

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Luka Pifko

# Mobilna aplikacija za izvajanje anket

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Rok Rupnik

Ljubljana 2012

Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. 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  $\LaTeX$ .*

**IZJAVA O AVTORSTVU DIPLOMSKEGA DELA**



Št. naloge: 00334/2012

Datum: 04.09.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **LUKA PIFKO**

Naslov: **MOBILNA APLIKACIJA ZA IZVAJANJE ANKET**  
**MOBILE APPLICATION FOR SURVEYS**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

Zasnujte sistem za podporo izvajanju anket na terenu. Sistem naj obsega spletno aplikacijo za vnos vprašalnikov za ankete ter različne grafične in tabelarične prikaze rezultatov anket. Drugi element sistema pa naj bo mobilna aplikacija, ki podpira izvajanje anket na terenu.

Mentor:

doc. dr. Rok Rupnik



Dekan:

prof. dr. Nikolaj Zimic

Spodaj podpisani Luka Pifko, z vpisno številko **63070382**, sem avtor diplomskega dela z naslovom:

*Mobilna aplikacija za izvajanje anket*

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Roka Rupnika,
- 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 v zbirki "Dela FRI".

V Ljubljani, dne 6. december 2012

Podpis avtorja:

## **Zahvala**

*Za pomoč in nasvete pri izdelavi diplomske naloge se zahvaljujem mentorju doc. dr. Roku Rupniku. Zahvaljujem se svojim staršem, ki so mi študij omogočili, me spodbujali in vsa leta finančno podpirali. Posebna zahvala gre tudi Anji, ki mi je ves čas stala ob strani.*

# Kazalo

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Predstavitev uporabljenih tehnologij</b>	<b>5</b>
2.1	Razvojne tehnologije v spletni aplikaciji . . . . .	5
2.1.1	Paket za vzpostavitev spletnega strežnika XAMPP . . .	5
2.1.2	Označevalni jezik HTML in novosti v HTML5 . . . . .	6
2.1.3	CSS . . . . .	7
2.1.4	Sistem za upravljanje s podatkovnimi bazami MySQL .	8
2.1.5	Skriptni programski jezik JavaScript . . . . .	9
2.1.6	Knjižnica JQuery . . . . .	9
2.1.7	Programski jezik PHP . . . . .	10
2.1.8	AJAX . . . . .	11
2.1.9	DOM . . . . .	11
2.2	Razvojne tehnologije v mobilni aplikaciji . . . . .	12
2.2.1	Knjižnica JQuery mobile . . . . .	12
2.2.2	Medplaformsko ogrodje PhoneGap . . . . .	14
2.2.3	Spletni programski vmesnik Web SQL Database . . . . .	15
2.3	Razvojna orodja . . . . .	16
2.3.1	Integrirano razvojno okolje Eclipse . . . . .	16
2.3.2	Ripple emulator . . . . .	17

2.3.3	Lita . . . . .	18
2.3.4	Orodje za modeliranje podatkovnih baz SQL Power Architect . . . . .	18
<b>3</b>	<b>Sestavljalec vprašalnikov</b>	<b>19</b>
3.1	Cilji in namen aplikacije . . . . .	19
3.2	Analiza . . . . .	20
3.2.1	Zahteve . . . . .	20
3.2.2	Primer uporabe . . . . .	21
3.2.3	Konceptualni model podatkovne baze . . . . .	21
3.3	Sestavljanje novega vprašalnika . . . . .	23
3.3.1	Vsebniki in medsebojna interakcija . . . . .	23
3.3.2	Projekt . . . . .	26
3.3.3	Različni tipi vprašanj . . . . .	26
3.3.4	Predloge različnih tipov vprašanj . . . . .	31
3.3.5	Lastnosti vprašanj in dodatne možnosti . . . . .	32
3.3.6	Dinamična gradnja vprašalnika in postopek gradnje vsebnika . . . . .	33
3.3.7	Pomnjenje stanja vprašalnika . . . . .	36
3.4	Potek in sprememba toka vprašalnika . . . . .	37
3.5	Pregled vprašalnikov . . . . .	39
3.5.1	Vpogled in možne operacije nad vprašalniki . . . . .	39
3.5.2	DataTables . . . . .	40
3.6	Grafična predstavitev rešenih anket . . . . .	41
3.6.1	Prikaz podatkov z uporabo Google Charts . . . . .	41
<b>4</b>	<b>Mobilna aplikacija za reševanje anket</b>	<b>43</b>
4.1	Osnovne zahteve in cilji mobilne aplikacije . . . . .	43
4.2	Realizacija osnovnih zahtev . . . . .	44
4.3	Povezani način ( <i>Online mode</i> ) . . . . .	45
4.3.1	Sinhronizacija vprašalnikov iz oddaljene baze podatkov . . . . .	45
4.3.2	Pošiljanje rešenih anket v oddaljeno bazo podatkov . . . . .	48

## KAZALO

4.4	Nepovezani način ( <i>Offline mode</i> ) . . . . .	49
4.4.1	Izbira vprašalnika . . . . .	49
4.4.2	Izpolnjevanje vprašalnika . . . . .	51
<b>5</b>	<b>Sklepne ugotovitve</b>	<b>55</b>
	<b>Kazalo slik</b>	<b>59</b>
	<b>Literatura</b>	<b>63</b>



# Seznam uporabljenih kratic

**AJAX** - Asinhroni JavaScript in XML, skupina medsebojno povezanih spletnih razvojnih tehnik.

**API** - Application Programming interface; programski vmesnik.

**CSS** - Cascading Style Sheets; stilske podloge, ki so namenjene za definiranje stila spletnih dokumentov.

**DOM** - Document Object Model; objekt z vsebino spletnega dokumenta.

**HTML** - Hyper Text Markup Language; označevalni jezik za oblikovanje spletnih strani.

**IDE** - Integrated Development Kit; integrirano razvojno okolje.

**JS** - JavaScript; objektni skriptni programski jezik.

**JSON** - JavaScript Object Notation; zapis objekta v skriptnem jeziku JavaScript.

**JQuery** - programska knjižnica objektno skriptnega programskega jezika JavaScript.

**PHP** - PHP HyperText Processor; skriptni programski jezik namenjen izdelavi dinamičnih spletnih strani.

**SQL** - Structured Query Language; strukturni povpraševalni jezik za delo s podatkovnimi zbirkami.

**XAMPP** - Paketni spletni strežnik XAMPP, sestavljen iz X, Apache, MySQL, PHP in Perl.

# Povzetek

Glavni cilj diplomskega dela je izdelava sistema za podporo terenskem anketiranju. V uvodu je opredeljen problem terenskega anketiranja in razlogi za izdelavo takega sistema. Opisane so vse uporabljene tehnologije in orodja, ki so bila uporabljena za doseg cilja. Izdelana je spletna aplikacija namenjena za sestavljalce oziroma načrtovalce vprašalnikov. Sestavljanje vprašalnikov poteka dinamično in zgolj na odjemalčevi strani do točke, ko uporabnik želi vprašalnik zapisati v podatkovno bazo. Poleg sestavljanja vprašalnikov lahko uporabnik v spletni aplikaciji tudi pregleduje vprašalnike in izvaja različne operacije nad njimi. Možen je tudi grafični pregled zbranih odgovorov anketarjev na mobilni napravi. Poleg načrtovanja in pregleda vprašalnikov v spletni aplikaciji je bila razvita še mobilna aplikacija, z namenom izpolnjevanja vprašalnikov. Aplikacija je namenjena terenskim anketarjem. Za izdelavo smo uporabili medplatformsko ogrodje PhoneGap, ki deluje na velikem številu popularnih mobilnih operacijskih sistemih. Mobilna in spletna aplikacija poenostavita proces sestavljanja vprašalnikov, zbiranja odgovorov in celovit pregled zbranih odgovorov respondentov. V celoti odstranita potrebo po uporabi papirnate oblike in podjetjem, ki se ukvarjajo z raziskovanjem trga, lahko predstavljata znatno znižanje stroškov. Jedro diplomskega dela je načrtovanje, razvoj in predstavitev mobilne in spletne aplikacije. V zaključku so predstavljene sklepne ugotovitve in razmislek o možnih dodatnih funkcionalnostih aplikacije.

Seznam ključnih besed: Spletna aplikacija, mobilna aplikacija, terenski anketar, PhoneGap, jQuery, jQuery Mobile, XAMPP, AJAX, HTML.

# Abstract

The main objective of the diploma thesis is to produce a system that supports field interviewers. The introduction defines the problem and reasons for making such a system. All the technologies which we have used to achieve the objective are described within. We have made a web application designed for questionnaire designers. The designing of the questionnaires is dynamic and client-side up until the user decides to send the data to the database. In addition to designing questionnaires, the user has an overview of all the questionnaires and can perform various operations on them. There is also a graphical overview of the interview responses collected by the interviewer on the mobile device. We have also developed a mobile application for field interviewers. It can download and store the questionnaires, of which the responses can then be logged and reported back to the server. For the production of the mobile application we have used PhoneGap framework that is intended to work on a large number of popular mobile operating systems. Mobile and web applications simplify the process of creating questionnaires, collecting responses and viewing comprehensive reports of the collected answers of respondents. The system fully removes the need to use paper forms which can lead to significant cost savings for enterprises engaged in researching the market. The core of the thesis is the design, development and presentation of the mobile and web applications. In the conclusion, we provide findings and reflection on possible additional application functionality.

List of keywords: Web application, mobile application, field interviewer, PhoneGap, jQuery, jQuery Mobile, XAMPP, AJAX, HTML.

# Poglavje 1

## Uvod

Anketiranje je zbiranje podatkov, s pomočjo katerih lahko podatke o stališčih in mnenjih vprašancev posplošujemo na večjo populacijo. Je ena izmed najbolj pogostih uporabljenih metod v družbenih vedah. V znanstvenem raziskovanju je anketiranje postavljanje točno določenih vprašanj, točno določeni skupini in številu ljudi. Ko raziskujemo trg s pomočjo ankete in želimo posploševati na večjo ali celotno ciljno populacijo je zelo pomembna izbira vzorca. Večji kot je vzorec, večji je strošek za opravljanje same dejavnosti anketiranja. V dobi digitalizacije, kjer se vse seli iz fizične oblike v digitalno, zaradi nižanja stroškov ali zagotavljanja lažjega izvajanja, se je vse več panog uspelo modernizirati in slediti bliskovitemu razvoju tehnologije. V diplomskem delu smo se lotili digitalizacije procesa terenskega anketiranja, kjer želimo papirnate vprašalnike nadomestiti z naprednimi spletnimi in mobilnimi tehnologijami.

V panogi raziskovanja in analizi trga smo na slovenskem trgu opazili zastajanje za tehnološkim napredkom. Vsaka organizacija, ki ne želi zaostajati, mora slediti razvoju. Kratkoročno celotna mobilizacija in spreminjanje načina dela ne prinese kratkoročnih pozitivnih posledic v finančnem smislu, dolgoročno se v večini organizacij sledenje tehnološkemu premikom obrestuje.

Uveljavljen in zastarel proces zbiranja mnenj se prične s sestavljanjem vprašalnika, na podlagi opredeljenih raziskovalnih problemov, ciljev in hipo-

tez. Ko je vprašalnik sestavljen, je potrebno natisniti veliko število izvodov tega vprašalnika, odvisno od zastavljenega vzorca. Izvode je potem potrebno distribuirati vsem terenskim anketarjem, katerim je potrebno fizično predati te izvode. Ko anketarji prevzamejo vprašalnike je v primeru napake sestavljalca ankete, potrebno slednji omenjeni proces ponoviti in ponovno natisniti vprašalnike in jih fizično predati anketarjem. Naslednji korak je dejansko anketiranje respondentov in nezmožnost nadzora nad anketarji. Skoraj nemogoče je opravljati nadzor in preverjanje kakovosti ali se izogniti ponarejanju vprašalnikov s strani anketarja. Obstaja tveganje, da anketarji sami rešujejo ankete, ker se zavedajo nemoči organizacij, ki izvajajo ankete. Ravno zaradi zavedanja anketarjev in njihove želje po večjem zaslužku z manj vloženega truda, se te anomalije dogajajo in lahko v celoti ustvarijo napačne rezultate raziskave. Ko so vsi vprašalniki izpolnjeni in jih anketarji dostavijo vodji projekta je potrebno delo dodatnih ljudi, da vse te odgovore zberejo v neko podatkovno shrambo, kjer bodo vsi rezultati vsakega vprašanja zbrani na enem mestu in bodo na voljo raziskovalcem za obdelavo. Torej mora nekdo vse te podatke vpisovati, večinoma se to vpisuje v namensko aplikacijo, kjer nekdo odgovore na vprašanja zapisuje in se ti podatki zapisujejo v bazo podatkov.

Zgornji odstavek dobro prikaže nekaj očitnih pomankljivosti, katere so prisotne pri terenskem anketiranju in katere bi lahko vsaj delno odpravili z uporabo pametnih mobilnih naprav ali tabletnih računalnikov. Namen spletne aplikacije je poenostavitev dela za sestavljalca vprašalnikov, da bi na čim bolj preprost in enostaven način imel ves nadzor nad kreiranjem, urejanjem in pregledovanjem vprašalnikov. Mobilna aplikacija služi anketarjem, da sestavljene ankete prevzamejo v digitalni obliki na svoji mobilni napravi in jih potem izpolnjujejo z respondenti.

Poglavitne prednosti digitalne oblike terenskega anketiranja:

- Ni stroškov, ki jih prinese papirnata oblika(kartuše, papir, tiskalnik).
- Napake sestavljalca anket so lažje popravljive, ni potrebe po ponovitvi

faze tiskanja vprašalnikov in predaja vprašalnikov anketarjem.

- Lažja predaja vprašalnikov, anketarji preko mobilne naprave, ki ima povezavo do interneta, vprašalnike naložijo iz oddaljenega strežnika.
- Večji nadzor nad rešenimi vprašalniki. Anketarji ne morejo ponarediti datuma in ure opravljene ankete. Možen je tudi nadzor lokacije in zapis koordinatov, kje je bila anketa opravljena.
- Vsi zbrani odgovori se lahko neposredno zapišejo v oddaljeno podatkovno bazo, kjer so praktično že pripravljene za nadaljno obdelavo. V celoti se odpravi delo dodatnih ljudi, ki morajo te ankete iz papirne oblike spraviti v obliko, ki je primerna za obdelavo in pripravo zaključkov raziskave.
- Anketarju je olajšano delo, ker lažje popravlja odgovore na vprašanja za katere se respondent premisli.
- Možnost podajanja mnenj na podlagi multimedijskih vsebin, ki jih na papirju ni možno predstaviti, kot so slikovni in zvokovni posnetki.
- Sodeč po trenutnem trendu, kjer ima že veliko ljudi lastne pametne mobilnike, bi anketiranje opravljali kar preko svoje naprave.

V okviru diplomske naloge smo razvili celoten sistem, ki sestoji iz spletne aplikacije namenjene za sestavljalce vprašalnikov in mobilne aplikacije. Slednja lahko teče na večih mobilnih operacijskih sistemih in na njej zbiramo odgovore respondentov. Na začetku so opisane vse uporabljene tehnologije in njihova kratka predstavitev. V naslednjih poglavjih je opisana izdelava in razvoj spletne in mobilne aplikacije.



## Poglavje 2

# Predstavitev uporabljenih tehnologij

### 2.1 Razvojne tehnologije v spletni aplikaciji

V tem podpoglavju so opisane vse uporabljene tehnologije, ki smo jih uporabili za izdelavo spletnega dela, za podporo terenskemu anketarju.

