

Mladi za napredek Maribora 2017  
34. srečanje

## MOBILNA APLIKACIJA BUBBLE

Raziskovalna naloga: Računalništvo

Avtor: ROK VIRANT, KRISTIAN GRGIĆ, MAJA KITEK  
Mentor: MANJA SOVIČ POTISK, JOŽE ŠTRUCL  
Šola: SREDNJA ELEKTRO-RAČUNALNIŠKA ŠOLA MARIBOR

Maribor, januar 2017

## Kazalo vsebine

Kazalo slik .....	3
Kazalo grafikonov .....	3
Zahvala .....	4
1. POVZETEK .....	5
SUMMARY .....	5
2. UVOD .....	6
3. FUNKCIONALNOST APLIKACIJE .....	7
3.1 Nalaganje in prenos datotek na strežnik .....	7
3.2 Kreiranje podatkovne baze .....	9
3.3 Grafične vmesniške komponente mobilne aplikacije .....	10
4. PLATFORMA ANDROID .....	11
4.1 Kaj je Android? .....	11
4.2 Open Handset Alliance .....	11
4.3 Prednosti platforme Android .....	12
4.4 Zgodovina .....	13
4.5 Arhitektura .....	14
4.6 Stanja aktivnosti in prehodi med njimi .....	15
5. ANDROID STUDIO .....	17
5.1 Struktura projekta .....	17
5.2 Uporabniški vmesnik .....	19
5.3 Android SDK .....	20
5.4 Emulator – Navidezna mobilna naprava .....	20
6. ADOBE ILLUSTRATOR .....	22
6.1. Uvod v program .....	22
6.2. Opis in zgodovina programa .....	22
6.3. Naše delo s programom .....	23
7. APLIKACIJA BUBBLE .....	25
7.1. Zagon aplikacije .....	25
7.2. Glavna aktivnost .....	26
7.3. Ostale aktivnosti .....	27
7.1 ANKETA .....	31
8.1. Starost anketirancev .....	31
8.2. Kakšni mobilni operacijski sistem uporabljate? .....	32

8.3. Ali poznate Dijaski.net?	32
8.4. Koliko uporabljate Dijaski.net?	33
8.5. Bi vam koristil Dijaski.net v obliki mobilne aplikacije?	33
7.2 ZAKLJUČEK	34
7.3 DRUŽBENA ODGOVORNOST	35
VIRI	36

## Kazalo slik

Slika 1: Delovanje lokalnega spletnega strežnika	7
Slika 2: Različna področja na strežniku	8
Slika 3: Prikaz dokumentov PHP na strežniku	9
Slika 4: Primer podatkovne baze	10
Slika 5: Platforma Android	11
Slika 6: Open Handset Alliance	12
Slika 7: Različne verzije API za Android	13
Slika 8: Zgradba operacijskega sistema Android	15
Slika 9: Življenjski cikel aktivnosti platforme Android	16
Slika 10: Android Studio logotip	17
Slika 11: Prikaz projekta v Android Studiu	18
Slika 12: Izbira pogleda "Problems"	19
Slika 13: Glavno okno programa Android Studio	19
Slika 14: Prikaz okna za izbiro emulatorja oz. telefona	21
Slika 15: Galaxy Nexus Emulator	21
Slika 16: Prva verzija logotipa Bubble	23
Slika 17: Druga verzija logotipa Bubble	23
Slika 18: Tretja verzija logotipa Bubble	24
Slika 19: Četrta verzija logotipa Bubble	24
Slika 20: Peta verzija logotipa Bubble	24
Slika 21: Zadnja verzija logotipa Bubble	24
Slika 22: Aktivnost ob zagonu aplikacije	25
Slika 23: Glavna aktivnost	26
Slika 24: Prva aktivnost za izbrano področje	27
Slika 25: Notranji pomnilnik telefona	28
Slika 26: Aktivnost slo2 brez prikaza vsebine na strežniku	29
Slika 27: Aktivnost slo2 s prikazom vsebine na strežniku	29

## Kazalo grafikonov

Grafikon 1: Relativno število naprav, ki uporabljajo določeno platformo za Android	14
Grafikon 2: Starost anketirancev	31
Grafikon 3: Uporabnost mobilnih operacijskih sistemov	32
Grafikon 4: Poznavanje strani Dijaski.net	32
Grafikon 5: Uporaba spletne strani Dijaski.net	33
Grafikon 6: Dijaski.net kot mobilna aplikacija	33

## **Zahvala**

Zahvaljujemo se našima mentorjema za vso pomoč, spodbudo in nasvete pri pisanju raziskovalne naloge. Prav tako se zahvaljujemo sošolcem, ki so nam pomagali izboljšati aplikacijo z vidika uporabnikov.

## 1. POVZETEK

Idejo in navdih za raziskovalno nalogo smo dobili med uporabljanjem spletne strani Dijaski.net. Zanimalo nas je, kako bi ta sistem lahko uveljavili kot mobilno aplikacijo z nekaj izboljšavami. Naša ideja je, da bi se preko mobilne aplikacije povezali študentje ali dijaki in si med seboj pomagali. Ustvarili bi neke vrste spletno učilnico, do katere ima vstop vsak registriran uporabnik. V spletni učilnici bi lahko dijaki dostopali do raznih poučnih vsebin. Med temi vsebinami bi našli učno snov za določen predmet, prav tako bi našli različne vaje, teste ... Uporabniki bi te datoteke lahko naložili na telefon in na server, da bi do njih imeli dostop še drugi. Aplikacijo ustvarjamo v programskem jeziku Java, saj želimo, da je na voljo uporabnikom operacijskega sistema Android. Pri celotnem poteku dela smo najprej ustvarili vizualni del aplikacije, nato pa uredili še kodo, ki omogoča delovanje in funkcionalnost vseh gumbov v aplikaciji. Nato smo ustvarili še server, na katerega je možno nalagati datoteke.

## SUMMARY

We got the idea for this research assignment while we were using the web page called Dijaski.net. We were wondering, how could we make it into a mobile application with some improvements. Our idea is, that students could connect over the application and help each other study. We would create an online classroom, every registered user would have access to it. Students will be able to access educational content. Among these contents they would be easily able to find learning material, exercises, old exams and tests. The users could download the files on their phones and also upload them on the server, so others can access them as well. We are making the application in a programming language called Java, we want the application to be accessible over mobile phones, which use the Android operating system. First, we created the visual part of the application, then we edited the code, which enables the application to work and function. We also created a server, to which we can upload or download files.

## 2. UVOD

Namen aplikacije je širjenje znanja s pomočjo mobilnih telefonov. Naši cilji so, da med seboj povežemo dijake in študente, da si lahko pomagajo. Z našo aplikacijo želimo omogočiti študentom in dijakom dostop do raznih zapiskov, nalog in vaj, testov in maturitetnih pol. Prav tako želimo med njimi vzpostaviti komunikacijo preko aplikacije. Predstavljajte si spletno učilnico, v kateri se lahko uči več ljudi hkrati. Problem, ki smo ga uvideli, je, da nekateri ljudje premalo cenijo izobrazbo, morda bi jim bilo lažje, če bi imeli podatke dobesedno pri roki. Veliko dijakov, študentov in tudi profesorjev že uporablja Dijaski.net. Na spletni strani je naloženih veliko datotek, vendar so redke, ki imajo pravilne podatke. Na področju slovenščine se nahaja veliko obnov za posamezna literarna dela, vendar je vsaka drugačna. Naša aplikacija bi omogočala ocenjevanje naložene datoteke, prav tako tudi podana mnenja ali popravke. Naša aplikacija bo torej omogočala:

- nalaganje datotek na strežnik ali telefon,
- komunikacijo med uporabniki,
- ocenjevanje in komentiranje naloženih datotek.

### 3. FUNKCIONALNOST APLIKACIJE

Glavni cilj aplikacije je, da bi se vsi ljudje povezali in si pomagali, se učili ter ustvarjali skupaj. Trenutno je izdelana in delujoča aplikacija v prvi verziji, katera vsebuje nalaganje in prenos dokumentov preko lokalnega strežnika na telefon. Aplikaciji nameravamo v roku enega meseca dodati še gumb za iskanje in pogovorna okna, kjer lahko uporabnik pride v stik z ostalimi uporabniki te aplikacije.

Dolgotrajen cilj je, ustvariti t. i. učilnice, kjer bi lahko določena oseba z večjim znanjem in boljšimi ocenami svojega profila predavala učencem, dijakom in vsem ljudem, ki se želijo naučiti česa s svojega področja. Dodali bi tudi točkovanje, kar pomeni, da bi se npr. uporabniku izpisali najboljši predavatelji oz. učitelji na določenem področju. Na njegovem profilu bi si lahko uporabnik ogledal njegove izkušnje, cilje, interese in na podlagi tega ugotovil, ali bi se udeležil njegovega spletnega predavanja oz. učenja. Uporabnik bi se lahko seveda udeležil skupinskega učenja z ostalimi učenci, ki želijo doseči nekaj skupnega. S tem bi spodbujali sodelovanje in pomoč med ljudmi oz. dijaki.

