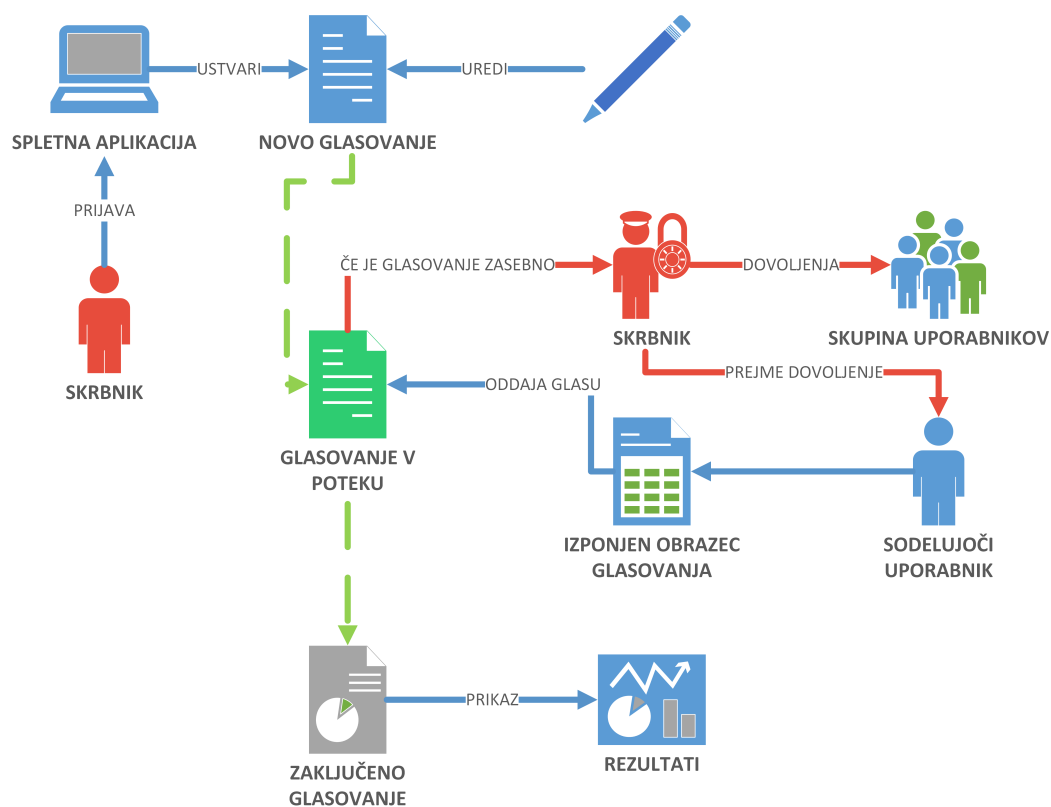
API implementira izvajanje naslednjih zahtev: prijava in registracija uporabnika, pridobitev glasovanj, pridobitev posameznega glasovanja, kreiranje novega dovoljenja, manipuliranje s stanjem glasovanja. Vsem opravilom sta skupni začetna validacija uporabnika in format končnega odgovora.

Poglavje 4

Testiranje

Pred izvedbo testa je bilo treba spletno aplikacijo namestiti na spletni strežnik. Nahaja se na strežniku podjetja Lucis d.o.o. z uporabo virtualnega okolja. V prvem koraku namestitve je bilo ustvarjeno virtualno okolje z uporabo programa Virtualenv. Znotraj virtualnega okolja smo namestiti ustrezne knjižnice in programe, ki jih zahteva predlagana rešitev. V naslednjem koraku je bilo treba ločeno od strežnika zagnati izvajalca opravil, ki na vsaki dve minuti izvede ustrezna opravila. Zagon strežnika pa je bil izveden v zadnjem koraku.

Testiranje je potekalo na dveh mobilnih napravah (Sony Xperia M2 - Android v4.4.2 in LG Optimus L9 - Android v4.1.2) in prenosnem računalniku (HP EliteBook 8560p). V ta namen je bilo ustvarjenih šest uporabniških računov z izmišljenimi podatki. Polovica ustvarjenih uporabnikov je bilo hkrati skrbnikov. Testni primer vključuje prijavo uporabnika skrbnika, ki ustvari novo zasebno glasovanje s tremi vprašanji. Ob prehodu glasovanja v stanje izvajanja, skrbnik začne izdajati dovoljenja. Ob prejetih dovoljenjih uporabniki oddajo svoj glas. V zadnjem koraku skrbnik zapre glasovanje in prikaže rezultate vsem navzočim uporabnikom. Za lažjo predstavbo posameznih korakov je podana slika 4.1 in podrobnejši opis posameznega koraka testiranja.