#### 2.1.1 Paket za vzpostavitev spletnega strežnika XAMPP

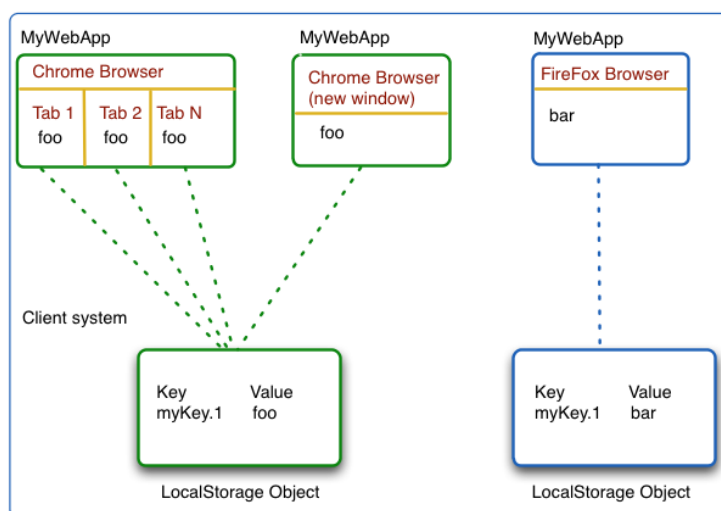
Paketni spletni strežnik XAMPP (angl. *X, Apache, MySQL, PHP and Perl*) je zastonjski in odprtokoden<sup>1</sup> sistem za hitro vzpostavitev spletnega strežnika na operacijskem sistemu Windows. Najpomembnejši gradniki paketa so spletni strežnik Apache, podatkovna zbirka MySQL in prevajalniki za skriptne programske jezike PHP in Perl.

Primarno je bil namen spletnega strežnika XAMPP kot razvojno orodje, ki bi programerjem omogočal lažjo vzpostavitev testnega okolja, kjer ni potrebne povezave do interneta. Ravno ta razlog je bil tudi glavna prednost, zaradi katere smo se odločili uporabljati to orodje. Snovalci paketnega spletnega strežnika XAMPP odsvetujejo uporabo s privzetimi nastavitvami v

---

<sup>1</sup>Odprta koda(*Open source*) omogoča vsem dostopnost do kode produkta in se smatra kot ena izmed možnih pristopov načrtovanja





Slika 2.1: Porazdelitev lokalne shrambe med različne brskalnike na isti napravi[3].

komercialne namene zaradi varnostnih pomankljivosti[4].

### 2.1.2 Označevalni jezik HTML in novosti v HTML5

HTML (*HyperText Markup Language*) je označevalni jezik za izdelovanje spletnih strani in njenih vsebin, ki jih je mogoče prikazati v katerem koli od spletnih brskalnikov[1]. Predstavlja osnovo spletnega dokumenta in z njim določamo strukturo in semantični pomen delov dokumenta. Dokument je zapisan v obliki HTML elementov, ki so sestavljeni iz značk (*angl. tags*) in znotraj teh značk je vsebina. Dokument HTML je znakovna datoteka, katero lahko izdelamo in urejamo v vsakem urejevalniku besedil.

Najnovejša različica je HTML5, ki je prinesla nekaj sprememb na podlagi strukture dokumenta in to se pozna pri dodajanju novih značk (*section, header, footer, article*, itd.), novih tipov obrazcev (*date, month, time, number, email, url*, itd.). Novost so tudi novi elementi za podporo multimedijskih vsebin: `<canvas>`, `<video>` in `<audio>`.

Prav tako je ena od novosti lokalna shramba (*angl. local storage*), ka-

tera je naslednica mehanizma piškotkov (angl. *cookies*). Lokalna shramba omogoča hitrejšo in varnejšo shrambo podatkov, kjer so podatki shranjeni na odjemalčevi strani in jih ni potrebno vsakič znova zahtevati od strežnika. V primerjavi s piškotki omogoča precej večjo količino shranjenih podatkov [2]. Podatki so shranjeni v obliki parov ključ-vrednost. Take vrste lokalnega shranjevanja podatkov smo uporabili tudi pri razvoju spletne in mobilne aplikacije, več o sami uporabi v aplikaciji je napisano na strani 36. Na sliki 2.1 je predstavljena porazdelitev lokalne shrambe med različnimi spletnimi brskalniki na isti napravi, kjer je razvidno da si spletni brskalniki ne delijo lokalne shrambe.

### 2.1.3 CSS

CSS (angl. *Cascading Style Sheets*) so stilske podloge, ki so namenjene za definiranje stila dokumentov HTML. Z definicijo različnih pravil določimo prikaz in pozicioniranje teh elementov na spletni strani. V okviru pravil lahko določamo različne attribute elementov, kot so barve, velikosti, odmike, poravnave in vrsto drugih.

Bistvo uporabe stilskih podlog je ločitev strukture strani, katero določimo z označevalnim jezikom HTML od načina kako bomo stran predstavili. Za predstavitev poskrbimo z definiranjem pravil in obnašanja elementov s stilskimi podlogami.[5]

Najnovejša različica stilskih podlog je CSS 3, ki je razvijalcem spletnih strani olajšala izdelavo nekaterih lepotnih dodatkov. Večinoma so to dodatki, za katere so morali razvijalci spletnih strani uporabljati kar nekaj zvijač in precej vrstic kode, da so prišli do zelenega efekta. S prihodom CSS 3 je to olajšano z enostavno spremembo atributa v stilski podlogi. Nekaj dodanih atributov je opisanih v tabeli 2.1.

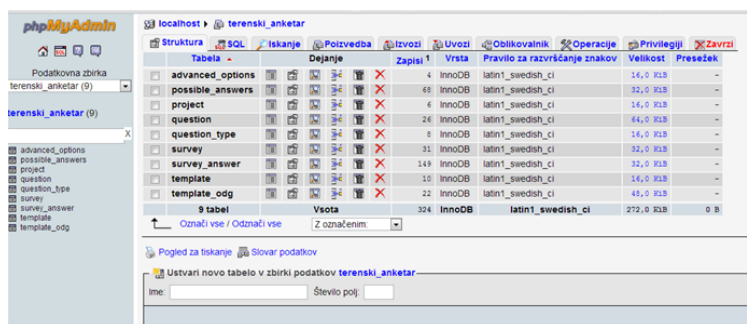
<b>Atribut</b>	<b>Lastnost</b>	<b>Opis</b>
border-radius	Zaokroženi robovi	Na elementih ustvari efekt zaokroženih robov.
border-image	Robna slika	Uporaba lastne slike za okraševanje okvirja elementov.
box-shadow	Senčenje elementov	Preprosto senčenje elementov.
text-shadow	Senčenje besedila	Preprosto senčenje besedila.
webkit-column-count	Stolpčna postavitev elementov	Postavitev z večimi stolpci, kjer lahko definiramo tudi število stolpcev.
opacity	Prozornost	Določanje prozornosti elementov

Tabela 2.1: Nekaj najbolj popularnih novitet v CSS 3

### 2.1.4 Sistem za upravljanje s podatkovnimi bazami MySQL

MySQL je sistem za upravljanje s podatkovnimi bazami. MySQL je odprtokodna implementacija relacijske podatkovne baze, ki za delo s podatki uporablja jezik SQL.[7] Napisan je v programskih jezikih C in C++ in deluje na različnih operacijskih sistemih. MySQL je najbolj razširjen odprtokodni sistem za upravljanje s podatkovnimi bazami in zagotavlja večuporabniški dostop do večih podatkovnih baz. Poleg odprtokodne različice, ki je bila izdana pod licenco GNU (angl. *General public license*) obstajajo tudi različne komercialne različice, ki nudijo nekaj dodatnih funkcionalnosti za razliko od odprtokodne. Nekatere bolj znane spletne aplikacije, ki uporabljajo MySQL so Joomla, WordPress in phpBB.

Za lažji dostop do podatkovne baze z grafičnim vmesnikom smo pri izdelavi uporabili orodje phpMyAdmin, ki je del paketa XAMPP. Orodje je napisano v programskem jeziku PHP in je odprtokodno. Namenjeno je lažji ad-



Slika 2.2: PhpMyAdmin orodje za administracijo sistema za upravljanje s podatkovnimi bazami MySQL.

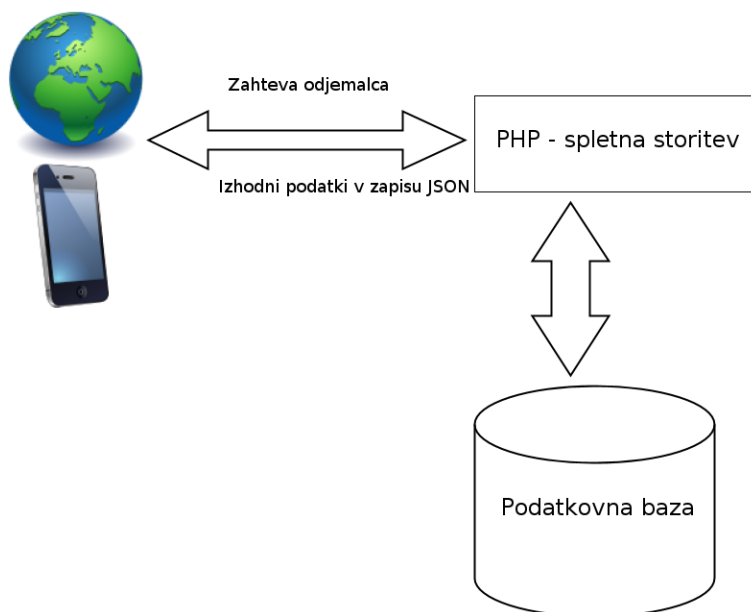
ministraciji sistema za upravljanje s podatkovnimi bazami MySQL s pomočjo spletnega brskalnika. Poleg administracije je z orodjem možno tudi kreiranje, spreminjanje in brisanje podatkovnih baz, tabel in vnosov v tabeli. Omogoča nam hitro in enostavno izvajanje poizvedb, pregledovanje vnosov, urejanje lastnosti polj v tabeli. Priročna lastnost je tudi uvoz in izvoz datotek sql, ki jih kreirajo orodja za modeliranje podatkovne baze.

### 2.1.5 Skriptni programski jezik JavaScript

JavaScript je objektni skriptni programski jezik, katerega namen je izdelava interaktivnih spletnih strani. V sodelovanju s kodo HTML lahko stran naredi dinamično in interaktivno. JavaScript podpora je vgrajena neposredno v vseh večjih spletnih brskalnikih in za njeno delovanje ne potrebujemo namestiti dodatnih prevajalnikov.

### 2.1.6 Knjižnica JQuery

jQuery je najbolj popularna programska knjižnica objektno skriptnega programskega jezika JavaScript. Uporablja jo 55% od 10.000 najbolj obiskanih spletnih strani[8]. Razvijalcu olajša delo, ker poskrbi za vse razlike med spletnimi brskalniki, za katere bi brez knjižnice moral poskrbeti sam.



Slika 2.3: Grafičen prikaz spletne storitve in medsebojne interakcije med podatkovno bazo, odjemalcem in spletno storitvijo.

Podpira in olajša usmerjanje HTML dokumentov, izbiro in manipulacijo DOM objektov in ustvarjanje animacij. V spletni aplikaciji smo ga uporabili za izdelavo AJAX klicev in pogosti manipulaciji z DOM objekti.

### 2.1.7 Programski jezik PHP

Programski jezik PHP se uporablja za razvoj dinamičnih spletnih vsebin. Vgrajen je v izvorno kodo HTML, ki je interpretirana s strani spletnega strežnika na katerem teče procesni modul PHP, ki kot izhod iz izvorne kode generira spletno stran.

V aplikaciji smo ga uporabili kot spletno storitev, ki teče na strežniku in prejema zahtevke. Spletne strani preko klica AJAX pošljejo podatke v obliki JSON, spletna storitev zahtevo sprejme in jo procesira. V večini primerov je uporabljen za povezavo do podatkovne baze, kjer naredimo določeno poizvedbo in podatke vrnemo spletni strani, ki je klicala spletno storitev. Proces

je viden na sliki 2.3.

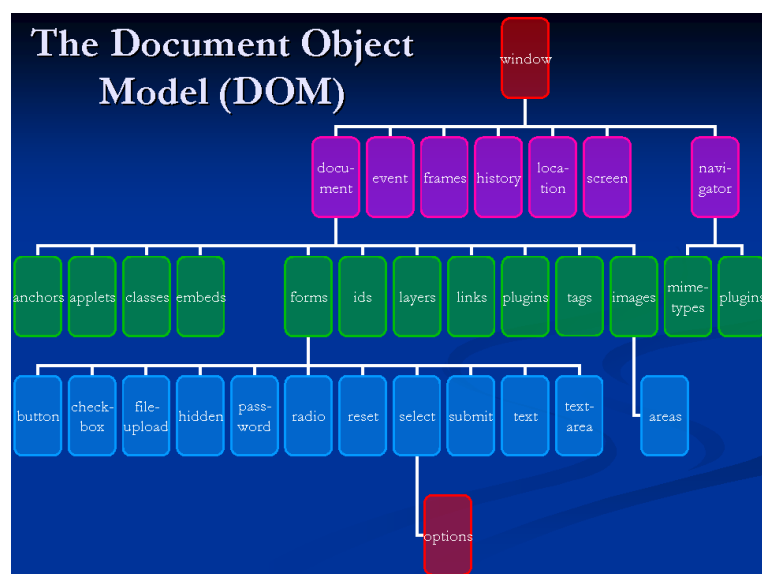
### 2.1.8 AJAX

AJAX (angl. *Asynchronous Javascript And XML*) opisuje delovanje spletnih aplikacij oziroma način komunikacije med odjemalcem in strežnikom. Pri tej komunikaciji gre za asinhrono pošiljanje in pridobivanje podatkov s pomočjo JavaScript in XML. Asinhronost zagotavlja, da se ne naloži celotna stran znova, ampak se naložijo samo delci spletne strani, kar nam zagotavlja boljšo odzivnost spletne strani. Za pošiljanje podatkov med strežnikom in odjemalcem ni nujna uporaba XML, lahko uporabimo tudi objekt JSON (angl. *JavaScript Object Notation*). Slednjo vrsto predstavitve podatkov za komunikacijo med odjemalcem in strežnikom smo uporabili tudi sami, za izdelavo spletne aplikacije. Odjemalec pošlje zahtevo, strežnik na podlagi zahteve generira odgovor JSON, ki ga pošlje nazaj odjemalcu. Na podlagi odgovora se zgodi ustrezna akcija, ki običajno vključuje manipulacijo objektov DOM.

### 2.1.9 DOM

DOM (angl. *Document Object Model*) je standard, ki je neodvisen od jezika in platforme in opisuje objektni model oziroma zgradbo za predstavitev dokumentov HTML, XHTML in XML. Standardni model DOM je poenotil in olajšal delo programerjem, ki jim ni potrebno skrbeti za različne spremembe, ki so si jih izmislili načrtovalci spletnih brskalnikov. Vsak odstavek, podoba, povezava in podobni gradniki so tako predstavljeni kot individualne entitete, katere lahko naslavljamo, manipuliramo ali celo opravimo celotno zamenjavo katere od entitet.

V razviti spletni aplikaciji sestavlja vprašalnik začne na prazni strani, kjer potem poljubno dodaja posamezna vprašanja in jim spreminja lastnosti ter dodaja število možnih odgovorov. Zato je dodajanje, odstranjevanje in spreminjanje posameznih elementov drevesa DOM zelo pogost pojav. Celotna stran in logika sestavljanja vprašalnika je na podlagi dinamičnosti, ki jo



Slika 2.4: Okrnjena predstavitev drevesa DOM in njegovih objektov[6].

dosežemo z manipulacijo drevesa DOM.