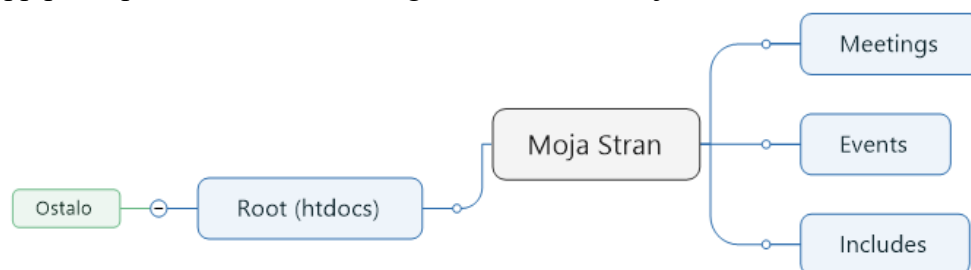
Potek trenutne aplikacije deluje tako, da do nje dostopamo preko glavnega menija v telefonu. Uporabnik se s svojim ustvarjenim računom prijavi v bazo in tako dostopa do aplikacije. Na voljo ima različna področja, kjer lahko pri vsakem naloži (na strežnik) ali pa poljubno prenese (iz strežnika) datoteko oz. dokument.

#### 3.1 Nalaganje in prenos datotek na strežnik

Pri poskusnem delovanju smo se odločili za uporabo paketa LAMP. LAMP označuje skupek odprtokodne programske opreme, ki tvori popolnoma delujoč spletni strežnik, ki je sposoben gostiti dinamične spletne strani. Paket sestavljajo:

- **Linux** kot operacijski sistem,
- **Apache** kot spletni strežnik,
- **MySQL** kot strežnik podatkovne baze in
- **Perl, PHP** oz. **Python** kot skriptni jezik.

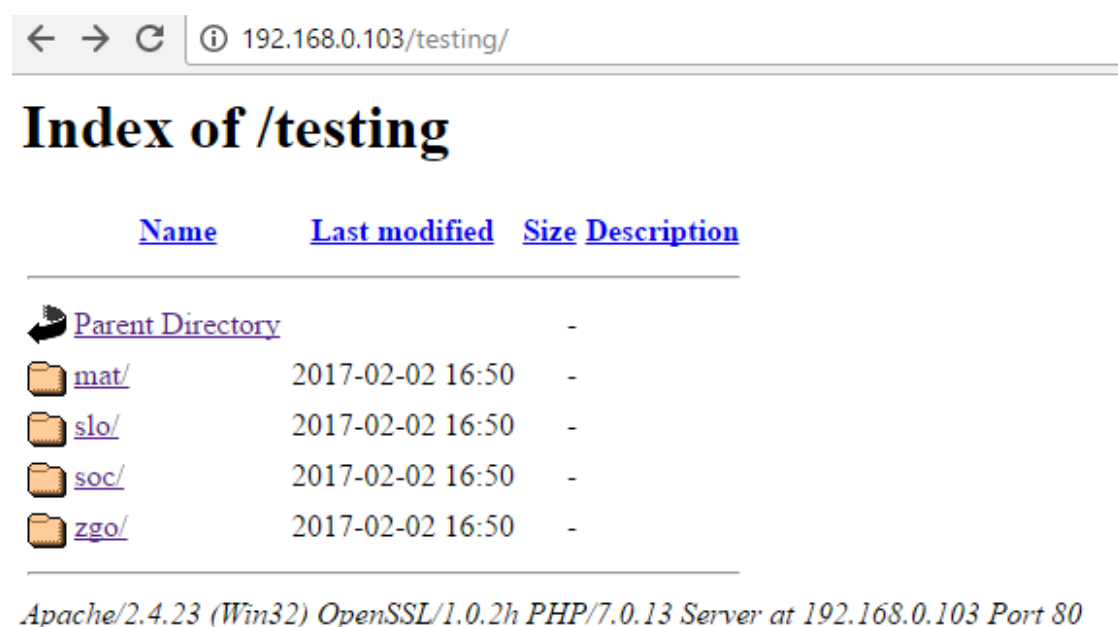
LAMP spada pod XAMPP in je sestavljen tako, da je namestitev spletnega strežnika čim hitrejša namesto zamudnega zaporedja ukazov. Za pravilno delovanje je potrebno Xampp paket prenesti z interneta in ga inštalirati na svojem računalniku.



Slika 1: Delovanje lokalnega spletnega strežnika

Kakor je prikazano na zgornji sliki, se mapa htdocs nahaja v glavni mapi xampp, ki se ustvari na lokalnem disku po inštalaciji paketa. V mapo htdocs lahko dodajamo poljubne mape in na tak način ustvarimo lokalni strežnik, do katerega dostopamo z IP naslovom svojega računalnika in imena svoje mape. V iskalno vrstico vpišemo (*svoj IP naslov ali localhost*)/(*ime ustvarjene mape*). V ustvarjeno mapo lahko dodajamo različne .html dokumente in na tak način ustvarimo svojo spletno stran, do katere dostopamo preko lokalnega strežnika.

Apache spada v paket LAMP in predstavlja spletni strežnik. Za pravilno delovanje Apache strežnika moramo imeti delujočo internetno povezavo in odprta porta 80 in 443, ki omogočata pravičen prenos datotek na strežnik. Za pravilno delovanje aplikacije smo morali tudi napisati kodo v obliki programskega jezika Java. V programu smo dodali URL, preko katerega lahko dostopamo do samega strežnika. Na strežnik smo dodali 4 mape, ki predstavljajo 4 trenutna področja v prvi verziji aplikacije. Te 4 mape so: Matematika, Slovenščina, Sociologija in Zgodovina.




Slika 2: Različna področja na strežniku

V vsaki od teh map se nahajata dva PHP dokumenta in mapa (ime področja)-dat, kamor se shranjujejo vsi dokumenti in podatki, od kod lahko prenašamo datoteke. Da smo se lahko povezali in vzpostavili delovanje med aplikacijo in strežnikom, smo na koncu morali še dodati dva PHP dokumenta, ki dovoljujeta nalaganje in prenos datotek na strežnik. V primeru, da proces ne deluje pravilno, nam javita, kaj je narobe. Za nalaganje datotek se uporablja .php dokument z imenom save\_file, medtem ko se za prenos datotek uporablja .php dokument z imenom script. Oba dokumenta se nahajata v vsaki izmed štirih map.



## Index of /testing/mat

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 <a href="#">Parent Directory</a>		-	
 <a href="#">matdat/</a>	2017-02-02 17:43	-	
 <a href="#">save_file.php</a>	2017-02-02 16:51	235	
 <a href="#">script.php</a>	2017-02-02 17:40	239	

Apache/2.4.23 (Win32) OpenSSL/1.0.2h PHP/7.0.13 Server at 192.168.0.103 Port 80

*Slika 3: Prikaz dokumentov PHP na strežniku*

Za izklop in vklop strežnika uporabljamo Xampp kontrolno ploščo v obliki programa, kjer vključimo Apache in strežnik prične pravilno delovati. Moramo vedeti, da je ta strežnik lokalni, kar pomeni, da omogoča nalaganje in prenos datotek samo v bližini lokalnega omrežja. Na daljavo to ni mogoče, zato si prizadevamo, da bi kupili svoj lasten strežnik in s tem omogočili vsem uporabnikom delitev datotek.

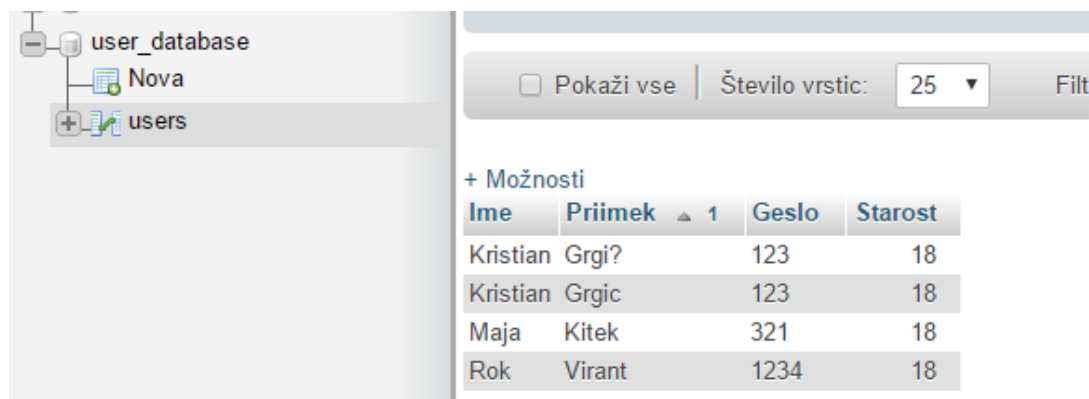
### 3.2 Kreiranje podatkovne baze

V aplikaciji smo uporabnikom omogočili ustvaritev svojega uporabniškega računa. Uporabnik za registracijo vnese svoje ime, priimek, geslo in starost. Podatki se vpišejo v ročno ustvarjeno podatkovno bazo z imenom »user\_data«, ki vsebuje tabelo »users«. V to tabelo se shranjujejo vsi ustvarjeni uporabniki. Ko je uporabnik dodan, se lahko v bazo kadarkoli vpiše s svojim uporabniškim računom, če ima omogočeno internetno povezavo. Za kreiranje podatkovne baze smo uporabili MySQL – strežnik podatkovne baze, ki je vključen v paket LAMP. Pod ta strežnik spada orodje za ustvarjanje podatkovnih baz z imenom phpMyAdmin, kjer se tudi nahaja naša podatkovna baza.