Slika 4.1: Diagram poteka testiranja.

V prvem koraku se skrbnik prijavi v spletno aplikacijo z ustreznim E-Mail naslovom in geslom. V naslednjem koraku izbere Ustvari novo glasovanje v menijski vrstici in vnese naslednje vrednosti v podan obrazec:

- **Naslov:** Testiranje aplikacije
- **Opis:** Testni primer delovanja spletne in mobilne aplikacije
- **Pričetek:** 25.8.2014 16:00
- **Konec:** 30.8.2014 12:00
- **Udeležba skrbnika glasovanja pri glasovanju:** Da
- **Anonimno:** Da
- **Javno:** Ne

Sledi ustvarjanje vprašanj glasovanja, kjer skrbnik vnese vrednosti:

1. Potrditev zapisnika z dne 14.7.2014

- (a) *Da*
- (b) *Ne*

2. Glasujte za vodjo projekta

- (a) *Janez Novak*
- (b) *Peter Klepec*
- (c) *Ana Muha*

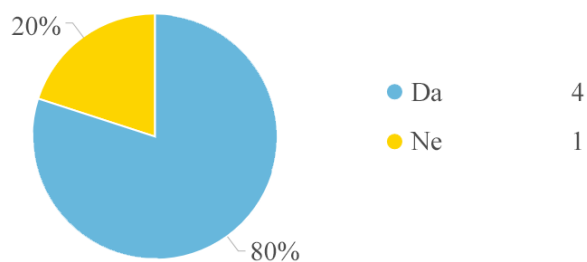
3. Ocenite predstavitev osebe: Miha Klopčič

- (a) *Slabo*
- (b) *Dobro*
- (c) *Odlično*

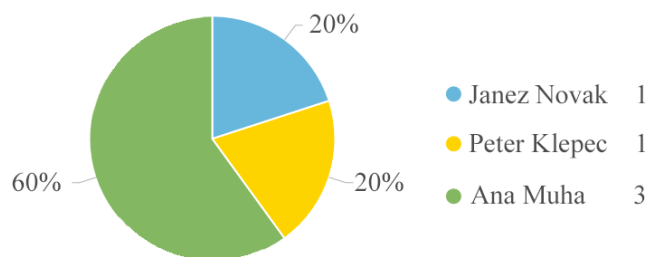
Po končanem ustvarjanju skrbnik odpre glasovanje s pritiskom na gumb *Od-pri glasovanje*, ki je pod možnostmi glasovanja. Ker je glasovanje zasebne narave mora skrbnik izdati dovoljenje uporabnikom. Trije uporabniki so že vključeni v skrbnikovi skupini uporabnikov z imenom *Zaposleni*. Navedeni skupini izda skupinsko dovoljenje, preostalima uporabnikoma pa izda dovoljenje s pomočjo mobilne aplikacije. Skrbnik se na mobilni napravi *Sony Xperia M2* prijavi v predlagano mobilno aplikacijo ter prične deliti dovoljenja za izbrano glasovanje. Hkrati se na drugi mobilni napravi, *LG Optimus L9*, prijavi prvi uporabnik, ki želi prejeti dovoljenje. Za pridobitev dovoljenja uporabnik prisloni NFC napravo k skrbnikovi NFC napravi in pošlje zahtevek s pritiskom na zaslon. Skrbnikova naprava izvede zahtevo za dovoljenje in sporoči uspešen rezultat izvedbe. Nato ponovimo isti postopek, le da se tokrat na mobilni napravi *LG Optimus L9* prijavi v aplikacijo drugi uporabnik.

Ko vsi uporabniki prejmejo dovoljenje za pristop h glasovanju, vsak posamezni uporabnik odpre navedeno glasovanje, ki je na seznamu zasebnih glasovanj in izpolni ter odda obrazec. V zadnjem koraku, ko so že vsi uporabniki oddali svoj glas, skrbnik zaključi glasovanje in vsi uporabniki si na projektorju ogledajo rezultate glasovanja (slika 4.2) preko skrbnikovega uporabniškega računa.

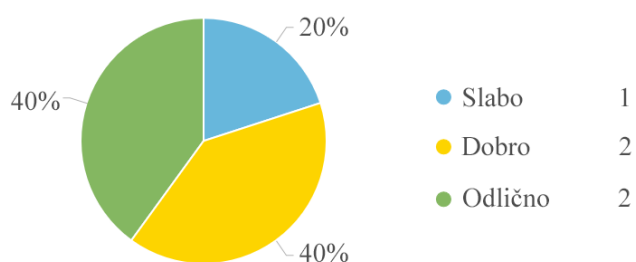
1. Potrditev zapisnika z dne 14.7.2014



2. Glasujte za vodjo projekta



3. Ocenite predstavitev osebe: Miha Klopčič



Slika 4.2: Rezultati glasovanja v obliki tortnih diagramov.