## 2.2 Razvojne tehnologije v mobilni aplikaciji

V tem podpoglavju so opisane vse uporabljene tehnologije, ki smo jih uporabili za izdelavo mobilnega dela za reševanje anket.

### 2.2.1 Knjižnica JQuery mobile

Na dotik optimizirano spletno ogrodje jQuery mobile nam ponuja enoten uporabniški vmesnik na vseh priljubljenih mobilnih napravah oziroma njihovih platformah[9]. Namenjen je za izdelavo mobilnih aplikacij za pametne mobilne telefone in tablične računalnike. Popularnost uporabe knjižnice strmo narašča ob dejstvu, da ima dobro zaledje podpore skupnosti razvijalcev in da je hčerinski produkt že tako uspešne knjižnice jQuery. Razvijalci, ki so vajeni uporabe slednje knjižnice, se lahko zelo hitro priučijo uporabe tudi mobilne različice.

O popularnosti pričajo tudi gigantska podjetja ali organizacije, ki so se poslužila uporabe te knjižnice[10]:

- Univerza Stanford
- Zabavišni park Disney World v Orlando
- Švedski prodajalec hišne opreme Ikea
- Letalski prevoznik United Airlines

Prednost knjižnice je da ko uporabnik pritisne na povezavo, se vsebina naloži z asinhronim zahtevkom AJAX. Tako se izboljša odzivnost aplikacije, ker ni potrebno ob vsaki spremembi strani ponovno osvežiti celoten dokument HTML, temveč le določen del.

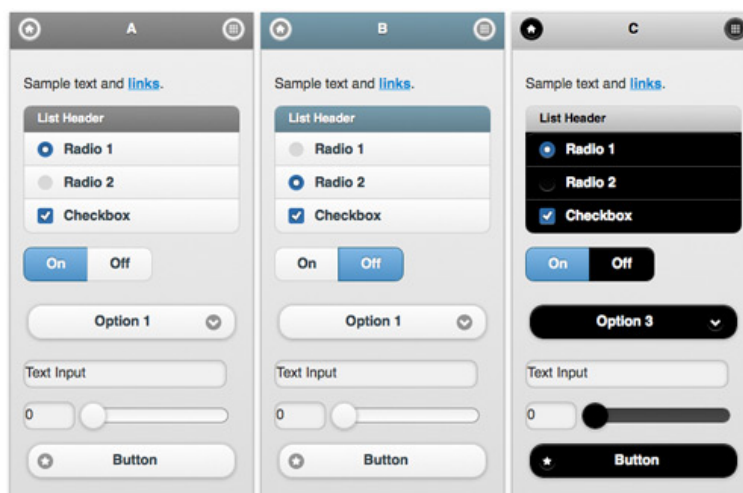
Za uporabo te knjižnice smo se odločili, ker je večina spletne aplikacije za sestavljalca vprašalnikov napisana v programskem jeziku JavaScript s knjižnico jQuery. Tako je bil potreben preskok na drugačen programski jezik minimalen, ker sta knjižnici zelo podobni. Poleg vsega podpira veliko število najbolj popularnih mobilnih operacijskih sistemov.



Slika 2.5: Podprti mobilni operacijski sistemi s strani jQuery mobile knjižnice[11].

Pomanjklivost aplikacije smo opazili na starejših pametnih mobilnih telefonih, ki nimajo najnovejše strojne opreme, kjer je bilo vidno počasnejše delovanje. Vseeno pa ekipa razvijalcev knjižnice pospešeno odpravlja te napake in z vsako novo različico, ki jo izdajo, občutno popravijo odzivnost mobilnih aplikacij na starejših mobilnih napravah.





Slika 2.6: Prikaz gradnikov HTML na mobilni aplikaciji in različne vizualne teme, ki so v sklopu knjižnice jQuery Mobile[12].

Dodatna motivacija za uporabo te knjižnice je prihranek pri času za razvoj, ker je poskrbljeno za privlačno vizualno podobo, z že pripravljenimi stilskimi podlogami CSS. Naša naloga je samo izbira primerne teme. Uporabljamo lahko tudi več tem hkrati. Tako je vizualno poskrbljeno za različne gradnike HTML, kot so povezave, potrditvena stikala (angl. *checkbox*), izbirni gumbi (angl. *radio button*).

### 2.2.2 Medplatformsko ogrodje PhoneGap

Naloga ogrodja PhoneGap je pretvoriti spletne tehnologije v domorodne (angl. *native*) aplikacije, katere lahko izvajamo na večih platformah. Z uporabo spletnih tehnologij HTML, CSS in JavaScript lahko razvijalci izdelajo mobilno aplikacijo, ki je neodvisna od platforme. Podpira veliko število operacijskih sistemov (iOS, Android, Blackberry, Palm, Windows Mobile, itd.), ki se izvajajo na mobilnih napravah.

Mobilne aplikacije tečejo v izvornem oknu brskalnika (iOS - ObjectiveC UIWebView, Android - android.webkit.WebView) in ravno to nam omogoča

razvoj s spletnimi tehnologijami[13]. Grafični vmesnik ustvari novo plast v teh izvornih oknih spletnega brskalnika in zapolni celotno dolžino in širino zaslona mobilne naprave. Vsak dokument HTML je pravzaprav svoje okno v mobilni spletni aplikaciji. Potrebno je poskrbeti tudi za vizualno privlačen grafični vmesnik s pomočjo stilskih podlog CSS. Za razvoj mobilne aplikacije smo uporabili teme, ki so del knjižnice jQuery mobile. Drugače bi morali za grafičen izgled poskrbeti sami in napisati tudi stilske podloge.

Z ogrođjem PhoneGap imamo preko programskih vmesnikov (angl. *Application Programming Interface*) dostop do virov mobilne naprave, kot so pospeškometer, kamera, kompas, kontaktni podatki, lokacija, obvestila in shramba. Kljub dejstvu, da se PhoneGap ponaša s podporo kar sedmim različnim platformam, smo pri razvoju naleteli na težavo, saj nekateri viri mobilne naprave ne delujejo na vseh platformah.

Lokalna podatkovna baza na telefonu je ena od teh virov, ki ne deluje na mobilnem operacijskem sistemu Windows. Za delovanje aplikacije na slednjem bi morali razviti dodaten mehanizem, ki bi uporabljal drugačen način shranjevanja. Poleg tega je programski vmesnik za podporo lokalne podatkovne baze (*Web SQL Database*) zastarel in ga je organizacija WC3 leta 2010 odstranila iz spiska aktivno vzdrževanih projektov[14].

### 2.2.3 Spletni programski vmesnik Web SQL Database

Programski vmesnik v ogrođju PhoneGap temelji na standardu WC3 Web SQL Database. Podatkovna baza Web SQL je spletni programski vmesnik za shranjevanje podatkov v podatkovno bazo, nad katero lahko izvajamo poizvedbe SQL. Spletni programski vmesnik je podprt s strani večjih brskalnikov kot so Chrome, Opera in Safari. Spletni brskalnik Mozilla Firefox programski vmesnik podpira zgolj s pomočjo namestitve dodatka.

Razvijalci slednjega spletnega brskalnika so javno oznanili, da programskega vmesnika načrtno niso vključili v njihovo različico, ker ga smatrajo kot problematičnega, saj potrebuje dostojno alternativo. Tu se pojavi spletni programski vmesnik Indexed Database, katera podpora je vključena pri-

marno v spletnem brskalniku Firefox[15]. Tehnologija je zaenkrat še v poskusni fazi in še ni standardizirana. Ena izmed ciljnih mobilnih naprav za katere smo razvijali aplikacijo je tudi iPhone, ki uporablja spletni brskalnik Safari, ki pa zaenkrat še ne podpira omenjene alternative.

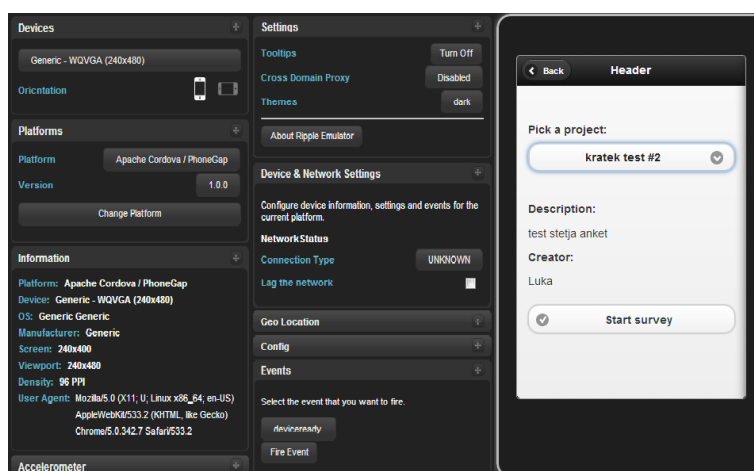
Najbolj preprosta realizacija dolgoročne shrambe podatkov, ki bi omogočila shranjevanje večjih količin podatkov je lokalna shramba (angl. *local storage*). Deluje na preprost način shranjevanja parov ključev in pripadajočih vrednosti. Podprta je v celoti v vseh novejših spletnih brskalnikih in je najbolj preprosta za uporabo. Dostop do podatkov ni mogoč s uporabo poizvedb SQL in nikakor ni primerljiva zmogljivostim, ki jih nudi podatkovna baza.

Čeprav standard Web SQL nima več zagotovljene podpore in vzdrževanja, smo se zaradi večih razlogov odločili za uporabo le tega. Delovanje na najbolj popularnih mobilnih operacijskih sistemih Android in iOS, princip podatkovne baze in podpora ogrodja PhoneGap so prednosti, ki so najbolj vplivale na izbiro.

## 2.3 Razvojna orodja

### 2.3.1 Integrirano razvojno okolje Eclipse

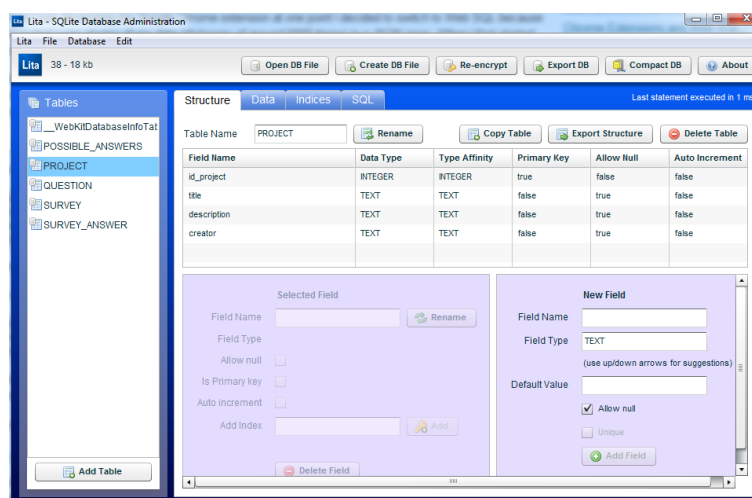
Za razvoj smo uporabljali odprtokodno in brezplačno integrirano razvojno okolje Eclipse. Z dodatki podpira široko paleto programskih jezikov (Java, Perl, PHP, JavaScript, HTML, Python, R, Ruby, itd.). Deluje na več operacijskih sistemih. Omogoča nam vse potrebne operacije, ki so potrebne za kvalitetno izdelavo in razvoj programskih aplikacij v različnih programskih jezikih. Poleg osnovnih operacij kot so pisanje in urejanje programske kode nam omogoča tudi razhroščevanje (angl. *debugging*) in prevajanje programske kode.



Slika 2.7: Glavni elementi posnemovalnika Ripple in operacije, ki jih uporabnik lahko izvede.

### 2.3.2 Ripple emulator

Posnemovalnik mobilnega okolja Ripple Emulator nam omogoča pregled mobilnih aplikacij v spletnem brskalniku. Tako nam za samo preverjanje delovanja in grafičnega vmesnika mobilne aplikacije ni potrebno vsakič znova aplikacijo nameščati na mobilno napravo. Namenjen je za testiranje mobilnih aplikacij, ki so grajene na ogrodju PhoneGap, pravitako pa podpira ogrodje za izdelavo mobilne aplikacije s spletnimi tehnologijami Blackberry WebWorks. Podpira veliko število različnih zaslonskih dimenzij pametnih mobilnih telefonov in tabličnih računalnikov. Za boljšo predstavo omogoča vertikalno in horizontalno postavitev mobilne naprave. V posnemovalniku lahko tudi imitiramo našo lokacijo in preverjamo obnašanje mobilne aplikacije ob spremembi lokacije uporabnika. Na voljo je tudi pošiljanje sprožilcev posnemovalniku za različne dogodke, kot so kliki na razne gumbe. Podpira tudi posnemanje omrežja naprave, senzorja GPS in pospeškomera (angl. *accelerometer*).



Slika 2.8: Orodje Lita za administracijo podatkovnih baz SQLite.

### 2.3.3 Lita

Majhen, zastonjski in odprtokoden program za administracijo podatkovnih baz SQLite. Preprost grafični vmesnik, ki nam omogoča vse potrebno za večino osnovnih operacij, ki so možne na podatkovnih bazah[16]. Orodje smo uporabili zaradi lažjega razhroščevanja in odkrivanja napak, ki so se porajale pri izdelavi lokalne podatkovne baze Web SQL.

### 2.3.4 Orodje za modeliranje podatkovnih baz SQL Power Architect

Orodje SQL Power Architect smo uporabili za izdelavo konceptualnega modela podatkovne baze, ki je viden v razdelku 3.2.3. Omogoča povezavo do podatkovnih baz s tehnologijo JDBC (angl. *Java Database Connectivity*). Z orodjem je mogoča izdelava skript SQL iz konceptualnega modela. Možna je tudi primerjava konceptualnega modela izvorne podatkovne baze z modelom, ki ga ustvarimo v programu. Po primerjavi lahko tudi sinhroniziramo izvorno podatkovno bazo z zgrajenim konceptualnim modelom, kjer nam program generira poizvedbe SQL na podlagi razlik v modelih.

# Poglavje 3

## Sestavljalac vprašalnikov

### 3.1 Cilji in namen aplikacije

Namen izdelave spletne aplikacije je izdelava orodja, katera bo omogočala oskrbniku oziroma sestavljalcu vprašalnikov enostavno vnašanje in urejanje vprašalnikov. Sestavljalac vprašalnik gradi dinamično in ima na voljo različne tipe vprašanj, pri primernih tipih tudi definiranje različno število možnih odgovorov. Na voljo je tudi preprosto urejanje vrstnega reda vprašanj znotraj vprašalnika. Doseči smo hoteli, da sestavljalci naše orodje uporabljajo že pri samem začetku projekta, kjer vprašalnik še ni v celoti določen in se lahko na podlagi dobrega grafičnega vmesnika lažje predstavljajo, kako bo vprašalnik izgledal in kakšen je tok oziroma potek vprašanj. Poleg sestavljanja vprašalnikov spletna aplikacija omogoča pregled nad vsemi vprašalniki in možnost brisanja in urejanja teh vprašalnikov. Zbrane podatke od terenskih anketarjev, ki jih pridobijo z anketiranjem na mobilni napravi, si sestavljalac lahko ogleda v pregledni obliki grafov.

Torej spletna aplikacija sestoji iz različnih strani oziroma modulov, katerih naloge so:

- Sestavljanje novega vprašalnika in urejanje že izdelanih.
- Izdelava predlog vprašanj in odgovorov, ki se pogosto uporabljajo.