Do orodja phpMyAdmin dostopamo podobno kot do svojega strežnika oz. spletne strani. Najprej vpišemo svoj IP naslov računalnika in takoj zatem phpMyAdmin (*svoj IP naslov ali localhost*)/phpMyAdmin). Po pravilnem delovanju se nam prikaže domača stran orodja, kjer lahko takoj kreiramo podatkovno bazo in tabele.

Tabelo lahko poljubno spreminjamo, po želji odstranimo ali dodamo uporabnike in spreminjamo njihove lastnosti, kar je znotraj baze dosti lažje. Takšno orodje lahko uporabljamo za male in velike baze in če želimo bazo po potrebi prenesti, ni potrebno znova vpisovati podatke, ampak prenesemo bazo v obliki .sql dokumenta. Baze lahko prostovoljno tudi preurejamo s SQL programskim jezikom. Tabele lahko tudi združujemo v relacije. Ko jih med seboj povežemo, se določeni tipi uporabijo tudi v drugi tabeli. Ta način je primerno uporabiti, kadar ne želimo na novo vpisovati

podatkov naših uporabnikov, temveč jih samo prenesemo iz ene tabele v drugo s pomočjo SQL jezika



Ime	Priimek	Geslo	Starost
Kristian	Grgi?	123	18
Kristian	Grgic	123	18
Maja	Kitek	321	18
Rok	Virant	1234	18

Slika 4: Primer podatkovne baze

Podatkovni tipi imena, priimeka in gesla so VARCHAR, podatkovni tip starosti pa INT. V bazo lahko po potrebi shranimo ogromno število uporabnikov in njihovih podatkov. V primeru, da uporabnik pozabi svoje geslo ali pa, da mu pošljemo pomembne informacije, potrebujemo uporabnikov e-poštni naslov. V prihodnjem mesecu oz. kadar bo aplikacija na voljo vsem uporabnikom, bo ta funkcija dodana.

### 3.3 Grafične vmesniške komponente mobilne aplikacije

Uporabnik se sreča z različnimi vmesniškimi komponentami kot so prijavi sistem uporabnika, gumb nastavitve, gumbi za izbiro področja, prikaz datotek na serverju, nalaganje in prenos datotek.

Prijavi sistem je prikazan na prvi aktivnosti oz. prvi strani, ki se prikaže uporabniku in je namenjena, da si uporabnik ustvari novi uporabniški račun. Na začetku se mu prikažeta dve tekstovni vrstici, v kateri mora uporabnik vpisati svoje že ustvarjeno uporabniško ime in geslo. V primeru, da uporabnik še nima lastnega računa, lahko pritisne na gumb »Registracija«, kjer si lahko ustvari novi uporabniški račun. Navesti mora svoje ime, poljubno uporabniško ime in geslo ter svojo starost. V prihodnje bo potrebno tudi navesti e-poštni naslov.

V aplikacijo je vključen gumb za nastavitve, kjer lahko uporabnik poljubno spreminja svoje podatke na računu. Spremeni lahko svoje uporabniško ime in tudi geslo. Gumb za nastavitve je t.i. »Image Button«, kar pomeni, da vsebuje poljubno sliko namesto štirikotne oblike gumba.

Gumbi za izbiro področja so navadni štirikotni gumbi, ki omogočajo uporabniku, da izbere področje, na katero želi naložiti ali prenesti datoteke iz strežnika.

Gumba za naložitev in prenos se prikažeta takoj po izbiri področja. Če pritisnemo prvi gumb, se prikaže notranji pomnilnik našega telefona, kjer lahko izberemo sliko, video ali dokument, ki ga želimo naložiti na strežnik. Pri izbiri drugega gumba se prikaže gumb za prikaz datotek na serverju.

Gumb za prikaz datotek na serverju uporabniku poda informacije, katere datoteke se nahajajo na samem serverju na določenem področju. Uporabnik lahko poljubno izbere datoteko, ki se mu v ozadju prenese na telefon. Datoteka se po prenosu nahaja pod mapo prenosi oz. »Downloads« na uporabnikovem mobilnem telefonu.

## 4. PLATFORMA ANDROID

Za platformo Android smo se odločili zato, ker je leta 2016 pokrivala 81.9 % vseh uporabnikov.

### 4.1 Kaj je Android?

Android je programska platforma in operacijski sistem za pametne mobilne telefone, ki temelji na Linuxovem jedru. Razvija ga Google v sodelovanju s podjetji združenja Open Handset Alliance (OHA). Android je v prvi vrsti namenjen mobilnim telefonom, vendar ga najdemo tudi na drugih napravah kot so tablični računalniki, ročne ure in podobno. Ime platforme oziroma operacijskega sistema izhaja iz angleške besede »android«, kar pomeni robot, ki je podoben človeku in se obnaša kot človek.



*Slika 5: Platforma Android*

### 4.2 Open Handset Alliance

Open Handset Alliance (OHA) je ime za poslovno združenje številnih podjetji, ki ga je Google ustanovil zaradi prizadevanja za skupni razvoj odprtih standardov na področju prenosnih naprav, k čemur teži Android. Združenje OHA teži k pospešenemu razvoju inovacij na področju mobilnih storitev ter se zavzema za bogatejše, cenejše in boljše mobilno doživetje vsakega posameznika. Združenje je bilo javnosti uradno predstavljeno 5. novembra 2007, ko je bil uradno napovedan tudi Android. Združenje danes šteje 47 pomembnih podjetji s področja tehnologije programske in strojne opreme ter telekomunikacij.



## open handset alliance

*Slika 6: Open Handset Alliance*

### 4.3 Prednosti platforme Android

Android ima kar nekaj pomembnih prednosti pred ostalimi platformami za mobilne telefone. Nekatere od teh prednosti so:

**1. Android je odprtokoden in povsem brezplačen.**

Za potrošnike to pomeni, da jim bodo na voljo cenejše in funkcionalnejše mobilne naprave ter večje število inovativnih storitev. Proizvajalcem mobilnih telefonov Android omogoča znižanje stroškov razvoja programske opreme ter hitrejšo realizacijo njihovih konceptov, prav tako jim ob fleksibilnosti sistema še vedno prinaša možnost krojenja lastne identitete z razvojem samosvojih rešitev

**2. Android omogoča razvijalcem več možnosti pri razvoju programske opreme.**

Android omogoča, da dodatne aplikacije koristijo vse strojne zmožnosti naprav; to so za primer na dotik občutljiv zaslon, senzor pospeška, kompas, GPS, kamera, mikrofoni, strojni grafični pospeševalnik itd. Dodatne aplikacije lahko spreminjajo ali zamenjujejo osnovne aplikacije in komponente uporabniškega vmesnika

**3. Android se ponaša z najboljšim uporabniškim vmesnikom.**

Ta uporabniški vmesnik je zelo hitro odziven, enostaven in pregleden. Omogoča intuitivno upravljanje prek zaslona s prstom brez uporabe pisala oziroma »stajlusa«. Android podpira večopravnost in omogoča, da aplikacije tečejo v ozadju.

**4. Naprave z operacijskim sistemom Android se samodejno sinhronizirajo z Googlovimi storitvami.**

Na ta način se podatki zavarujejo pred izgubo v primeru izgube mobilnega telefona, hkrati pa to tudi pomeni, da so vsi naši osebni podatki ves čas ažurni, ne glede na to, ali do njih dostopamo z osebnega računalnika ali mobilnega telefona.

**5. Uporabnik lahko hitro, enostavno in brez pomoči osebnega računalnika nalaga aplikacije.**

To omogoča storitev Google Play, preko katere je mogoče neposredno z mobilne naprave brskati po seznamih aplikacij in iger, jih nalagati na mobilni telefon in jih ocenjevati.