Poglavje 5

Sklepne ugotovitve

Med izdelavo diplomskega dela sem se prvič spoznal s tehnologijo NFC. Ob spoznanju njenega uporabniškega potenciala sem bil začuden, nad tako skromno uporabo NFC aplikacij med uporabniki NFC mobilnih naprav. Sprva sem bil zelo skeptičen glede implementacije dela aplikacije, ki uporablja to tehnologijo. Pa vendar se je na koncu izkazalo, da je le to sila preprosto in intuitivno. Pri dejanski uporabi Android Beam-a se pokažejo določene pomanjkljivosti, na katere razvijalec aplikacij ne more vplivati. Za uspešno oddajo NFC zahtevka morata biti napravi ena ob drugi in če je le možno tudi pri miru. Takšen manever zahteva od uporabnika koncentracijo in mirno roko, kjer v nasprotnem primeru pride do neuspešnega prenosa podatkov. To se največkrat zgodi ob prenosu večje količine podatkov, kjer proces traja dlje časa zaradi nizke histrosti prenosa.

Kot vsaka aplikacija ali program tudi Glasovalni sistem NFC vsebuje določene pomanjkljivosti ali bolje rečeno prostor za izboljšave in implementacijo dodatnih funkcionalnosti. V nadaljnjem razvoju aplikacije bi bilo treba implementirati uvoz uporabnikov iz že obstoječe podatkovne baze. S tem bi pridobili preprost prenos aplikacije v okolje poljubne organizacije. Smiselno bi bilo tudi nadgraditi varnost tako spletne kot tudi mobilne aplikacije. Predvsem mobilna aplikacija je ranljiva zaradi uporabe NFC tehnologije, saj je to področje dokaj novo in varnost ni njena boljša stran. Uporaba predlagane rešitve na mobilni napravi ne omogoča izvajanja vseh funkcionalnosti brez uporabe brskalnika. To je za današnje standarde aplikacij moteča pomanjkljivost, ki pa za raziskovalne namene ni ključnega pomena. Glasovalni sistem tudi vsebuje možnosti nadgradnje, bolj pomembna bi bila npr.:

dodatni tipi vprašanj, ponastavitev zaključenih glasovanj, deljenje glasovanj preko družabnih omrežij, itd.

Literatura

- [1] Szu-Hui Wu, “Promoting collaborative mobile payment by using NFC-Micro SD technology”, *IEEE Computing*, 10th International Conference on Services Computing, 2013.
- [2] Balazs Benyo, Balint Sodor, Tibor Doktor and Gergely Fordos “Student attendance monitoring at the university using NFC”, Wireless Telecommunications Symposium (WTS), 2012.
- [3] Kerem OK, Vedat COSKUN, Mehmet N. AYDIN, and Busra OZDENIZCI “Current Benefits and Future Directions of NFC Services”, International Conference on Education and Management Technology, 2010.
- [4] B. Benyo, A. Vilmos, K. Kovacs and L. Kutor, “The Design of NFC Based Applications”, Intelligent Engineering Systems, 2007
- [5] Kathy Sierra, “Head First Java”, O’Reilly Media; 2nd edition (February 9, 2005)
- [6] Marty Alchin, “Pro Django”, Expert’s Voice in Web Development, Apress; 2 edition (July 9, 2013)
- [7] Vedat Coskun, “Professional NFC Application Development for Android”, Wrox; 1 edition (April 22, 2013)
- [8] Android NFC dokumentacija, dostopno na:
<http://developer.android.com/guide/topics/connectivity/nfc/index.html>
- [9] Django Allauth vtičnik, dostopno na:
<https://github.com/pennersr/django-allauth>

- [10] Izvedba linux cron opravil, dostopno na:
<http://www.cyberciti.biz/faq/how-do-i-add-jobs-to-cron-under-linux-or-unix-oses/>

- [11] SQLite dokumentacija, dostopno na:
<http://www.sqlite.org/docs.html>

- [12] PyCharm dokumentacija, dostopno na:
<http://www.jetbrains.com/pycharm/documentation/>

- [13] PDF/NFC Forum Type Tags, dostopna na:
<http://67.222.41.204/wp-content/uploads/2013/12/NFC-Forum-Type-Tags.pdf>