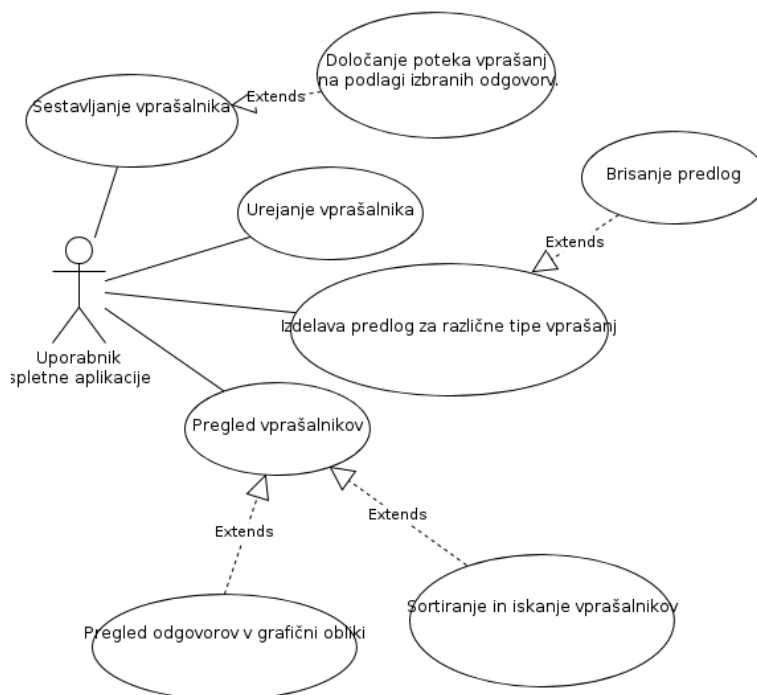
- Sprememba toka vprašalnika na podlagi respondentovih odgovorov.
- Pregled vprašalnikov in možnost brisanja zastarelih vprašalnikov.
- Grafičen prikaz zbranih odgovorov.

## 3.2 Analiza

### 3.2.1 Zahteve

Preden smo pričeli z izdelavo aplikacije smo zbrali določene zahteve, katere smo poskušali doseči:

- Spletna aplikacija, ki deluje na večini najbolj popularnih spletnih brskalnikih.
- Enostaven in intuitiven uporabniški vmesnik.
- Dodajanje različnih tipov vprašanj.
- Vrstni red vprašanj se lahko menja že med samo izdelavo vprašalnika.
- Omogočiti sestavljalcu dinamično dodajanje vprašanj in možnih odgovorov pri vprašanjih, kjer je potrebno podati možne odgovore.
- Pomnjenje stanja vprašalnika.
- Sestavljalac lahko z enim klikom v vprašalniku izbere ponavljajoča se vprašanja in pripadajoče možne odgovore in vprašanje se dinamično generira.
- Pregleden nadzor nad vprašalniki.
- Iskanje in sortiranje vprašalnikov.
- Vpogled v zbrane odgovore respondentov v grafični obliki.
- Spreminjanje poteka vprašalnika na podlagi respondentovega odgovora.



Slika 3.1: Diagram primera uporabe spletne aplikacije.

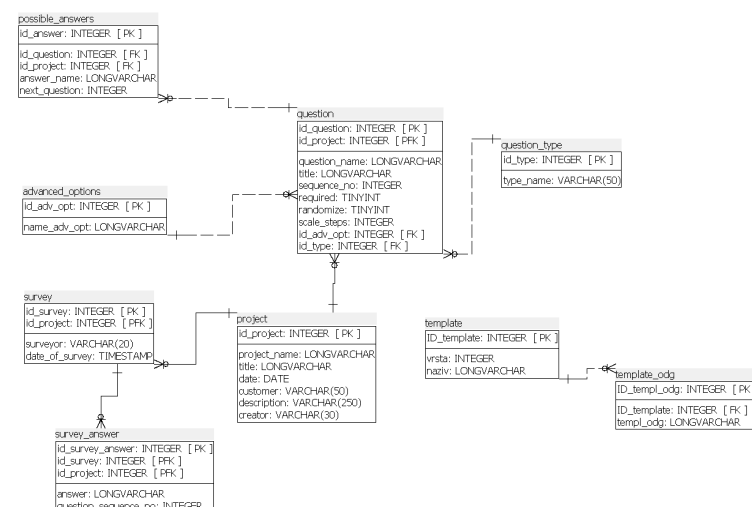
### 3.2.2 Primer uporabe

Diagram primera uporabe je vizualna predstavitev možnih interakcij uporabnika z zunanjimi sistemi. Orisuje na kakšne načine lahko uporabnik oziroma akter komunicira z sistemom, da bi lahko izpolnil določen cilj. Slika 3.1 prikazuje možne interakcije uporabnika s spletno aplikacijo.

### 3.2.3 Konceptualni model podatkovne baze

Relacijsko modeliranje je namenjeno za predstavitev strukture podatkovne baze in obstajajo različni načini zapisa pri relacijskem modeliranju[17]. Konceptualni nivo je predstavitev celotne strukture podatkovne baze in vsebuje informacije o vsebini podatkovne baze. Neodvisen je od posamezne vrste podatkovne baze. Konceptualni model za spletno aplikacijo je viden na sliki 3.2.





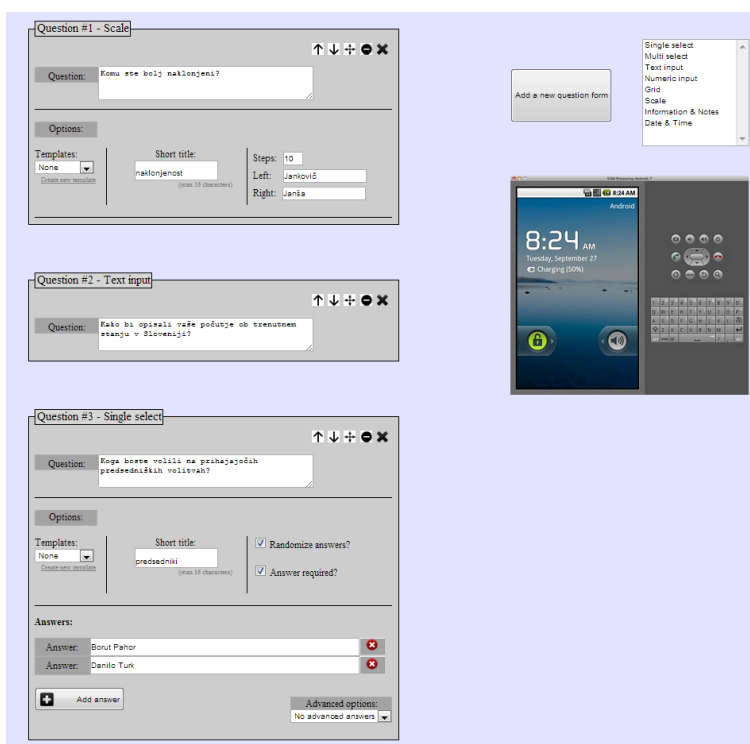
Slika 3.2: Konceptualni model podatkovne baze spletne aplikacije.

Središče celotne strukture podatkovne baze je entiteta projekt, ki hrani osnovne podatke o imenu projekta, nazivu, datumu, naročniku, opisu in lastniku projekta. Deluje tudi kot povezovalni most med vprašanji in rešenimi anketami. Vprašanje ima vrsto atributov, ki ga opisujejo. Poleg samega besedila vprašanja še tip vprašanja, kratek naziv, sekvenčna številka vprašanja in ali je odgovorer na vprašanje nujen. Vsako vprašanje ima možno eno ali več odgovorov. Pri vprašanjih odprtega tipa, kjer respondent odgovori na vprašanje z besedilom je odgovor lahko karkoli. Če je vprašanje zaprtega tipa, kjer anketar lahko označi en odgovor ali več je teh možnih odgovorov seveda več. Konceptualni model vsebuje tudi dve entiteti v kateri se bodo zapisovale predloge vprašanj in odgovorov.

Entiteti anketa (angl. *survey*) in odgovori na anketo (angl. *survey answers*) sta namenjeni za zbiranje odgovorov respondentov, ki jih zbere anketar preko mobilne naprave. Več o samem shranjevanju in sinhronizaciji podatkovnih baz je zapisano v poglavju 4.

## 3.3 Sestavljanje novega vprašalnika

V razdelku so opisane lastnosti modula oziroma strani za sestavljanje novega vprašalnika. Snovalcu vprašalnika omogoča dodajanje različnih tipov vprašanj in definiranje lastnosti vprašanj. Primer že zgrajenega vprašalnika za lažjo predstavo je viden na sliki 3.3.



Slika 3.3: Primer vprašalnika s tremi vprašanji.

### 3.3.1 Vsebniki in medsebojna interakcija

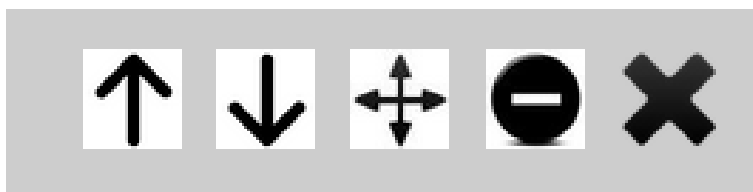
Vsebnik (angl. *container*) je osnovni element spletne strani namenjene snovalcu vprašalnika. Spletna stran za izdelavo vprašalnika je sestavljena iz večih vsebnikov, katere dinamično generira uporabnik. Vsak vprašalnik ima en statičen vsebnik, kjer snovalec zapiše informacije o projektu, ostali vsebniki so dodani poljubno. Vsebniki, ki jih dodaja uporabnik so dejansko

različni tipi vprašanj in pripadajoče lastnosti. Na sliki 3.4 je prikazan primer uporabniško dodanega vsebnika.

The image shows a web-based interface for editing a 'Single select' question. The title is 'Question #2 - Single select'. The question text is 'Koliko je ena plus dva?'. Below the question, there are several configuration options: 'Options:' section with 'Templates:' (set to 'None'), 'Short title:' (set to 'racun'), 'Randomize answers?' (checkbox), and 'Answer required?' (checkbox). The 'Answers:' section contains three entries: 'Answer: ena', 'Answer: dva', and 'Answer: tri', each with a red 'X' delete button. At the bottom, there is an 'Add answer' button and an 'Advanced options:' dropdown menu set to 'No advanced answers'.

Slika 3.4: Primer uporabniško dodanega vsebnika za vprašanje zaprtega tipa, kjer je na voljo samo en odgovor.

Z uporabo orodne vrstice lahko uporabnik manipulira z vsebniki. Na voljo ima upravljanje z vrstnim redom, minimiziranje in brisanje vsebnikov. Uporabnik lahko na vrstni red vsebnikov vpliva preko enostavnega klika na enega od gumbov, katera imata narisan puščico v orodni vrstici, vidni na sliki 3.5. Klik na ta gumb pomakne vsebnik s preprosto animacijo za en položaj višje ali nižje. Opravi se zamenjava vsebnikov in tisti, ki je bil na določenem položaju, se premakne na položaj starega vsebnika iz katerega je bila klicana menjava položaja. V primeru da želi uporabnik vsebnik premakniti na položaj, ki je bolj oddaljen od trenutnega lahko s klikom na srednji gumb v orodni vrstici aktivira stanje primi in spusti (angl. *drag and drop*). To funkcionalnost smo dosegli s pomočjo knjižnice jQuery UI, ki je



Slika 3.5: Orodna vrstica vsebnika.

namenjena oblikovanju grafičnega vmesnika za spletne strani. Implementacija stanja primi in spusti je zelo preprosta in jo dosežemo tako, da kličemo metodo `sortable("enable")` na objektu DOM. Ta metoda vpliva na celotno območje objekta DOM in tako s klikom kamorkoli znotraj vsebnika omogoči stanje primi in spusti. To smo priredili tako, da se stanje omogoči takrat, ko uporabnik pritisne na srednji gumb v orodni vrstici, onemogoči se ko uporabnik uspešno zamenja položaj vsebnikov. Dodatki, ki smo jih napisali v kodi so vidni na sliki 3.6.

```
// DRAG HOLDER EVENT
$("#drag_holder"+$counter_g).on("mousedown",function(){
  // Ob kliku na gumb omogočimo drag&drop funkcionalnost, drugače je onemogočena.
  $("#holder_container").sortable("enable");
  $("#holder_container").sortable({ opacity: '0.8',
    revert:500,
    update: function(){
      // Ko se menjajo pozicije holderjev, osveži in spremeni številke vprašanj v legendi
      refresh_legend();
      // Ko se položaja vsebnika zamenjata zopet onemogoči drag&drop funkcionalnost
      $("#holder_container").sortable("disable");
    }
  });
});
```

Slika 3.6: Omogočanje stanja primi in spusti ob kliku na srednji gumb v orodni vrstici.

S klikom na gumb za minimizacijo vsebnika se okno skrči. Tako imamo večji pregled nad vsebniki in lažje opravljamo tudi zamenjavo vrstnega reda vsebnikov. Klik na zadnji desni gumb v orodni vrstici vsebnik v celoti odstrani iz vprašalnika.

### 3.3.2 Projekt

Projekt je prvi vsebnik v vsakem vprašalniku. Vsak vprašalnik ga mora imeti in snovalec tako poda osnovne informacije kot so naziv, opis in naročnik projekta. Vnešeni podatki morajo biti čim bolj nazorni, ker so informacije o projektu vidne pri pregledu vprašalnikov in v mobilni aplikaciji, kjer anketarji prenašajo vprašalnike na lokalno podatkovno bazo na mobilnem telefonu.



The image shows a screenshot of a web form titled "Project information". It contains three input fields:

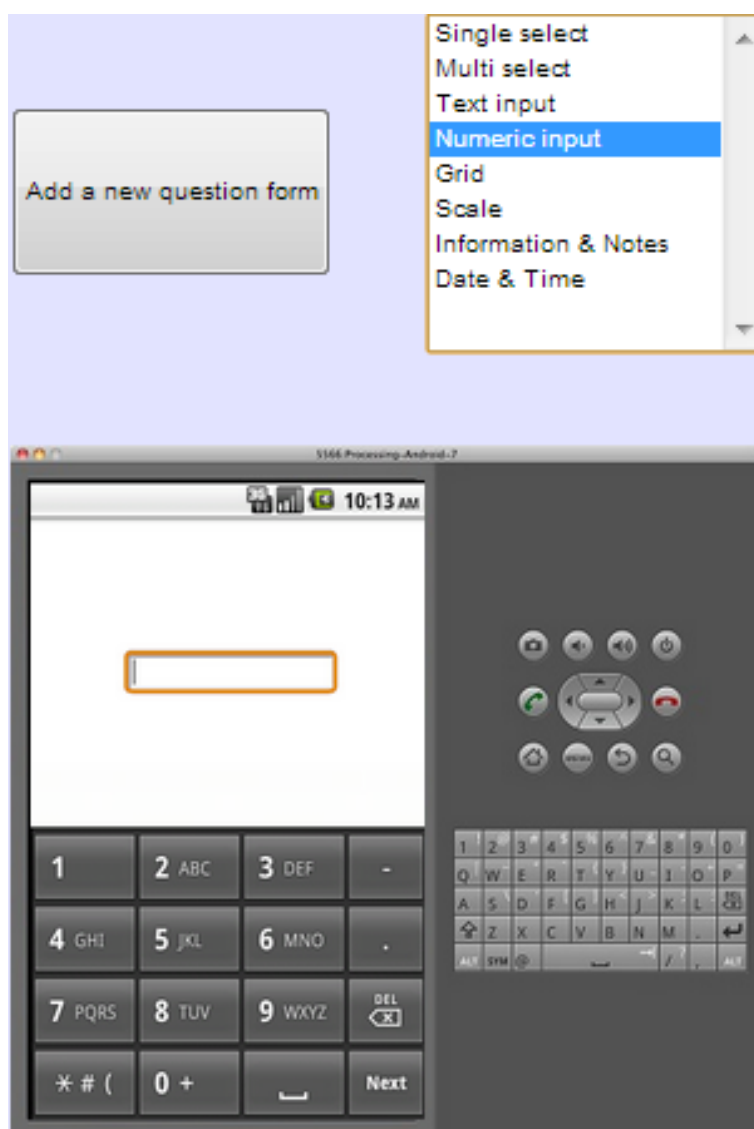
- Project title:** Predsedniške volitve 2012
- Project description:** Vprašalnik volitev za predsednika Republike Slovenije v letu 2012.
- Customer name:** Vlada RS

Slika 3.7: Primer vsebnika za vnos osnovnih podatkov o projektu.