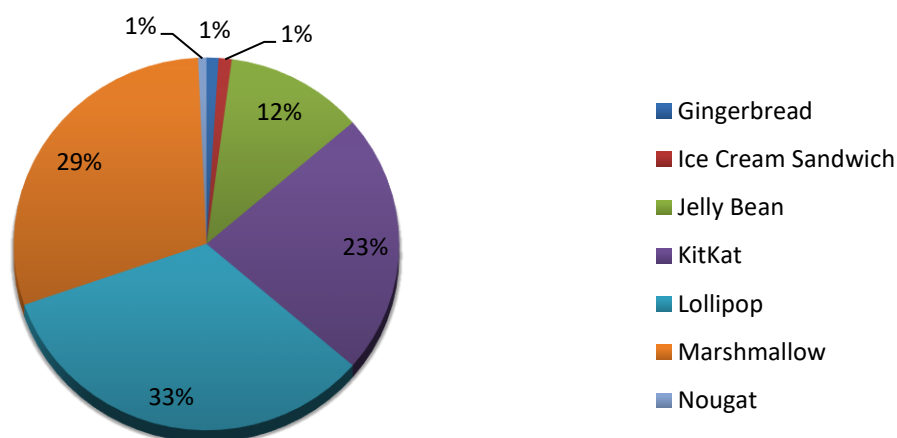
## 6. Operacijski sistem Android omogoča enostavno posodobitev v primeru, da je na spletu dostopna novejša različica programske opreme.

### 4.4 Zgodovina

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.1%
4.1.x	Jelly Bean	16	4.0%
4.2.x		17	5.9%
4.3		18	1.7%
4.4	KitKat	19	22.6%
5.0	Lollipop	21	10.1%
5.1		22	23.3%
6.0	Marshmallow	23	29.6%
7.0	Nougat	24	0.5%
7.1		25	0.2%

*Slika 7: Različne verzije API za Android*

## Relativno število naprav, ki uporabljajo Android za določeno platformo



Grafikon 1: Relativno število naprav, ki uporabljajo določeno platformo za Android

### 4.5 Arhitektura

Operacijski sistem Android je sestavljen iz petih elementov. Ti elementi so: aplikacije, njihovo ogrodje, knjižnice, prevajalnik, in Linux kernel.

#### Aplikacije

Vse aplikacije so napisane v programskem jeziku Java, a se za izdelavo programov uporablja tudi Xml. Aplikacije so sestavljene v Android paket s končnico ".apk."

Vsaka aplikacija se požene v svojem Linux procesu. Operacijski sistem požene proces takrat, ko mu je poslana zahteva za izvajanje aplikacije. Ko z izvajanjem aplikacije zaključimo, se proces zapre. To omogoča rabo pomnilnika tudi drugim aplikacijam. Vsak posamezni proces se prevede posebej, kar omogoči izoliranje aplikacij, da delujejo med sabo neodvisno. Vsaki aplikaciji se ob zagonu ustvari tudi lastna identifikacijska koda, kateri se nato dodajo pravice za uporabo strojne opreme.

#### Aplikacijsko ogrodje

V aplikacijskem ogrodju se nahajajo vse systemske aplikacije, ki se uporabljajo za koordiniranje aplikacij. Te aplikacije so:

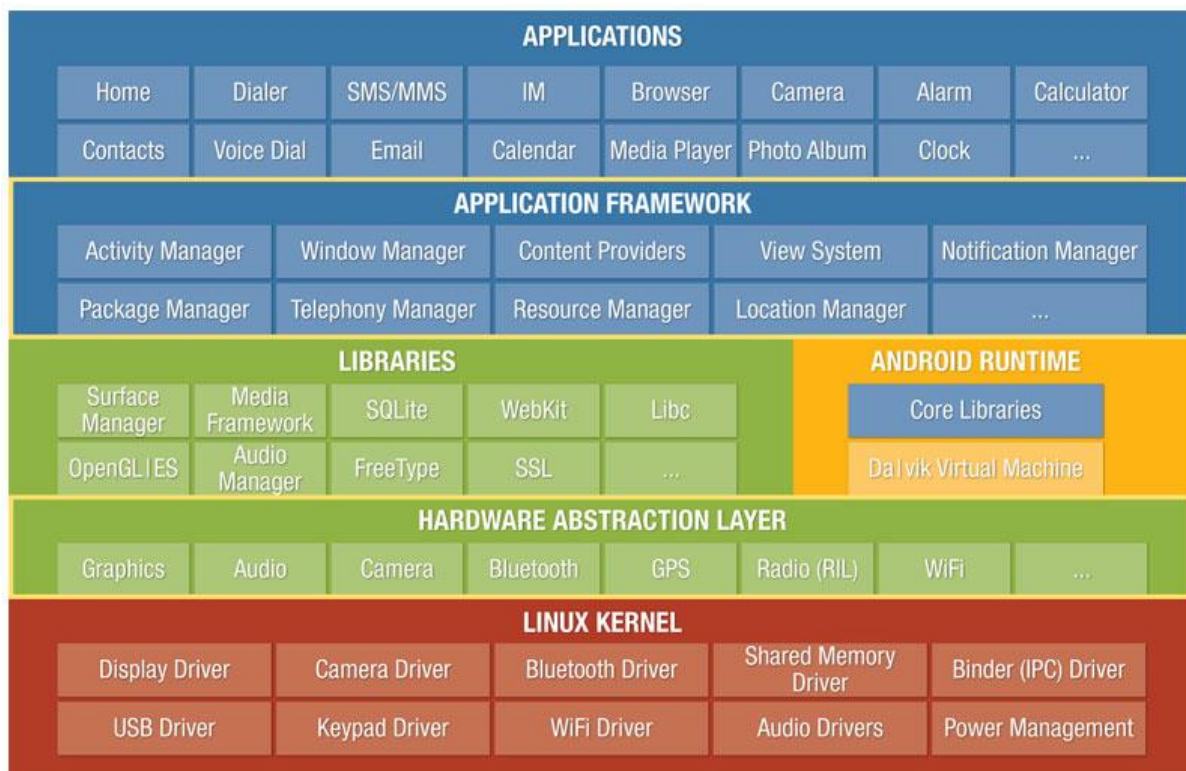
- upravljevec aktivnosti,
- upravljevec pomnilnika,
- upravljevec lokacij,
- upravljevec obvestil,

## Knjižnice

So temeljni del sistema Android. Do njih navaden uporabnik operacijskega sistema ne more dostopati. Uporabljajo jih razvijalci za dostop do strojnih komponent naprave.

## Prevajalnik

Operacijski sistem Android za prevajanje kod aplikacij do verzije 4.4 uporablja prevajalnik JIT (Just in time compiler). To omogoča prenos aplikacij na več različnih prenosnih naprav brez ponovnega pisanja izvirne kode. V verziji 4.4 je uvedena možnost vklopa prevajalnika ART (Android Run Time), ki vse aplikacije prevede ob zagonu (AOT).



Slika 8: Zgradba operacijskega sistema Android

## 4.6 Stanja aktivnosti in prehodi med njimi

Aktivnost je v vsakem trenutku v enem izmed štirih možnih stanj:

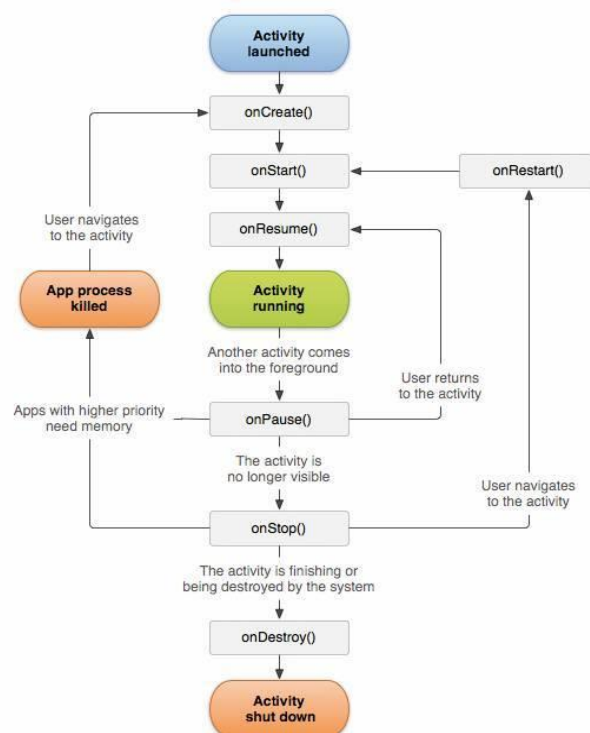
- aktivna (active) - aktivnost je prikazana na zaslonu v ospredju in se odziva na uporabnikove akcije (dotike zaslona, pritiske tipk ...);
- prekinjena (paused) - aktivnost se ne odziva na uporabnikove akcije, vendar je še vedno vsaj delno vidna na zaslonu (izbira opcij, potrjevanje dialogov ...);
- ustavljena (stopped) - aktivnost na zaslonu ni več vidna, a je še vedno na zgodovinskem skladu;
- uničena (destroyed) - aktivnosti ni na zaslonu in je tudi ni več na zgodovinskem skladu.