### 3.3.3 Različni tipi vprašanj

Snovalec ankete ima na voljo dodajanje različnih tipov vprašanj, ki jih ločimo v dve vrsti. Vprašanja odprtega tipa so vprašanja, kjer respondent nima na voljo množice odgovorov na podlagi katerih se odloči in je lahko odgovor karkoli. Vprašanja zaprtega tipa imajo omejeno množico odgovorov, ki jih definira sestavljalca in respondent mora izbrati eno od ponujenih možnosti. Posebnost so vprašanja z omejeno množico odgovorov, kjer lahko snovalec vprašalnika doda na izbiro možnost "drugo" in dodatno vnosno polje, kamor anketar zapiše respondentov odgovor.

Uporabniku smo poskušali olajšati in vizualno predstaviti vsak tip vprašanja. Tako se na zaslon ob kliku na določen tip vprašanja, ki ga želi snovalec dodati v vprašalnik, prikaže slikovna predstavitev tipa vprašanja.



Slika 3.8: Slikovni predogled in predstavitev tipov vprašanj in izbirno polje preko katerega lahko uporabnik dodaja vprašanja.

Ko se uporabnik odloči za izdelavo vprašalnika začne praktično skoraj s prazno stranjo. Prikazan je zgolj vsebnik, kamor sestavljalec vnese podatke o projektu in izbirni meni, kjer izbere tip vprašanja, ki ga želi dodati v vprašalnik.

Pri **vprašanjih odprtega tipa** ločimo vsebnike glede na željeno obliko

odgovora na vprašanje. Odgovor je lahko v numerični ali znakovni obliki. Pri zahtevanem numeričnem odgovoru na mobilnih napravah ob kliku na vnosno polje prikažemo številčno tipkovnico, medtem ko pri zahtevanem znakovnem odgovoru na mobilni napravi prikažemo znakovno tipkovnico.

The image displays two screenshots of a survey editor interface. The top screenshot is titled "Question #3 - Text input". It shows a question field with the text "Kako vam je ime?". Below the question field, there are options for "Templates" (set to "None"), "Short title" (set to "ime" with a "(max 16 characters)" note), and a checked checkbox for "Answer required?". The bottom screenshot is titled "Question #4 - Numeric input". It shows a question field with the text "Koliko ste stari?". Below the question field, there are options for "Templates" (set to "None"), "Short title" (set to "starost" with a "(max 16 characters)" note), and an unchecked checkbox for "Answer required?". Both screenshots include a toolbar with icons for up, down, plus, minus, and close.

Slika 3.9: Numerično in znakovno vprašanje odprtega tipa.

Pri **vprašanjih zaprtega tipa** je respondentu na voljo izbira izmed končne množice odgovorov. Odgovori so lahko podani kot znakovni ali v obliki skale. Pri znakovnih snovalec vprašalnika določi dva ali več možnih odgovorov izmed katerih respondent izbira. Zaradi različnih dimenzij zaslonov mobilnih naprav, preko katerih lahko anketar opravlja ankete se za implemen-

tacijo mrežnega prikaza nismo odločili, ker bi lahko povzročalo preveč težav. Mrežni prikaz je primeren kadar imamo eno vprašanje in vrsto možnih odgovorov, respondent pa se za vsak odgovor opredeljuje na lestvici (od ena do pet, nezadovoljen do zelo zadovoljen). Tak tip vprašanj je v razviti aplikaciji možno simulirati z malce več truda, kjer se vprašanja ponavljajo, nanizamo pa različne odgovore in pripadajočo lestvico. Snovalec lahko določi da je možen zgolj in le en odgovor (angl. *single select*), lahko pa je možnih več odgovorov (angl. *multi select*).



Question #5 - Multi select

Question: kateri stranki ste najbolj naklonjeni?

Options:

Templates: None

Short title: stranke (max 16 characters)

Randomize answers?  
 Answer required?

Answers:

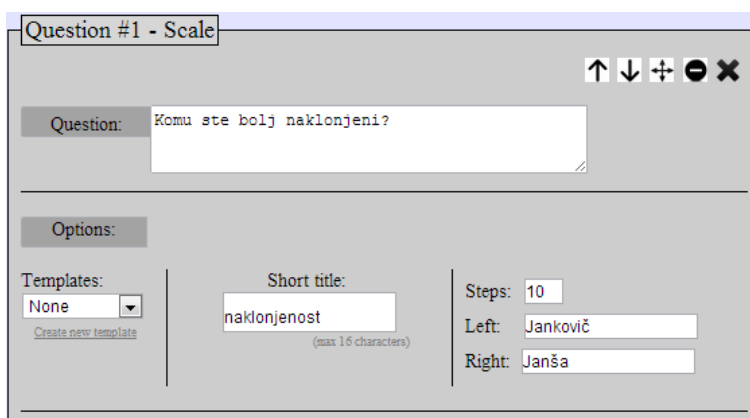
Answer:	Združeni zeleni	<input type="checkbox"/>
Answer:	Zares - socialno liberalni	<input type="checkbox"/>
Answer:	Stranka za skladen razvoj	<input type="checkbox"/>
Answer:	Slovenski krščanski demokrati - SKD	<input type="checkbox"/>
Answer:	Socialni demokrati SD	<input type="checkbox"/>
Answer:	Radikalna stranka Slovenije	<input type="checkbox"/>
Answer:	Primorska je naša	<input type="checkbox"/>
Answer:	Premik	<input type="checkbox"/>
Answer:	Pozitivna Slovenija	<input type="checkbox"/>
Answer:	Piratska stranka Slovenije	<input type="checkbox"/>
Answer:	Lista za čisto pitno vodo	<input type="checkbox"/>

Pick one of the additional advanced options, you may only chose one!

Advanced options:

Slika 3.10: Znakovno vprašanje odprtega tipa, kjer je na voljo eden ali več možnih odgovorov.

Če je odgovor v obliki skale ima respondent možnost izbire odgovora, kjer ima omejen numerični interval od neke začetne do končne številke. Snovalec določi levo in desno robno vrednost skale in respondent se odloči na katero stran in kako močno se nagiba na katero od strani. Če je izbrana sredina skale pomeni, da je odgovor nevtralen in da se ne nagiba k nobeni robni vrednosti.



The screenshot shows a window titled "Question #1 - Scale". At the top right, there are navigation icons: up, down, plus, minus, and close. The main area contains a "Question:" label followed by a text input field with the text "Komu ste bolj naklonjeni?". Below this is an "Options:" section. On the left, there is a "Templates:" dropdown menu currently set to "None" with a "Create new template" link below it. In the center, there is a "Short title:" label followed by a text input field containing "naklonjenost" and a note "(max 16 characters)". On the right, there are three input fields: "Steps:" with the value "10", "Left:" with the value "Jankovič", and "Right:" with the value "Janša".

Slika 3.11: Vprašanje zaprtega tipa v obliki skale.

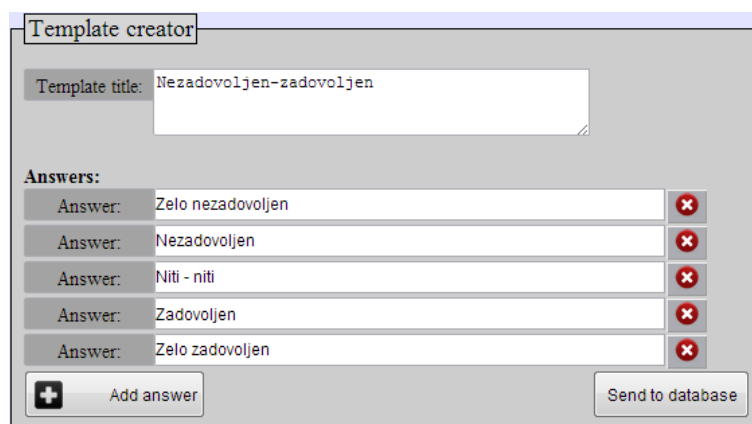
Naslednji tip je **informacija anketarju**, kjer sestavljalec na določenem mestu v vprašalniku zapiše kakršnekoli informacije anketarju, če želi nekaj poudariti, opomniti anketarja ali podati kakšno posebno navodilo. Tak tip nima možnosti respondentovega odgovora in zato tudi nima vnosnih polj. Služi zgolj za informiranje anketarja.

### 3.3.4 Predloge različnih tipov vprašanj

Pri sestavljanju vprašalnikov opazimo da se določeni odgovori in vprašanja ponavljajo. Tako se kot primer vprašanje o starosti pojavi skoraj v vsakem vprašalniku zaradi statističnega pregleda nad strukturo vzorca anketirancev. Razvili smo sistem predlog, ravno zaradi ponavljajočih se vprašanj in odgovorov. S to funkcionalnostjo lahko snovalec vprašalnikov prihrani ogromno časa.

Za vsak tip vprašanja smo izdelali stran, kjer lahko sestavljalec ankete sestavi novo predlogo. Kreirana predloga je možna za izbiro le v vsebniku točno določenega tipa, za katerega je predloga bila sestavljena. V predloge namenjene za vprašanja zaprtega tipa lahko snovalec zapiše možne odgovore. Za vprašanja odprtega tipa sestavljalec zabeleži ponavljajoča se vprašanja.

Ko je uporabnik zadovoljen z novo nastalo predlogo se podatki zapišejo



Template creator

Template title: Nezadovoljen-zadovoljen

Answers:

Answer:	Zelo nezadovoljen	X
Answer:	Nezadovoljen	X
Answer:	Niti - niti	X
Answer:	Zadovoljen	X
Answer:	Zelo zadovoljen	X

+ Add answer

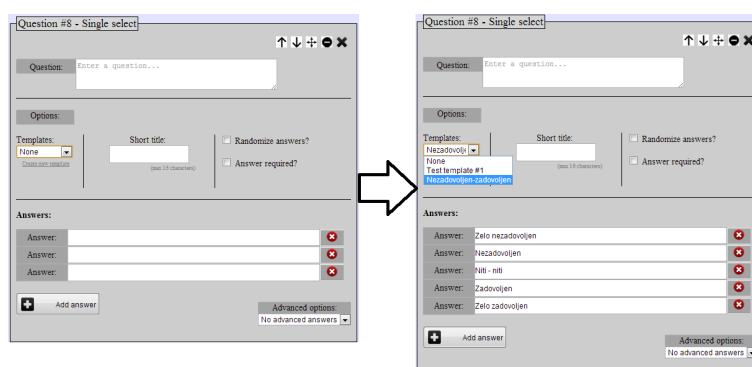
Send to database

Slika 3.12: Kreiranje nove predloge za vprašanje zaprtega tipa z možnim enim odgovorom.

v podatkovno bazo in sestavljalec lahko predlogo že uporablja pri gradnji vprašalnika. Vsak vsebnik vsebuje možnost gradnje iz predlog. Vse nazive predlog pridobimo s pomočjo klica AJAX in spletnih storitev in tako v vsakem vsebniku zgradimo spustni seznam (angl. *drop-down list*). Dodali smo poslušalca dogodkov (angl. *event listener*), ki se sproži kadar uporabnik izbere eno izmed možnosti v spustnem seznamu. Ko se zgodi sprememba v spustnem seznamu najprej pobrišemo vsebino vmesnika, kjer so zbrani odgovori ali vprašanje. Nato se zopet zgodi klic AJAX, ki tokrat od spletne storitve zahteva seznam odgovorov pripadajoče predloge v obliki JSON in na podlagi tega seznama zgradi del vsebnika, kjer so podani možni odgovori. Vsebnik odgovorov se zgradi brez potrebne osvežitve celotne strani, ker podatke predloge pridobi s tehnologijo AJAX. Na sliki 3.13 lahko vidimo uporabo predloge, ki smo jo kreirali na strani za izdelavo novih predlog, vidno na sliki 3.12.

### 3.3.5 Lastnosti vprašanj in dodatne možnosti

Vsem tipom vprašanj je skupna dodatna možnost izbire nujnosti odgovora. Z izbiro te možnosti bo moral anketar nujno označiti enega ali več možnih



Slika 3.13: Primer uporabe predlog na vprašanju zaprtega tipa.

odgovorov pri zaprtem tipu vprašanj. Pri odprtem tipu vprašanj pa vnosno polje ne sme biti prazno.

Če je vprašanje zaprtega tipa, kjer je možnost izbire en odgovor ali več ima snovalec vprašalnika možnost izbire naključnega vrstnega reda odgovorov. Pri tej izbiri se bodo odgovori na zaslonu mobilne naprave pojavljali v naključnem vrstnem redu.

Pri zaprtem tipu vprašanj je na voljo tudi izbira dodatnih možnih odgovorov, od katerih lahko uporabnik izbere le enega. Ti dodatni odgovori so drugo, nič od naštetega in drugo ter vnosno polje, kamor lahko anketar zapiše odgovor respondentu. Ravno slednja možnost, kot omenjeno pri definicij tipov vprašanj spremeni zaprt tip vprašanja v odprtega. Poleg množice možnih odgovorov, ki jih je vnesel sestavljalec vprašalnika, lahko respondent tako vnese karkoli.

### 3.3.6 Dinamična gradnja vprašalnika in postopek gradnje vsebnika

Ko se uporabnik odloči za izdelavo vprašalnika začne praktično skoraj s prazno stranjo. Prikazan je zgolj vsebnik, kamor sestavljalec vnese podatke o projektu in izbirni meni, kjer lahko izbere tip vprašanja, ki ga želi dodati v vprašalnik. Ob kliku na tip vprašanja v izbirnem meniju je uporabniku na



```

▶<fieldset id="holder2" class="holder">...</fieldset>
▼<fieldset id="holder3" class="holder">
  <legend>Question #3 - Single select</legend>
  <div id="holder_title3" class="holder_title">
    ▶<a id="remove_holder3" class="remove_holder" title="Remove whole holder." href="#">...</a>
    ▶<a id="minimize_holder3" class="minimize_holder" title="Minimize holder" href="#">...</a>
    ▶<a id="drag_holder3" class="drag_holder" href="#">...</a>
    ▶<a id="move_holder_down3" class="move_holder_down" href="#">...</a>
    ▶<a id="move_holder_up3" class="move_holder_up" href="#">...</a>
  </div>
  ▼<div id="holder_question3" class="holder_question">
    <label class="question_label"> Question:</label>
    <textarea type="text" name="question" value="Koga boste volili na prihajajočih predsedniških volitvah?" id="q3">
    <br>
  </div>
  ▼<div id="holder_options3" class="holder_options">
    ▶<div id="holder_options_top3" class="holder_options_top">...</div>
    ▼<div id="holder_options_left3" class="holder_options_left">
      " Templates:"
      ▶<select id="templates3" class="templates">...</select>
      <a href="template_maker.php?type=1&id_proj=349">Create new template</a>
    </div>
    ▶<div id="holder_options_middle3" class="holder_options_middle">...</div>
    ▶<div id="holder_options_right3" class="holder_options_right">...</div>
  </div>
  ▼<div id="holder_answer3" class="holder_answer">
    ▶<p>...</p>
    ▼<div id="ans_container30" class="ans_container">
      <label class="answerLabel"> Answer: </label>
      <input type="text" name="answer" id="q3a4" value="Milan Zver" class="ans">
      ▶<button id="btn_rem_a4" class="btn_rem_ans">...</button>
    </div>
    ▶<div id="ans_container31" class="ans_container">...</div>
    ▶<div id="ans_container32" class="ans_container">...</div>
  </div>
  ▼<div id="holder_button3" class="holder_button">
    ▶<button id="btn_add_3" type="button" class="btn_add" style="width:150px;">...</button>
  </div>
  ▼<div id="advanced_answers3" class="advanced_answers">
    <p>Advanced options: </p>
    ▼<select id="advan_answ_select3" class="advan_answ_select">
      <option value="0">No advanced answers</option>
      <option value="1">Other</option>
      <option value="2">Other + text entry</option>
      <option value="3">None of the above</option>
    </select>
  </div>
  </div>
  <span class="holder_number" style="display:none;">3</span>
  <span class="holder_type" style="display:none;">1</span>
</fieldset>
▶<fieldset id="holder4" class="holder">...</fieldset>
▶<fieldset id="holder5" class="holder">...</fieldset>
▶<fieldset id="holder6" class="holder">...</fieldset>
▶<fieldset id="holder7" class="holder">...</fieldset>
▶<fieldset id="holder8" class="holder">...</fieldset>
▶<fieldset id="holder9" class="holder">...</fieldset>
▶<fieldset id="holder10" class="holder">...</fieldset>
▶<fieldset id="holder11" class="holder">...</fieldset>
</div>

```

Slika 3.17: Prikaz celotne strukture vsebnika kot objekti DOM drevesa.

### 3.3.7 Pomnjenje stanja vprašalnika

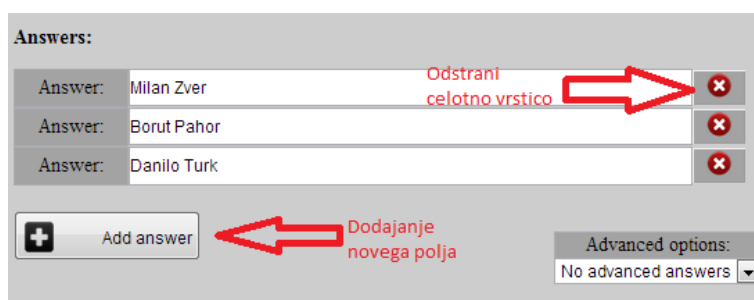
Zadostitev zahteve pomnjenja stanja vprašalnika smo realizirali z lokalno shrambo (angl. *local storage*), ki je bila predstavljena kot novost v HTML5. Podatki se shranjujejo preprosto po principu ključa in pripadajoče vrednosti. Tako lahko uporabnik brez skrbi osveži stran preko gumba v brskalniku. Sestavljalac vprašalnikov lahko stran tudi zapusti za dalj časa, podatki bodo trajno ostali v shrambi. Tako so podatki varni tudi ko uporabnik v celoti zapre spletni brskalnik. Spletna stran sestavljalac ob začetku nalaganja vedno preveri vsebino lokalne shrambe in s pomočjo metode *buildHolder*, ki je prikazana na sliki 3.14, v zanki zgradi vse vsebnike.

```
[[{"1","Koga boste volili na prihajajo\u010dih predsedni\u0161kih volitvah?","Milan Zver","Borut Pahor","Danilo Turk"},"predsedniki",  
razvoj","Slovenski kr\u0161\u010danski demokrati - SKD","Socialni demokrati SD","Radikalna stranka Slovenije","Primorska je nasa",  
naklonjenj","","naklonjenost","Jankovi\u010d","Jansa",10,0,0,0}], [{"3","Kako bi opisali vase pocutje ob trenutnem stanju v Slov  
4","Vasi mese\u010dni prihodki?","","prihodki","","","","0,0,0}], [{"1","Koliko ste stari?","15-25","25-35","35-50","50-60","60
```

Slika 3.15: Primer vrnjenih podatkov vpra\u0161alnikov od spletne storitve v obliki JSON.

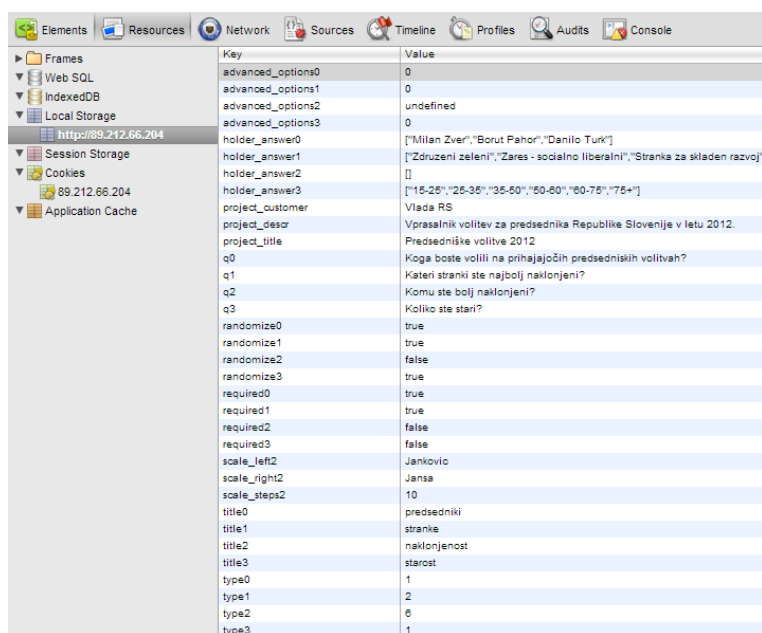
v obliki JSON. Primer vrnjenih podatkov je viden na sliki 3.15. Iz teh podatkov nato generiramo celoten vpra\u0161alnik s pomo\u010djo metode in podajanjem parametrov.

Pri vpra\u0161anjih zaprtega tipa se vnosna polja, kamor snovalec vnese mo\u017ene odgovore, generirajo dinami\u010dno. Ob kliku na dodajanje novega polja za nov mo\u017en odgovor, se na zadnje mesto v prostoru za mo\u017ene odgovore vrine novo vnosno polje. Ob kliku na izbris se odstrani celotna vrstica, v kateri je bil klik na izbris zaznan.



Slika 3.16: Dodajanje in odstranjevanje vnosnih polj za mo\u017ene odgovore znotraj vsebnika.

Vse te operacije zahtevajo spreminjanje in dodajanje DOM elementov. Zaradi dinami\u010dne izgradnje je posegov v strukturo DOM zelo veliko, posledica je slab\u0161a odzivnost aplikacije. Pri razvoju smo dinami\u010dnost postavili pred odzivnostjo spletne strani. Vseh razli\u010dnih gradnikov, ki se dinami\u010dno dodajo v strukturo DOM je precej in na sliki 3.17 je primer vpra\u0161alnika z enajstimi razli\u010dnimi vpra\u0161anji. Na sliki je prikazan tudi bolj podroben vpogled v strukturo enega vsebnika.



Slika 3.18: Pomnjenja stanja s pomočjo lokalne shrambe prikazano v spletnem razvijalskem okolju v brskalniku Chrome.

Zapisujemo si vse potrebne podatke za ponovno vzpostavitev stanja, kadar uporabnik prekine delo z zaprtjem ali osveževanjem okna v spletnem brskalniku. To storimo za vsak vsebnik. Primer shranjevanja v lokalno shrambo je prikazan na sliki 3.18.

### 3.4 Potek in sprememba toka vprašalnika

Ko končamo z gradnjo vprašalnika in podatke aplikacija zapiše v podatkovno bazo, lahko priredimo še vrstni red poteka vprašanj. Na podlagi respondentovih odgovorov lahko nekatera vprašanja preskočimo. Vprašanja med seboj niso v celoti neodvisna, zato ne bi bilo smiselno nekatera vprašanja postavljati respondentu, če odgovor na katero prejšnje vprašanje ni temu primeren. Kot primer nesmisla je vprašanje respondenta koga bi volili na predsedniških volitvah, v primeru da je ta mlajši od 18 let.



Question	Type	Answer	Next
Koga boste volili na prihajajočih predsedniških volitvah?	Single select	Milan Zver Borut Pahor Danilo Turk	2 2 2
Kateri stranki ste najbolj naklonjeni?	Multi select	Združeni zeleni Zares - socialno liberalni Stranka za skladen razvoj Slovenski krščanski demokrati - SKD Socialni demokrati SD Radikalna stranka Slovenije Primorska je naša Premik Pozitivna Slovenija Piratska stranka Slovenije Lista za čisto pitno vodo	3 3 3 3 3 3 3 3 3 3 3
Komu ste bolj naklonjeni?	Scale	Janković Jansa	4 4
Kako bi opisali vaše počutje ob trenutnem stanju v Sloveniji?	Text input		5
Sledi samo se nekaj zaključnih statističnih vprašanj.	Information & Notes		6
Vasi mesečni prihodki?	Numeric input		7
Koliko ste stari?	Single select	15-25 25-35 35-50 50-60 60-75 75+	Conclusion Conclusion Conclusion Conclusion Conclusion Conclusion

Slika 3.19: Sprememba toka vprašalnika glede na izbrane odgovore respondenta.

Za spreminjanje vrstnega reda vprašanj smo razvili stran, kjer sestavlja-  
lec lahko priredi tok vprašalnika, glede na podane odgovore respondenta.  
Za naslednje vprašanje pri vsakem možnem odgovoru lahko snovalec izbira  
vprašanja na intervalu od naslednjega vprašanja do zaključka vprašalnika.  
Tako sekvenčna številka naslednjega vprašanja ne more biti enaka sekvenčni  
številki vprašanja, ki je že bilo postavljeno. Pravtako naslednje vprašanje ne  
more imeti večjo sekvenčno številko kot zadnje vprašanje v vprašalniku. Tako  
v vprašalniku ne more priti do pojava ciklov, kjer bi se določeno zaporedje

vprašanj lahko ponavljalo v neskončnost.

## 3.5 Pregled vprašalnikov

V tem razdelku je opisana rešitev za zahtevo po enostavnem in preglednem uporabniškem vmesniku, kjer so zbrani vsi osnovni podatki o vprašalnikih. Vsem zahtevam je bilo mogoče ugoditi z vtičnikom (angl. *plugin*) *DataTables*, razvitim za knjižnico jQuery. Vtičnik ponuja tabelarno in pregledno predstavitev in osnovne operacije nad elementi tabele.

### 3.5.1 Vpogled in možne operacije nad vprašalniki

Oskrbniku vprašalnikov smo zagotovili enostaven pregled vseh zgrajenih vprašalnikov in množico različnih operacij nad njimi. Vprašalnike je možno sortirati po večih parametrih naenkrat, dodali smo zmožljivo iskanje vprašalnikov po različnih parametrih in izbiro števila vprašalnikov, ki so vidni naenkrat. To so osnovne operacije nad tabelarno prikazanimi vprašalniki. Znotraj vsake vrstice, ki predstavlja en vprašalnik, smo dodali še povezavo do urejanja in izbris le tega.

Klik na urejanje vprašalnika nas vodi do strani sestavljalca anket, kjer se dinamično generira vprašalnik. Ko vprašalnik uredimo in zapišemo v podatkovno bazo se ustvarita dva vprašalnika. Razlog je zahtevna usklajenost starih in novih odgovorov v podatkovni bazi. Vprašalnik, ki že ima nekaj zbranih odgovorov anketarja iz mobilne aplikacije, bi lahko ob pretiranih spremembah povzročal težave. Urejanje zato doda v podatkovno bazo nov vprašalnik in ne odstrani starega. Izbris starega lahko naredi oskrbnik vprašalnikov, če je prepričan da so odgovori, ki so bili zbrani za staro različico vprašalnika neuporabni.

ID	Title	Description	Creator	Last modified	Customer	No. of questions	No. of surveys made	Edit	Delete
372	test vseh tipov #2	test vseh različnih tipov	Luka	2012-10-17	abc	8	8	Edit	Delete
330	test prazen	oprazen	Luka	2012-10-17	prazen	3		Edit	Delete
342	kratk test #2	test sniga ankete	Luka	2012-10-18	sniga	2	10	Edit	Delete
343	test vseh tipov #2	test vseh različnih tipov	Luka	2012-10-20	abc	8	1	Edit	Delete
344	HP anketa	Opis ...	Luka	2012-10-20	Abakus plus d.o.o.	2	1	Edit	Delete
345	test vseh tipov #123	test vseh različnih tipov	Luka	2012-11-09	abc	8		Edit	Delete
350	Preddodna volitve 2012	Vprašnik volitev za predsednika Republike Slovenije v letu 2012.	Luka	2012-11-21	Vlada RS	7		Edit	Delete

Slika 3.20: Prikaz maske za pregled vprašalnikov in njihovo manipulacijo s pomočjo vtičnika DataTables.

Poleg osnovnih podatkov je izpisano tudi število vprašanj, ki jih ima vprašalnik. V enem stolpcu je tudi število narejenih anket v mobilni aplikaciji. Slednje je tudi povezava do strani, kjer je grafična predstavitev zbranih odgovorov.

### 3.5.2 DataTables

Vtičnik je namenjen tabelarni predstavitvi različnih podatkov in je razvit za uporabo s knjižnico jQuery[18]. Podatke mu lahko podamo na več različnih načinov (DOM, JSON, AJAX, itd.). Vtičnik je brezplačen in robusten, testiran z velikim programskim naborom testov enot (angl. *unit test*). Uporaba je zelo preprosta, podamo mu število stolpcev in dostavimo podatke s kate-regim od možnih načinov, za vse ostalo poskrbi vtičnik sam.

Primer uporabe v razviti aplikaciji je prikazan na sliki 3.21. Na voljo imamo nastavljanje nekaj lastnosti, kako naj se tabela obnaša in število stolpcev in značilnosti le teh. Potrebno je zagotoviti tudi podatke, ki jih v našem primeru pridobimo s spletne storitve v obliki JSON. Za osnovne operacije in manipulacijo s tabelo poskrbi vtičnik sam, dodali smo zgolj povezavo za urejanje projekta in slikovno povezavo za izbris projekta. Ob kliku na izbris projekta s klicem AJAX izbrisemo projekt iz podatkovne baze. V primeru uspeha nato iz tabele odstranimo še ustrezno vrstico. Zaradi asinhronosti ni potrebno osvežiti celotne tabele, tako malenkost pridobimo na odzivnosti spletne strani.

```

var oTable; // global variable for dataTable

$(document).ready(function() {

    oTable = $('#survey_table').dataTable({
        "bJQueryUI": true,
        "sPaginationType": "full_numbers",
        "bStateSave": true,
        "aoColumns": [
            { "sName": "id", "swidth": "5%", "bSortable": false, "bSearchable": false },
            { "swidth": "10%" },
            { "swidth": "25%" },
            { "swidth": "5%" },
            { "swidth": "10%" },
            { "swidth": "10%" },
            { "swidth": "3%" },
            { "swidth": "3%" },
            { "swidth": "5%" },
            { "swidth": "5%" }
        ],
        "bProcessing": true,
        "bServerSide": false,
        "sAjaxSource": "php/return_surveys_info.php"
    });
});

```

Slika 3.21: Uporaba vtičnika DataTables v spletni aplikaciji.