Androidna platforma omogoča zaznavanje prehoda med posameznimi stanji aktivnosti s pomočjo posebnih metod razreda Activity, ki jih sistem ob prehodih med stanji avtomatsko kliče. Te metode so:

- ***void onCreate(Bundle savedInstanceState),***
- ***void onStart(),***
- ***void onRestart(),***
- ***void onResume(),***
- ***void onPause(),***
- ***void onStop(),***
- ***void onDestroy().***

Prazne implementacije teh metod so že del razreda Activity, obvezno je implementirati le metodo onCreate, ki se kliče ob samem proženju aktivnosti. Če programer ne določi drugače, so prehodi med posameznimi stanji aktivnosti pri izvajanju aplikacije "nevidni", ob redefiniciji določenih metod pa lahko te prehode naredimo tudi vidne z izvedbo določenih akcij (sprememba vsebine). Z možnostjo redefinicije metod se imamo možnost odzvati na spremembe ob določenih dogodkih. Tako lahko ob koncu aktivnosti trajno shranimo uporabnikov vnos.

Kako so med seboj povezana posamezna stanja, v katerih je lahko aktivnost, je prikazano na spodnji sliki, ki vsebuje t. i. življenjski cikel aktivnosti.



Slika 9: Življenjski cikel aktivnosti platforme Android



## 5. ANDROID STUDIO

Za Android Studio smo se odločili zato, ker je relativno preprost program za razumevanje, je zastonj in je eden od najpopularnejših programov za razvijanje aplikacij. Napisan je v jeziku Java, namenjen je prav vsem uporabnikom, in sicer od mlajših do starejših. Priporočeno je vsaj osnovno znanje programiranja spletnega oblikovanja, sicer pa učenje sproti tudi ne povzroča nobenih problemov. Večkrat smo se tudi sami znašli v situaciji, kjer smo potrebovali pomoč in smo rešitev vedno našli na internetu. Pomoč vam lahko tudi brez problema posredujejo razvijalci programa, ki so vedno na voljo.

Sam program nam ni povzročal večjih težav, razen tega, da je včasih tako rekoč »zamrznil«. Ugotovili smo, da je delovanje predvsem odvisno od računalnika in njegovih specifičnih zmogljivosti. Delovanje in hitrost emulatorja (virtualno delovanje telefona, ki je prikazano na računalniku) v programu je odvisno od delovanja računalnika. Če je uporabnikov računalnik dokaj počasen, tudi delovanje emulatorja ne bo nič kaj boljše. Uporabnik ima tudi možnost povezati telefon na računalnik. To se nam zdi boljša možnost, ki jo priporočamo uporabnikom s starejšim računalnikom. Delovanje in graditev aplikacije preko lastnega telefona je precej hitrejše v primerjavi z emulatorjem.

V nadaljevanju bomo ostali pri tem programu, saj ponuja ogromno možnosti, ki jih lahko vključimo v našo aplikacijo.



*Slika 10: Android Studio logotip*

### 5.1 Struktura projekta

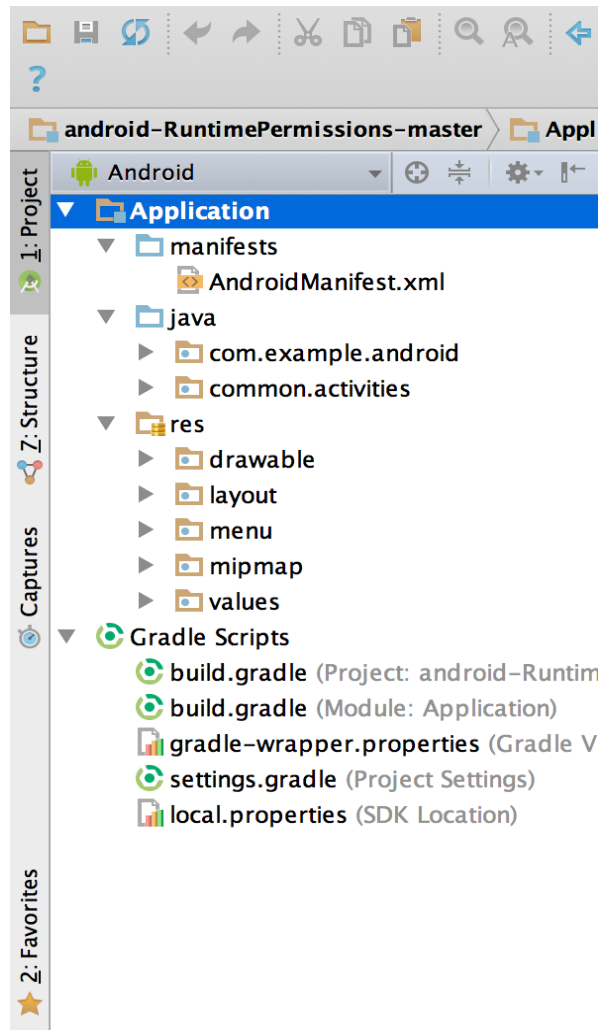
Vsak projekt v Android Studiu vsebuje enega ali več modulov z izvornimi kodnimi mapami in mapami virov. Tipi modulov so:

- aplikacijski Android moduli,
- knjižni moduli,
- aplikacijski Google moduli.

Android Studio samodejno prikaže dokumente v projektni vrstici. Vrstica je ustvarjena tako, da omogoča najhitrejši dostop do projekta in do datotek, ki jih vsebuje.

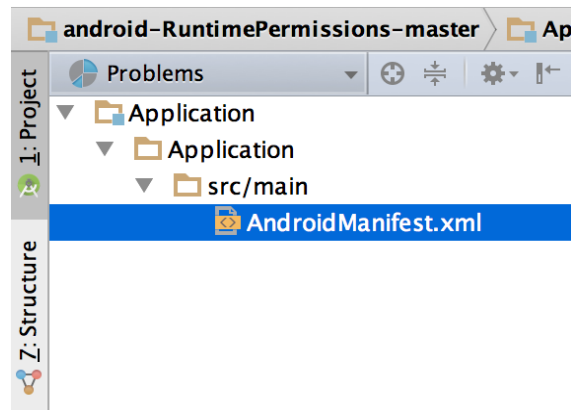
Vse mape gradnikov so prikazane na spodnjem stolpcu pod »Gradle Scripts« in vsak aplikacijski modul vsebuje naslednje mape:

- **Manifests:** vsebuje mapo *AndroidManifest.xml*, kjer uporabnik preureja aktivnosti aplikacije;
- **Java:** vsebuje vse izvirne mape in dokumente, ki so napisani v jeziku Java, vključno z dokumentom *JUnit* test code;
- **Res:** vsebuje vse ne-kodne vire, kot npr. *XML* razporeditve (layout), *UI* nize in *bitmap* slike.



Slika 11: Prikaz projekta v Android Studiu

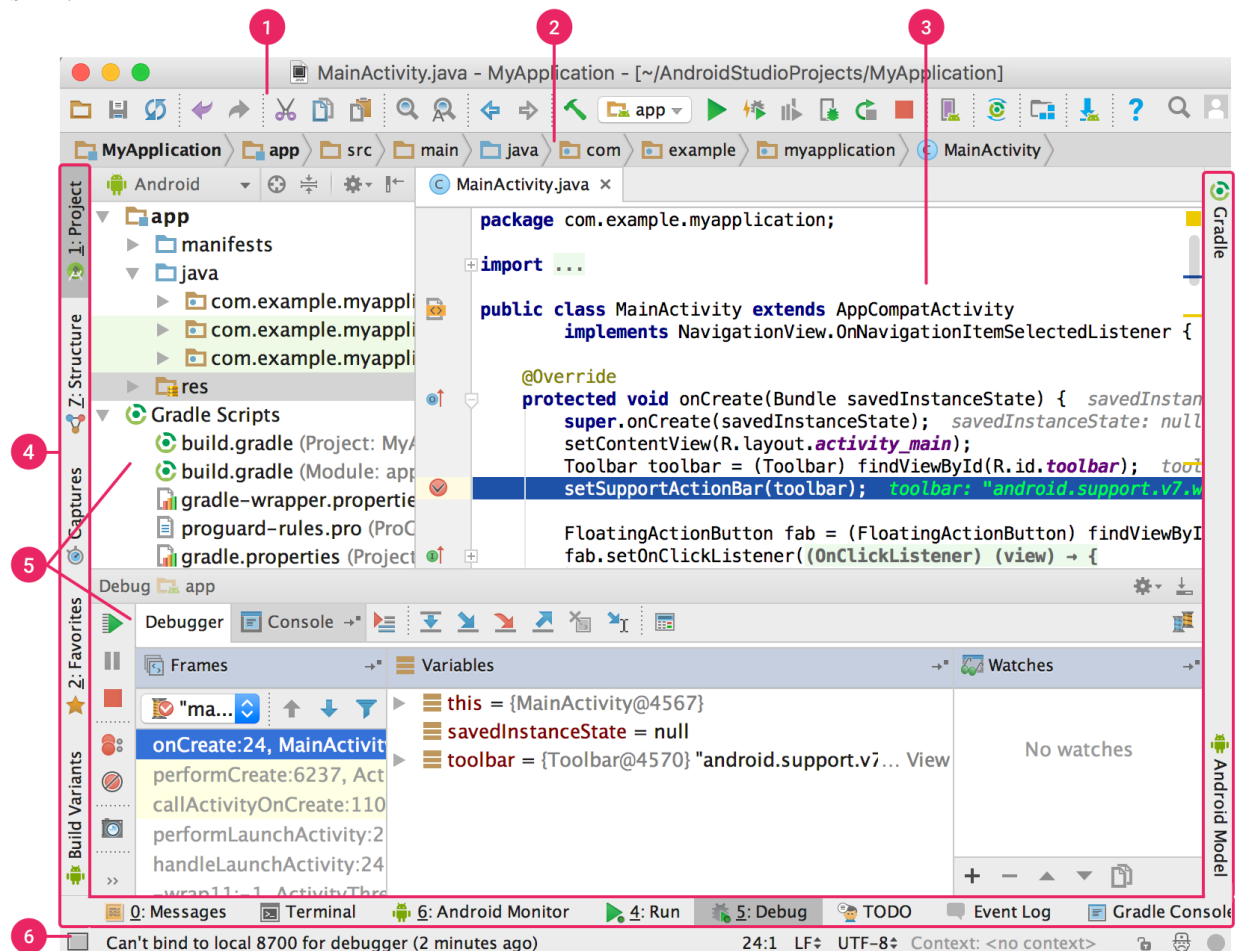
Po volji lahko tudi oblikujemo pogled projekta tako, da se osredotočimo na specifične vidike ustvarjanja aplikacije. Na primer izbira pogleda »Problems« namesto pogleda Android v povsem zgornjem levem kotu izpiše mape in dokumente, kjer se nahajajo napake, zaradi česar se aplikacija ne zažene oz. ne deluje.



Slika 12: Izbira pogleda "Problems"

## 5.2 Uporabniški vmesnik

Glavno okno programa je sestavljeno iz več logičnih delov, kar je prikazano na spodnji sliki:



Slika 13: Glavno okno programa Android Studio

1. Orodna vrstica nam omogoča uporabo različnih dejanj aplikacije vključno z zagonom aplikacije in uporabo orodij.
2. Vrstica za krmarjenje nam je v pomoč za upravljanje projekta in odpiranje map ter dokumentov za urejanje.
3. Urejevalno okno je prostor, kjer ustvarjamo in nadgrajujemo kodo.
4. Orodna okenska vrstica se nahaja ob stranskih koncih IDE okna in vsebuje gumbе, s katerimi lahko dostopamo do individualnih orodij za izboljšanje aplikacije.
5. Orodno okno nam omogoča dostop do specifične naloge kot na primer upravljanje projekta, iskanje, kontrolne verzije itd.
6. Vrstica stanja nam prikaže status našega projekta in vse možne napake v aplikaciji.

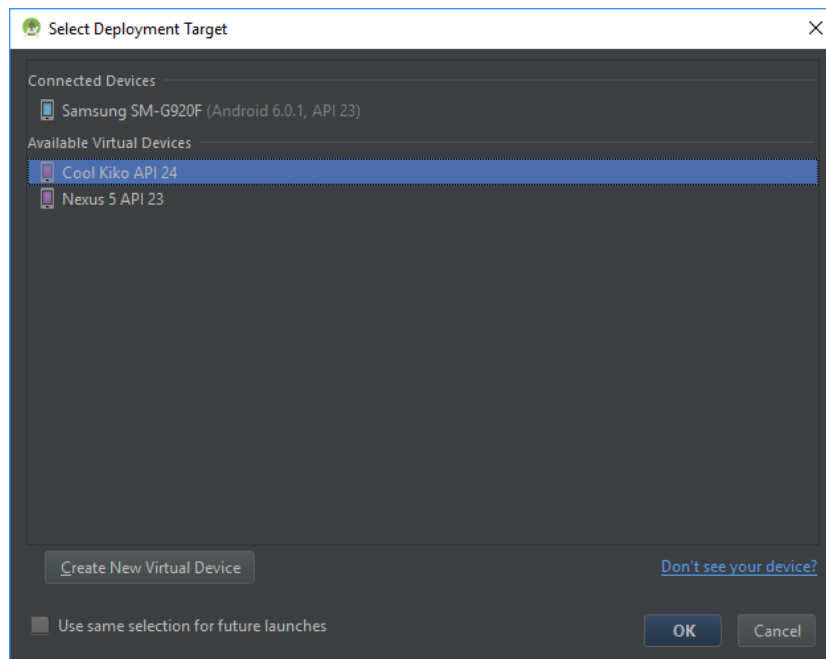
### 5.3 Android SDK

Android software development kit (SDK) vsebuje obširen paket ustvarjalnih orodij. Nekatera izmed teh orodij so »debugger«, knjižice, ročni emulator, dokumentacija in še več. Trenutno podpira računalnike, ki uporabljajo operacijske sisteme Linux, Mac OS X 10.5.8 in naprej ter Windows 7 in naprej. Od Marca 2015 SDK ni dostopen na platformi Android, vendar je razvijanje mogoče z uporabo posebnih aplikacij.

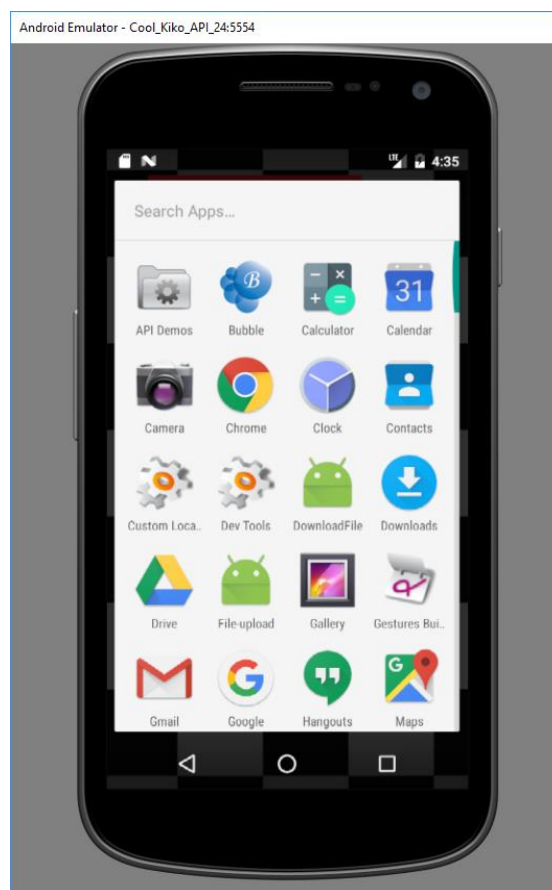
Android aplikacije so združene v paketih .apk formata in shranjene pod /data/app mapo v operacijskem sistemu Android (mape so dostopne samo glavnemu uporabniku iz varnostnih razlogov). APK paket vsebuje .dex datoteke, izvirne datoteke itd.

### 5.4 Emulator – Navidezna mobilna naprava

V paket SDK je vključena navidezna mobilna naprava, ki omogoča testiranje aplikacije in njenega vpliva na telefon. Za uporabo te naprave moramo omogočiti oz. ustvariti svoj emulator. V zgornji orodni vrstici kliknemo na gum AVD Manager, kjer lahko ustvarimo svojo navidezno mobilno napravo. Izberemo si lahko, kateri mobilni telefon bomo uporabili, v kateri API različici bo telefon prikazan in kako bo telefon poravnan, in sicer vodoravno ali navpično. Ko ustvarimo svoj emulator, ga zaženemo tako, da pritisnemo s klikom na gumb, ki zažene aplikacijo. Prikaže se nam okno, kjer izberemo svoj emulator v primeru, da jih imamo več. V tem oknu lahko tudi izberemo svoj mobilni telefon, če ga imamo povezanega na računalnik. Ko izberemo svoj emulator, se nam zažene in aplikacija se inštalira ter zažene sama. V primeru, da je kaj narobe, ali je kakšna napaka, se nam aplikacija ne zažene vse dokler napake ne razrešimo in ponovno zaženemo aplikacijo. Na spodnji sliki je prikazano okno, kjer lahko izberemo svoj ustvarjeni emulator ali telefon. Zraven telefona se nam še izpiše verzija Androida in API verzija telefona (v tem primeru Marshmallow).



*Slika 14: Prikaz okna za izbiro emulatorja oz. telefona*



*Slika 15: Galaxy Nexus Emulator*

## 6. ADOBE ILLUSTRATOR

### 6.1. Uvod v program

Adobe Illustrator je vsestransko uporaben program, idealen predvsem za oblikovanje manj obsežnih tiskovin kot so oglasi, plakati in letaki, ter za risanje logotipov, ikon ali ilustracij. Zaradi omogočanja preprostega in hitrega risanja, sestavljanja in kombiniranja vektorskih oblik in slike, ga oblikovalci vse pogosteje uporabljajo tudi za pripravo oblikovnih osnutkov spletnih strani.