## 3.6 Grafična predstavitev rešenih anket

Podatki so predstavljeni v obliki grafov in tabel. Izvoz v druge oblike ni možen. Za vizualen in uporabniku prijazen vpogled na zbrane odgovore respondentov, smo uporabili orodje Google Chart.

### 3.6.1 Prikaz podatkov z uporabo Google Charts

Orodje je plod razvijalcev podjetja Google in je brezplačno, lahko za uporabo in ponuja zelo veliko oblik prikazovanja zbranih podatkov[19]. Postopek uporabe je podoben uporabi vtičnika DataTables v tem smislu, da smo orodju Google Charts dostavili podatke in orodje nato kreira poljuben grafikon in tabelo. Nekaj težav smo imeli zgolj z formatom podatkov, ki jih zahteva orodje. Striktno zahteva primerno obliko podatkov iz katerih želimo ustvariti graf. Primer poslanih podatkov spletne storitve za izdelavo enega grafa:

```

{ "cols": [
    { "id": "", "label": "Answer", "type": "string" },
    { "id": "", "label": "Answer count", "type": "number" }
  ],
  "rows": [

```

```

    {"c":[{"v": " Borut Pahor " }, {"v": 1}]}],
    {"c":[{"v": " Danilo Turk " }, {"v": 1}]}],
    {"c":[{"v": " Milan Zver " }, {"v": 2}]}],
  ] }

```

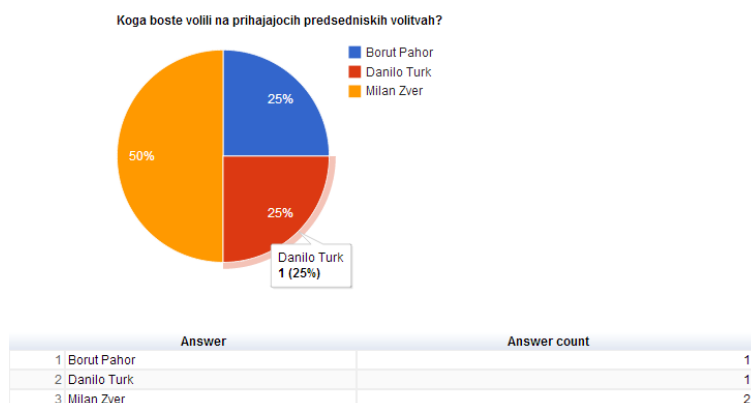
Ko so podatki v primerni obliki je izdelava grafa in pripadajoče tabele pogostosti pojavitve odgovorov preprosta. To dosežemo s klicem metode *draw*, kateri podamo omenjene podatke in lastnosti. Podati moramo še element DOM, kjer želimo da se graf ustvari. Primer programske kode za izdelavo krožnega grafikona (angl. *pie chart*) s podatki, ki smo jih omenili zgoraj:

```

var chart = new google.visualization.PieChart(document.getElementById(
'chart_div'+sequence_no));
chart.draw(data, {width: 600, height: 380, title:question});

```

Na sliki 3.22 je viden grafični prikaz v obliki krožnega grafikona, ki smo ga dosegli z danimi podatki in klicem metode *draw*. Grafi so interaktivni in s klikom na katero izmed polj na grafikonu se nam prikaže odstotek pogostosti in število pojavitev vsakega od odgovorov. Poleg grafičnega prikaza je zgrajen tudi prikaz pojavitev odgovorov v tabeli. Tudi tabela je interaktivna, klik na katerega od stolpcev nam uredi podatke po vrsti padajoče ali naraščajoče.



Slika 3.22: Prikaz grafa ustvarjenega z orodjem Google Chart.

# Poglavje 4

## Mobilna aplikacija za reševanje anket

### 4.1 Osnovne zahteve in cilji mobilne aplikacije

Mobilno aplikacijo smo razvili za namene reševanja vprašalnikov, ki jih sestavi snovalec vprašalnikov v spletni aplikaciji. Podrobne analize ni bilo narejene, poglavitna zahteva je enostavno pridobivanje in reševanje vprašalnikov.

Osnovne zahteve, ki smo jih pred izdelavo zastavili so:

- Možnost reševanja vprašalnikov brez internetne povezave.
- Pošiljanje zbranih odgovorov, kadar ima anketar dostop do spleta.
- Prijazen in enostaven uporabniški vmesnik.
- Delovanje na večih operacijskih sistemih.
- Prilagajanje uporabniškega vmesnika na različne dimenzije zaslonov mobilnih naprav.
- Reševanje vprašalnika in podpora vsem različnim tipom vprašanj.

- Upoštevanje toka vprašalnika glede na podane odgovore respondenta.
- Implementacija naključnega vrstnega reda odgovorov in nujnost izbire odgovora, kjer je sestavljalca to zahteval.

## 4.2 Realizacija osnovnih zahtev

Delovanje na večih platformah smo dosegli z uporabo ogrodja PhoneGap, ki podpira veliko število najbolj popularnih mobilnih operacijskih sistemov. Medplatformsko ogrodje je bolj podrobno opisano v razdelku 2.2.2. Za lažjo izdelavo uporabniškega vmesnika mobilne aplikacije smo se zanesli na uporabo knjižnice jQuery Mobile.

Za doseg zahteve izpolnjevanja vprašalnikov brez internetne povezave smo razvili stran, kjer se uporabnik z internetno povezavo poveže do zunanjega strežnika in podatkovne baze iz katere se zgradi seznam vseh vprašalnikov. Uporabnik ima na voljo prenos vprašalnikov, kjer se slednji shranijo v lokalno podatkovno bazo. Poleg prenosa ima na voljo tudi izbris vprašalnika iz lokalne podatkovne baze. Ko je vprašalnik zapisan v lokalni podatkovni bazi, lahko anketar rešuje vprašalnik z različnimi respondenti in zbira odgovore. Ko ima uporabnik možnost povezave do spleta zbrane odgovore na uporabnikovo zahtevo pošlje v oddaljeno podatkovno bazo.

Za reševanje vprašalnika smo razvili stran, kjer uporabnik izbere kateri vprašalnik želi reševati izmed tistih, ki jih je prenesel v lokalno podatkovno bazo. Gradnja trenutnega vprašanja se dinamično generira iz podatkov, ki jih pridobimo iz podatkovne baze, glede na trenutno sekvenčno številko vprašanja in tip vprašanja. Tekom reševanja vprašalnika si beležimo trenutno sekvenčno številko vprašanja v vprašalniku, naslednje vprašanje in izbrane odgovore respondenta.

## 4.3 Povezani način (*Online mode*)

Način v mobilni aplikaciji, kadar ima uporabnik možnost dostopa do spleta. Namenjen je povezovanju do oddaljenega spletnega strežnika in prenosu vprašalnikov ter pošiljanju zbranih odgovorov respondentov.

### 4.3.1 Sinhronizacija vprašalnikov iz oddaljene baze podatkov

Sinhronizacija vprašalnikov je na zahtevo uporabnika. Zaradi boljše odzivnosti mobilne aplikacije in varčevanja pri prenosu podatkov uporabniku dopuščamo izbiro, kdaj želi prenesti vprašalnike. Pravtako ima na izbiro katere vprašalnike želi v danem trenutku prenesti, ker zelo verjetno vseh ne potrebuje naenkrat.

Za samo sinhronizacijo smo razvili stran oziroma modul v mobilni aplikaciji, kjer se najprej prenesejo osnovni podatki o projektih. Izgled strani je prikazan na sliki 4.1. Na podlagi podatkov o projektih se zgradi seznam vseh projektov. Pri gradnji seznama za vsak projekt preverimo, če se nahaja v lokalni podatkovni bazi. V primeru da se nahaja, projektu v seznamu dodamo gumb za izbris projekta iz lokalne baze. Če projekta v lokalni podatkovni bazi še ni, mu dodamo gumb za prenos vprašalnika in zapis v lokalno podatkovno bazo.





Slika 4.1: Seznam projektov in možnosti prenosa in izbrisa iz lokalne podatkovne baze.

V namen shranjevanja podatkov na telefonu za trajno shrambo smo izdelali identično strukturo podatkovne baze tisti, ki se nahaja na spletnem strežniku. Ko uporabnik zahteva prenos nekega projekta, se ti podatki shranijo v lokalno podatkovno bazo, ki je shranjena na mobilni napravi in je

dostopna tudi brez povezave. Lokalno bazo smo kreirali s pomočjo ogrodja PhoneGap, ki uporablja spletni programski vmesnik Web SQL Database.

Ko uporabnik zahteva prenos nekega projekta se najprej pošlje zahteva AJAX spletni storitvi, ki vrne podatke o zahtevanem projektu v obliki JSON. Najprej pridobimo podatke o projektu in jih zapišemo v lokalno podatkovno bazo. Sledi zahteva spletni storitvi za vsa vprašanja in pripadajoče lastnosti in ponoven zapis v lokalno podatkovno bazo. Postopek se ponovi še za možne odgovore za vsako izmed vprašanj. Primer programske kode za zapis vseh vprašanj enega projekta v lokalno podatkovno bazo:

```
function writeQuestionsToLocalDatabase(data){
    $.each(data,function(index,question){
        localDB.transaction(
            function (tx) {
                tx.executeSql("INSERT INTO QUESTION VALUES(?,?,?,?,?,?,?,?,?,?,?)",
                    [question.id_question,
                    question.id_project,question.question_name,question.title,
                    question.sequence_no,question.required,question.randomize,
                    question.scale_steps,question.id_adv_opt,
                    question.id_type]
                    ,successCB("stored questions from remote to local"),errorCB);
            }
        );
    });
}
```

V zgornjem primeru programske kode za zapis vseh vprašanj metoda sprejme tabelo objektov *data*. Vsak element tabele vsebuje en objekt. Objekt predstavlja eno vprašanje, vse njegove lastnosti in kateremu projektu pripada. Predstavitev kaj dejansko vsebuje spremenljivka in kaj vsebuje objekt je vidno na sliki 4.2, katera je narejena v razvijalskem orodju spletnega brskalnika Chrome. Vsa vprašanja so del enega projekta oziroma so gradniki enega vprašalnika.

```

▼ [Object, Object, Object, Object, Object, Object, Object]
  ▼ 0: Object
    id_adv_opt: "0"
    id_project: "350"
    id_question: "953"
    id_type: "1"
    question_name: "Koga boste volili na prihajajocih predsedniskih volitvah?"
    randomize: "1"
    required: "1"
    scale_steps: "0"
    sequence_no: "1"
    title: "predsedniki"
    ▶ __proto__: Object
  ▶ 1: Object
  ▶ 2: Object
  ▶ 3: Object
  ▶ 4: Object
  ▶ 5: Object
  ▶ 6: Object
  length: 7

```

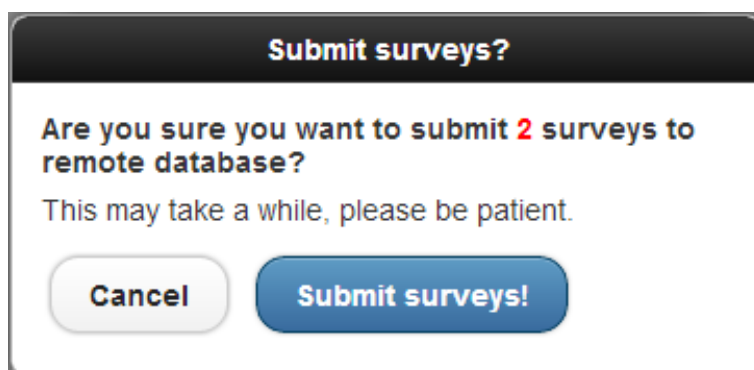
Slika 4.2: Prikaz podatkov vseh vprašanj, katere smo pridobili iz oddaljene podatkovne baze in jih zapišemo v podatkovno bazo.

Izbris projekta iz lokalne podatkovne baze je precej bolj preprost kot dodajanje novega projekta. Ob uporabnikovi zahtevi po izbrisu določenega projekta iz podatkovne baze s primernimi poizvedbami SQL odstranimo projekt, pripadajoča vprašanja in možne odgovore. Trenutna implementacija ob izbrisu določenega projekta ponovno zgradi celoten seznam vseh projektov, kar ni optimalno. Zaradi takih in podobnih pomankljivosti je mobilna aplikacija na starejših mobilnih napravah precej neodzivna.

### 4.3.2 Pošiljanje rešenih anket v oddaljeno bazo podatkov

Vse zbrane odgovore zapisujemo v lokalno podatkovno bazo in jih anketar ob priliki, ko ima dostop do spleta lahko pošlje v oddaljeno podatkovno bazo. V glavnem oknu ima možnost izbire pošiljanja vprašalnikov v oddaljeno podatkovno bazo. Uporabniku se prikaže dialog, kjer lahko vidi koliko vprašalnikov je na voljo za prenos do oddaljene podatkovne baze. Dialog je prikazan na sliki 4.3.

Ob kliku na pošiljanje rešenih vprašalnikov se sproži proces zapisovanja vseh vprašalnikov v oddaljeno podatkovno bazo. Najprej se pošljejo podatki o anketah v obliki JSON do spletne storitve. Ti osnovni podatki o anketi



Slika 4.3: Pošiljanje rešenih vprašalnikov iz lokalne v oddaljeno podatkovno bazo.

so kdo je izvajalec ankete oziroma anketar, ura in čas, kdaj je bila anketa izpolnjena in kateremu projektu oziroma vprašalniku pripada anketa. Če so osnovni podatki o anketah uspešno prenešeni, se začne prenos zbranih respondentovih odgovorov na vprašanja. Pri odgovorih se pošlje informacija kateri anketi in vprašalniku določen odgovor pripada, kakšen je odgovor in kakšno sekvenčno številko ima vprašanje, kateremu odgovor pripada. Če je celoten prenos vseh anket uspešen, počistimo vse vnose iz lokalne podatkovne baze, kjer so zbrane ankete in odgovori.

## 4.4 Nepovezani način (*Offline mode*)

To je način uporabe mobilne aplikacije, pri kateri ni potrebne internetne povezave. V tem načinu je možna izbira kateri vprašalnik želi uporabnik reševati in zbiranje odgovorov respondentov na vprašanja.

### 4.4.1 Izbira vprašalnika

Ob izbiri uporabnika za reševanje vprašalnikov, mora uporabnik najprej izbrati enega izmed vseh možnih vprašalnikov, ki so bili prenešeni in so shranjeni v lokalni podatkovni bazi. Na podlagi vseh vprašalnikov se zgradi



Slika 4.4: Zaslonska maska za izbiranje vprašalnika, ki ga uporabnik želi izpolnjevati.

spustni meni, kjer ima uporabnik možnost izbire vprašalnika. Ob kliku na katerega od vprašalnikov v spustnem meniju, se na zaslonu prikažejo tudi osnovni podatki o projektu, kot so kratek opis in ime snovalca vprašalnika.