### 6.2. Opis in zgodovina programa

Adobe Illustrator je bil razvit za Apple Macintosh decembra 1986 kot komercialni program oblikovanja splošnih oblik in izdelovanja fontov. Prva verzija programa je bila še zelo minimalistična in je omogočila le nekaj funkcij kot so risanje in izrezovanje oblik. Z uvedbo Adobe Illustrator7 v letu 1997, je Adobe dodal kritične spremembe v uporabniškem vmesniku glede urejanja. Dodali so mnogo funkcij in opcij za natančnejšo in preciznejšo oblikovanje. Program je omogočal urejevanje njihovih dokumentov tudi preko drugih operacijskih sistemov.

Illustrator CS je prva različica, ki je vključevala 3-dimenzionalne zmogljivosti, ki omogočajo uporabnikom, da oblikujejo in vrtijo oblike za ustvarjanje preprostih 3D objektov.

CS4 je izšel oktobra 2008. Ima številne izboljšave starih orodij, a uvaja nekatere povsem nove pridobljene iz FreeHand. Sposobnost, da ustvarite več artboardov, je eden glavnih dodatkov, ki je prav tako pridobljen iz programa FreeHand. Vseboval je tudi druga orodja, ki vključujejo Blob čopič, ki omogoča več udarcev prikrivanja.

CS5 je izšel oktobra 2010. Skupaj s številnimi izboljšavami obstoječih in novih funkcij Illustrator CS5 vključuje še orodje imenovano »Perspective Grid.« Program je odličen pripomoček, ki pomaga ljudem pri izdelovanju raznolikih projektov, in sicer ustvarjanje različnih oblik in objektov, ki jih lahko vsestransko spreminjamo z obstoječimi pripomočki.

Program lahko uporabljamo v večini operacijskih sistemov. Veliko ljudi se za ta program odloči, ker lahko z enostavnimi pripomočki ustvarijo raznolike projekte kot so na primer logotip, tridimenzionalno oblikovanje in risanje projektov. Zaradi tega smo se zanj odločili tudi mi.

### 6.3. Naše delo s programom

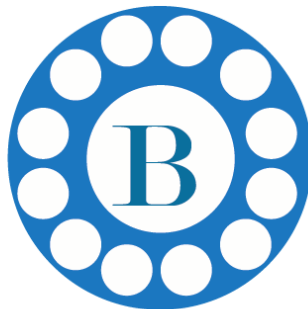
Tudi naša ekipa se je odločila za uporabo programa Adobe Illustrator. Ta program nam je omogočil enostaven in estetski razvoj našega logotipa in različnih gumbov, ki smo jih dodali v aplikaciji Bubble. Pomagal nam je tudi pri notranji arhitekturi aplikacije, saj smo lahko s tem programom narisali načrte za nadaljnji razvoj.

Naša prvotna ideja je bila logotip v obliki mehurčka, saj sovpada z imenom aplikacije. S to osnovno idejo smo pričeli z oblikovanjem loga. Prva verzija je bila sestavljena iz preprostega kroga, ki smo ga dopolnili z manjšimi belimi krogi. Ta logo je žal izgubil obliko mehurčka in prevzel obliko piškota, zato smo ga opustili.



*Slika 16: Prva verzija logotipa Bubble*

Nadaljevali smo z idejo, ki je prav tako vsebovala večji element v obliki kroga. Nanj smo dodali dvanajst manjših krogov in enega večjega. Za zaključek smo dodali črko "B" v sredino največjega kroga. Žal se je ta ideja izkazala kot napačna, saj je bila preveč zahtevna in agresivna, da bi si jo ljudje lahko hitro zapomnili.

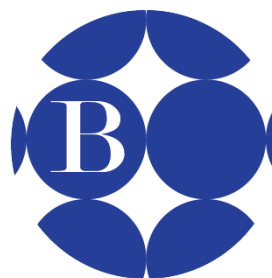


*Slika 17: Druga verzija logotipa Bubble*

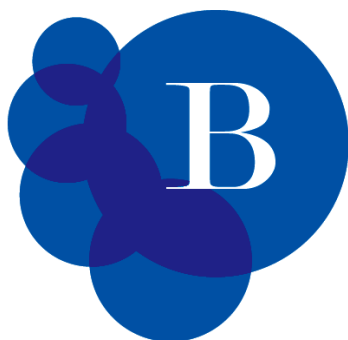
Ker nismo vedeli kako nadaljevati, smo narisali preprost moder krog kot osnovni element za nadaljnji razvoj logotipa. Ker naša aplikacija predstavlja bazo s poučnimi vsebinami, preko katere bi lahko individualne stranke dostopale, smo se odločili dodati štiri kroge z različnimi velikostmi. Tako se velikost kroga veča od leve proti desni. Krogi, ki se pripenjajo k osnovnemu krogu, naj bi prikazovali stranke, ki z uporabo te aplikacije nadgrajujejo svoje znanje. Logotip smo dodatno spremenili tako, da smo spremenili barvo, saj se nam je odtenek modre barve zdel preveč močan in neprimeren za takšno vrsto aplikacije. Barvo smo spremenili v svetlejši odtenek modre, da prikazuje neko nežnost, hkrati pa tudi produktivnost. Dodali smo še nekaj senc, ki omogočajo objektu, da izstopi iz dveh dimenzij. Tako se nam objekt zdi bolj realističen in naraven, saj ima iluzijo o trodimenzionalnosti. Tako smo končno pridobili obliko, s katero smo bili zadovoljni.



*Slika 18: Tretja verzija logotipa Bubble*



*Slika 19: Četrta verzija logotipa Bubble*



*Slika 20: Peta verzija logotipa Bubble*



*Slika 21: Zadnja verzija logotipa Bubble*



## 7. APLIKACIJA BUBBLE

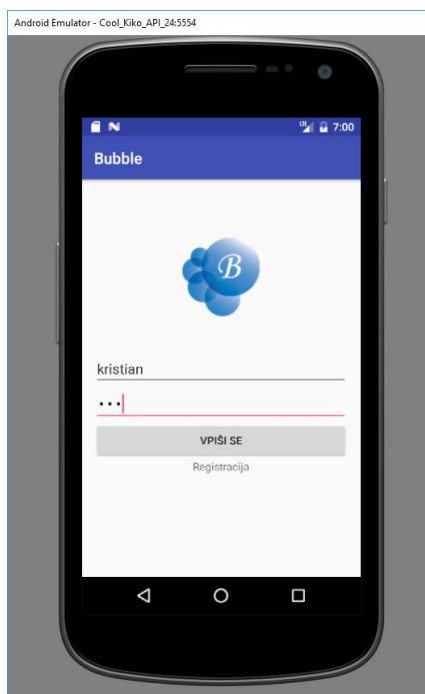
Razvili smo prvo verzijo aplikacije, kjer lahko uspešno naložimo in prenesemo datoteke na lokalni strežnik in ustvarimo svoj uporabniški račun in se z njim prijavimo v aplikacijo. Do zdaj so to vse možnosti, ki jih ponuja naša aplikacija. Vse deluje uspešno in večjih problemov glede aplikacije nimamo.

Skozi ves proces ustvarjanja aplikacije smo se sproti tudi učili, kajti nihče izmed nas ni povsem poznal jezika Java, zaradi česar smo si vsi skupaj med seboj pomagali. V ogromno pomoč nam je bil internet, kjer smo skupaj iskali rešitve in se medtem tudi učili. Izjemno priročna je spletna stran z imenom StackOverflow, kjer si ljudje s celega sveta pomagajo med seboj in skupaj ustvarjajo, zato smo tudi spoznali, da imamo podoben cilj - združiti ljudi ter jih pritegniti k skupnemu sodelovanju, učenju ustvarjanju in napredovanju.

Aplikacija je razdeljena na različne aktivnosti. Predstavili bomo vse aktivnosti v aplikaciji in podrobno razložili vsako izmed njih.

### 7.1. Zagon aplikacije

Ob zagonu aplikacije se nam prikaže prva aktivnost, kjer moramo vpisati svoje uporabniško ime in geslo. Oba fragmenta za uporabniško ime in geslo sta sestavljena iz t. i. tekstovnega pogleda. Vpisano geslo se nam prikaže zakrito. Nad tekstovnimi pogledi je prikazan logotip za aplikacijo Bubble.



*Slika 22: Aktivnost ob zagonu aplikacije*

Vpišemo lahko svoje uporabniško ime in geslo ter se prijavimo z obstoječim računom. V primeru, da nimamo uporabniškega računa, pritisnemo na spodnji gumb z imenom Registracija, kjer si ga lahko ustvarimo. Ob pritisku na gumb »Registracija« se nam prikaže druga aktivnost z imenom RegisterActivity. V tej aktivnosti si lahko ustvarimo svoj uporabniški račun. Podane imamo štiri tekstovne vrstice, kjer moramo navesti svoje podatke. Ob izpolnitvi pritisnemo na gumb z imenom »Registriraj se« in ustvari se nam uporabniški račun. Podatki se po registraciji nemudoma vpišejo v podatkovno bazo, če imamo vklopljeno internetno povezavo. Po ustvaritvi se lahko vrnemo na prejšnjo aktivnost in se prijavimo s svojim novo ustvarjenim uporabniškim računom.