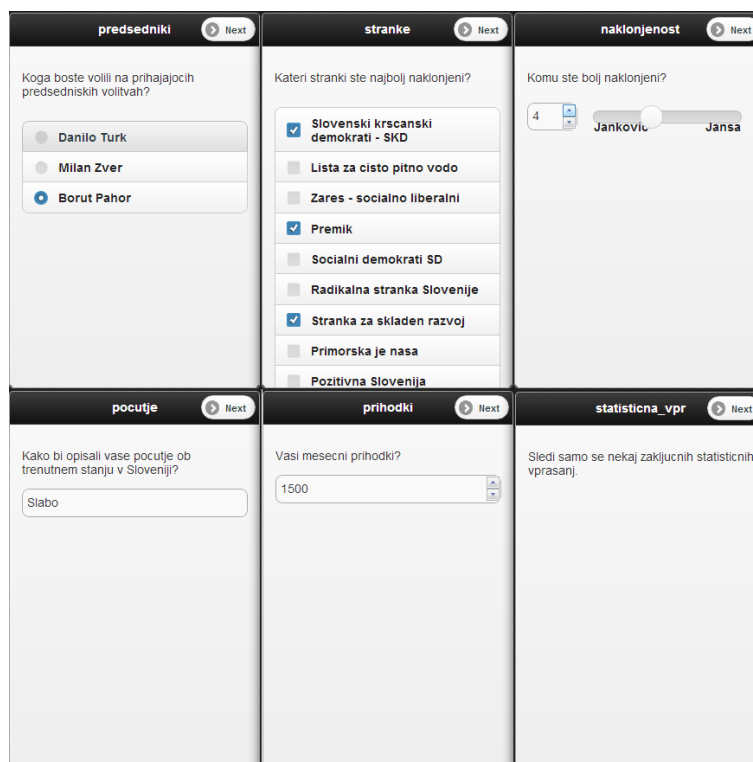
Ko je uporabnik izbral željen vprašalnik, se z gumbom na nadaljevanje začne izpolnjevanje odgovorov. Ker se odgovori zapisujejo v lokalno shrambo na tem mestu najprej ponastavimo vse vrednosti v lokalni shrambi in pobrišemo stare podatke. Razlog za ponastavitev lokalne shrambe je zaradi podatkov, ki bi lahko tam ostali, če katera od prejšnjih anket ni bilo v celoti izpolnjena.

#### 4.4.2 Izpolnjevanje vprašalnika

Za izpolnjevanje vprašalnika smo razvilil posebno stran, ki se dinamično zgradi na podlagi tipa vprašanja. Različne zaslonske maske za različne tipe vprašanj so vidne na sliki 4.5.

Za vsak tip vprašanja smo razvili posebno funkcijo, ki kot parameter prejme potrebne podatke za gradnjo. Ne glede na tip vprašanja je vedno na prvem mestu vprašanje, ki se zastavi respondentu. Za zaprte tipe vprašanj se podajo možni odgovori, v primeru skale robni vrednosti. Odprti tipi vprašanj teh možnih odgovorov nimajo. Poleg možnih odgovorov funkciji podamo še nujnost odgovora in naključno generacijo vrstnega reda možnih odgovorov. Vse potrebne podatke pridobimo iz lokalne podatkovne baze, ki se nahaja na mobilni napravi.

Ob izpolnjevanju vprašalnika si sproti beležimo trenutno sekvenčno številko vprašanja v lokalno shrambo. Sekvenčna številka pomeni zaporedno številko vprašanja v vprašalniku. Beležimo tudi številko vprašanja, ki sledi trenutnemu vprašanju. Privzeto je to številka naslednjega vprašanja, ki je po vrstnem redu za eno mesto oddaljeno za trenutnim. Ko uporabnik pritisne na gumb, ki ga vodi na naslednje vprašanje, preverimo kateri odgovor je izbran. Vsak odgovor ima v lokalni podatkovni bazi dodaten atribut, sekvenčno številko naslednjega vprašanja. Če je številka različna od številke v lokalni shrambi, slednjo posodobimo na novo vrednost.



Slika 4.5: Stran za izpolnjevanje vprašalnikov in prikaz vseh tipov vprašanj.

Ob kliku na gumb za naslednje vprašanje, se izbran odgovor zapiše v lokalno shrambo. Če je odgovorov na vprašanje več, se ti zapišejo v obliki tabele. Nato uporabnika preusmerimo zopet na isto spletno stran, kjer se dinamično generira neko drugo vprašanje. Sekvenčno številko naslednjega vprašanja preberemo iz lokalne shrambe. Ta informacija je dovolj, da lahko zopet zgradimo vse potrebno za naslednje vprašanje.

Ko pridemo do konca vprašalnika se anketarju ponudi možnost potrditve zbranih odgovorov. Če je zadovoljen z anketo se zbrani odgovori zapišejo v lokalno podatkovno bazo. Sledi samo še pošiljanje zbranih odgovorov v oddaljeno podatkovno bazo. Ta postopek je del povezanega načina in ga uporabnik lahko opravi na zahtevo, ko ima omogočeno povezavo s spletom.





## Poglavje 5

# Sklepne ugotovitve

Glavnina diplomskega dela je digitalizacija celotnega procesa terenskega anketiranja. V ta namen smo izdelali spletno in mobilno aplikacijo. Spletna aplikacija je namenjena sestavljalcu oziroma načrtovalcu vprašalnikov. Omogoča celovit pregled nad vprašalniki, grafično predstavitev in vpogled v zbrane odgovore in sestavljanje novih vprašalnikov ali urejanje že narejenih. Uporabniki mobilne aplikacije so anketarji, ki zbirajo odgovore respondentov na vprašanja, ki jih je sestavil načrtovalec. Razvili smo dva načina delovanja mobilne aplikacije, povezanega in nepovezanega. V povezanem načinu uporabnik lahko prenaša vprašalnike in pošilja zbrane odgovore do oddaljene podatkovne baze. V nepovezanem načinu anketar izpolnjuje vprašalnike. Uporaba spletne aplikacije je možna v vseh novejših spletnih brskalnikih, pogoj je podpora najnovejšega standarda HTML5. Mobilna aplikacija je podprta na večini modernih in pogosto uporabljenih mobilnih operacijskih sistemih, s pomočjo medplatformskega ogrodja PhoneGap. Zaradi velikega števila različnih mobilnih naprav, fizično testiranje na velikem številu mobilnih naprav ni bilo možno.

Uspešno smo realizirali in modernizirali celoten postopek terenskega anketiranja. Z digitalizacijo procesa smo občutno zmanjšali potreben čas za izdelavo vprašalnika, zbiranje odgovorov respondenta in analizo oziroma pregled nad zbranimi odgovori.

V celoti smo odstranili vse potrebe po uporabi papirja. Največja prednost je odstranitev faze, kjer je potrebno zbrane odgovore na papirju prenesti v strnjeno obliko. Za prenos odgovorov iz papirja v digitalno obliko je potreben človek. V razviti aplikaciji so odgovori zbrani v podatkovni bazi in primerni za več različnih oblik izvoza. Tako so na voljo za izdelavo zaključkov raziskave.

Sistem za terenskega anketarja je primeren za veliko podjetij, ki se ukvarjajo z raziskovanjem in analizo trga. Glede na visok porast pametnih mobilnih naprav, bi anketarji lahko mobilno aplikacijo naložili na njihovo lastno mobilno napravo. Tako podjetja ne bi imela stroškov z nabavo nove opreme.

Osnovne zahteve so izpolnjene, s spletno in mobilno aplikacijo smo omogočili izdelavo in izpolnjevanje vprašalnikov. Vseeno je prostora za napredek še dovolj. Pri razvoju strani za sestavljanje vprašalnika smo podprli osnovne tipe vprašanj, primanjkuje podpora bolj naprednim tipom vprašanj, kot je mrežni tip. Pregledovanje zbranih odgovorov je možno zgolj v tabeli in obliki krožnega grafikona. Zbrane podatke bi bilo možno predstaviti še v vrsto drugih oblikah. Mobilna aplikacija na starejših pametnih mobilnih napravah deluje prepočasi. Potrebna bi bila temeljita optimizacija celotne mobilne aplikacije. V primeru ponovne izdelave mobilne aplikacije se verjetno ne bi odločili za ponovno uporabo knjižnice jQuery Mobile. Lokalna podatkovna baza na mobilnih napravah je grajena na temeljih, ki so zastareli. Kljub temu, da standarda Web SQL verjetno ne bodo opustili tako kmalu, bi bilo primerno se na to pripraviti z univerzalno rešitvijo. Pri izpolnjevanju vprašalnika v mobilni aplikaciji ni možnosti vračanja na vprašanja, katera je anketar že izpolnil.

Nadzor anketarjev je boljši ker anketarji ne morejo ponarediti datuma in časa opravljanja ankete. Beležil bi se lahko lahko tudi čas reševanja ankete in na podlagi velikih odstopanj, bi lahko vodje enostavneje odkrivali goljufe. Poleg tega, bi za potrebe nadzora nad anketarji, lahko implementirali tudi zapisovanje geolokacijskih podatkov, kje je bila anketa opravljena.

Žal mobilne aplikacije nismo uspeli testirati na velikem številu mobil-

nih naprav in kako bi se aplikacija obnesla ob dejanskem reševanju anket na terenu. Potreben bi bil tudi stresni test (angl. *stress test*), kjer bi aplikacijo naenkrat uporabljalo zelo veliko anketarjev. Na tak način bi odkrili zmogljivosti sistema. Zaradi pomanjkanja testiranja obstaja možnost skritih pomankljivosti, ki jih sami nismo uspeli odkriti.



# Kazalo slik

2.1	Porazdelitev lokalne shrambe med različne brskalnike na isti napravi[3]. . . . .	6
2.2	PhpMyAdmin orodje za administracijo sistema za upravljanje s podatkovnimi bazami MySQL. . . . .	9
2.3	Grafičen prikaz spletne storitve in medsebojne interakcije med podatkovno bazo, odjemalcem in spletno storitvijo. . . . .	10
2.4	Okrnjena predstavitev drevesa DOM in njegovih objektov[6]. . . . .	12
2.5	Podprti mobilni operacijski sistemi s strani jQuery mobile knjižnice[11]. . . . .	13
2.6	Prikaz gradnikov HTML na mobilni aplikaciji in različne vizualne teme, ki so v sklopu knjižnice jQuery Mobile[12]. . . . .	14
2.7	Glavni elementi posnemovalnika Ripple in operacije, ki jih uporabnik lahko izvede. . . . .	17
2.8	Orodje Lita za administracijo podatkovnih baz SQLite. . . . .	18
3.1	Diagram primera uporabe spletne aplikacije. . . . .	21
3.2	Konceptualni model podatkovne baze spletne aplikacije. . . . .	22
3.3	Primer vprašalnika s tremi vprašanji. . . . .	23
3.4	Primer uporabniško dodanega vsebnika za vprašanje zaprtega tipa, kjer je na voljo samo en odgovor. . . . .	24
3.5	Orodna vrstica vsebnika. . . . .	25
3.6	Omogočanje stanja primi in spusti ob kliku na srednji gumb v orodni vrstici. . . . .	25

---

3.7	Primer vsebnika za vnos osnovnih podatkov o projektu. . . . .	26
3.8	Slikovni predogled in predstavitev tipov vprašanj in izbirno polje preko katerega lahko uporabnik dodaja vprašanja. . . . .	27
3.9	Numerično in znakovno vprašanje odprtega tipa. . . . .	28
3.10	Znakovno vprašanje odprtega tipa, kjer je na voljo eden ali več možnih odgovorov. . . . .	30
3.11	Vprašanje zaprtega tipa v obliki skale. . . . .	31
3.12	Kreiranje nove predloge za vprašanje zaprtega tipa z možnim enim odgovorom. . . . .	32
3.13	Primer uporabe predlog na vprašanju zaprtega tipa. . . . .	33
3.14	Prikaz metode buildHolder brez in z parametri. . . . .	34
3.15	Primer vrnjenih podatkov vprašalnikov od spletne storitve v obliki JSON. . . . .	35
3.16	Dodajanje in odstranjevanje vnosnih polj za možne odgovore znotraj vsebnika. . . . .	35
3.17	Prikaz celotne strukture vsebnika kot objekti DOM drevesa. . . . .	36
3.18	Pomnjenja stanja s pomočjo lokalne shrambe prikazano v spletnem razvijalskem okolju v brskalniku Chrome. . . . .	37
3.19	Sprememba toka vprašalnika glede na izbrane odgovore respondenta. . . . .	38
3.20	Prikaz maske za pregled vprašalnikov in njihovo manipulacijo s pomočjo vtičnika DataTables. . . . .	40
3.21	Uporaba vtičnika DataTables v spletni aplikaciji. . . . .	41
3.22	Prikaz grafa ustvarjenega z orodjem Google Chart. . . . .	42
4.1	Seznam projektov in možnosti prenosa in izbrisa iz lokalne podatkovne baze. . . . .	46
4.2	Prikaz podatkov vseh vprašanj, katere smo pridobili iz oddaljene podatkovne baze in jih zapišemo v podatkovno bazo. . . . .	48
4.3	Pošiljanje rešenih vprašalnikov iz lokalne v oddaljeno podatkovno bazo. . . . .	49

---

4.4	Zaslonska maska za izbiranje vprašalnika, ki ga uporabnik želi izpolnjevati. . . . .	50
4.5	Stran za izpolnjevanje vprašalnikov in prikaz vseh tipov vprašanj.	52





# Literatura

- [1] (2012) HTML Wikipedija. Dostopno na:  
<http://sl.wikipedia.org/wiki/HTML>.
- [2] (2012) HTML5 Web Storage. Dostopno na:  
[http://www.w3schools.com/html/html5\\_webstorage.asp](http://www.w3schools.com/html/html5_webstorage.asp).
- [3] (2012) How Local Storage Is Shared by the Browser. Dostopno na:  
<https://developers.google.com/web-toolkit/doc/latest/DevGuideHtml5Storage>.
- [4] (2012) Uradna spletna stran XAMPP. Dostopno na:  
<http://www.apachefriends.org/en/xampp.html>.
- [5] (2012) CSS Wikipedija. Dostopno na:  
<http://sl.wikipedia.org/wiki/CSS..>
- [6] (2012) The Document Object Model(DOM). Dostopno na:  
<http://dsmith77.wordpress.com/2008/07/20/the-document-object-model-dom/>.
- [7] (2012) MySQL Wikipedija. Dostopno na:  
<http://sl.wikipedia.org/wiki/MySQL>.
- [8] (2012) jQuery Wikipedija. Dostopno na:  
<http://en.wikipedia.org/wiki/JQuery>.
- [9] (2012) jQuery - Mobilna platforma. Dostopno na:  
<http://www.pomagalnik.com/tag/jquery-mobile/>.

- 
- [10] (2012) jQuery mobile galerija podjetij. Dostopno na:  
<http://www.jqmgallery.com/>.
- [11] (2012) jQuery mobile spletna stran. Dostopno na:  
<http://jquerymobile.com/>.
- [12] (2012) jQuery mobile prikaz tem. Dostopno na:  
<http://media02.hongkiat.com/jquery-mobile-custom-themes/jquery-mobile-custom-themes.jpg>.
- [13] (2012) PhoneGap explained visually. Dostopno na:  
<http://phonegap.com/2012/05/02/phonegap-explained-visually/>.
- [14] (2012) Web SQL Database. Dostopno na:  
<http://dev.w3.org/html5/webdatabase/>.
- [15] (2012) Indexed Database API Wikipedia. Dostopno na:  
[http://en.wikipedia.org/wiki/Indexed\\_Database\\_API](http://en.wikipedia.org/wiki/Indexed_Database_API).
- [16] Lita - SQLite Administration Tool. Dostopno na:  
<http://www.dehats.com/drupal/?q=node/58>.
- [17] (2012) Mag. Damjan Vavpotič, Podatkovno modeliranje s PowerDesigner. Dostopno na:  
[http://baze.fri.uni-lj.si/dokumenti/Podatkovno\\_modeliranje.pdf](http://baze.fri.uni-lj.si/dokumenti/Podatkovno_modeliranje.pdf).
- [18] (2012) DataTables. Dostopno na:  
[www.http://datatables.net/](http://datatables.net/).
- [19] (2012) Google Chart Tools. Dostopno na:  
<https://developers.google.com/chart/>.