## 7.2. Glavna aktivnost

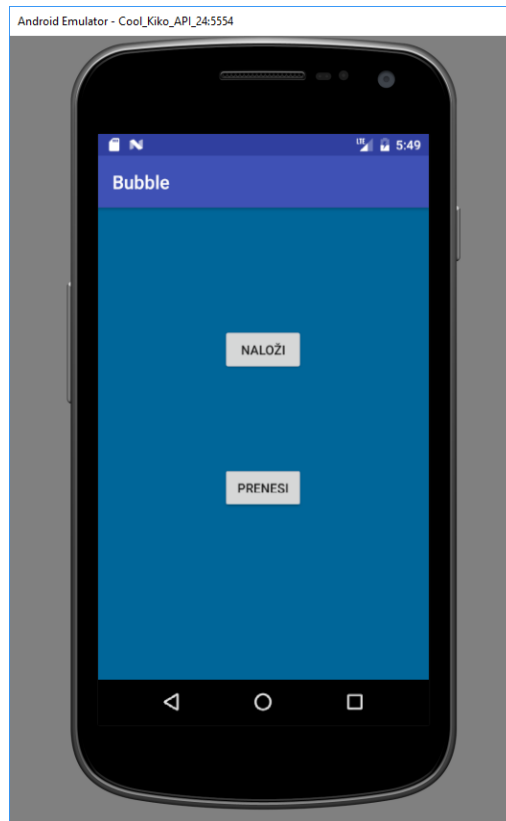
Po pritisku na gumb »Vpiši se« se uporabniku prikaže glavna aktivnost ali drugače rečeno MainActivity. V tej aktivnosti lahko vidimo 5 gumbov in predstavitveno poved, ki nam hitro razloži funkcijo te aplikacije. Prvi štirje gumbi, ki predstavljajo 4 področja, so razporejeni sorazmerno drug z drugim. Tekst je postavljen spodaj pod gumbi in takoj zatem lahko opazimo še zadnji gumb, ki nas popelje v aktivnost z imenom UserAreaActivity, kjer lahko spremenimo uporabniško ime in geslo uporabnika. Ozadje aplikacije razen prvih dveh aktivnosti je temno modre barve, saj se usklajuje z logotipom. Ob kliku na vsakega izmed štirih glavnih, se nam prikaže naslednja aktivnost, ki je poimenovana po področju, katerega smo izbrali. Aktivnosti za vsako področje so identične.



Slika 23: Glavna aktivnost

### 7.3. Ostale aktivnosti

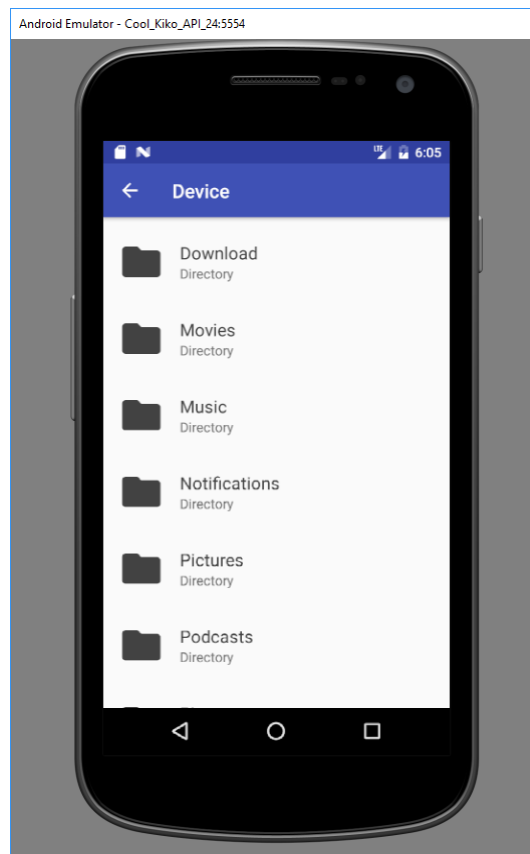
Ob kliku na gumb za določeno področje, npr. Slovenščina, se nam prikaže aktivnost z imenom »slo«. Na tem temno modrem ozadju sta prikazana dva navadna gumba, ki sta vzporedna drug z drugim. Ob kliku na prvi gumb lahko nalagamo datoteke, ob kliku na drugi gumb pa lahko prenašamo datoteke.



*Slika 24: Prva aktivnost za izbrano področje*

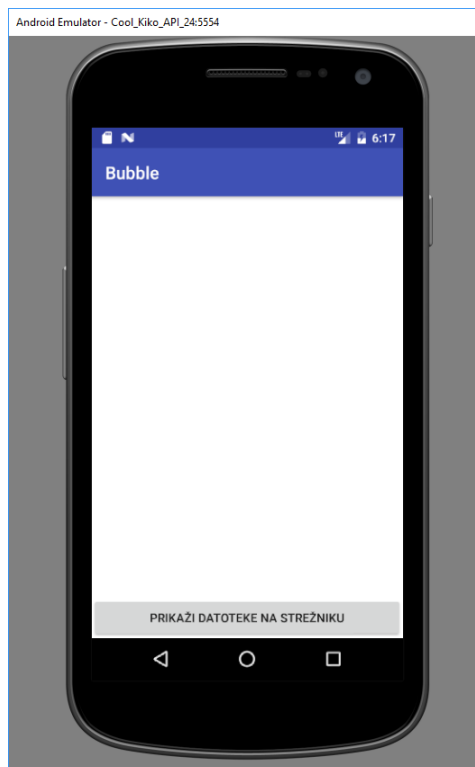
Če stisnemo na gumb »Naloži«, se nam prikažejo datoteke našega telefona. Za dostop do teh datotek in dokumentov smo morali uporabiti komando, ki nam zagotavlja dostop do notranjega pomnilnika našega telefona. Komanda je napisana v glavnem v glavnem .xml dokumentu naše aplikacije, ki je odgovoren za vse ustvarjene aktivnosti. Ko izberemo dokument, ki ga želimo naložiti na strežnik, se naš telefon poveže na URL, ki smo ga podali aplikaciji za dostop do strežnika in naloži dokument oz. datoteko. V primeru, da internetne povezave nimamo vzpostavljene, naložitev dokumenta ni bila uspešna.

Ker so določeni dokumenti večji od 8 MB, smo morali v xampp glavnem »phpconfig« dokumentu nastaviti maksimalno naloženo velikost na približno 50 MB. Velikost objavljenega dokumenta lahko spreminjamo po želji.

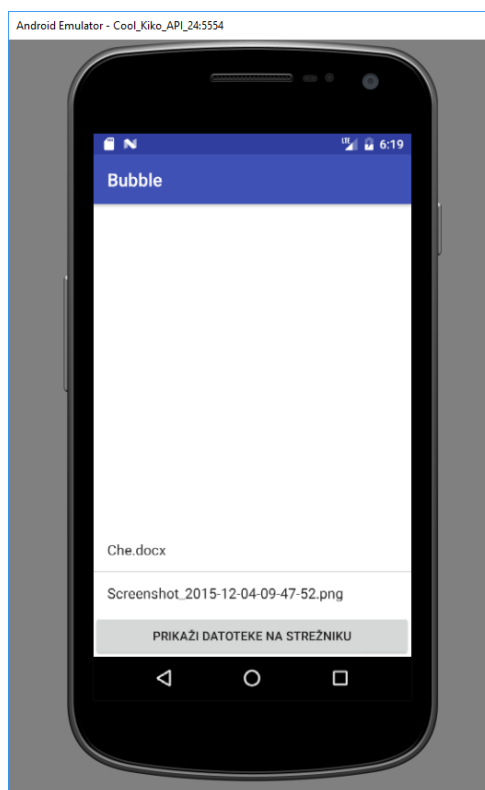


*Slika 25: Notranji pomnilnik telefona*

Če stisnemo na gumb »Prenesi«, se nam prikaže aktivnost z imenom »slo2«, kjer je prikazan 1 gumb, na katerega pritisnemo, da se nam prikažejo vsi dokumenti na strežniku za določeno področje, v tem primeru Slovenščina. Fragment v Android Studiu, na katerem se prikažejo dokumenti, se imenuje »List View«. List View nam omogoča, da po želji razporedimo prikaz določenih nizov v aplikaciji, zaseda določen prostor in zato v primeru, da imamo ogromno datotek na strežniku, lahko premikamo aktivnost aplikacije dol ali gor po želji.



*Slika 26: Aktivnost slo2 brez prikaza vsebine na strežniku*



*Slika 27: Aktivnost slo2 s prikazom vsebine na strežniku*

Če želimo aplikacijo izklopiti ali se vrniti v glavni meni, lahko to storimo tako, da se vrnemo v prvo aktivnost (LoginActivity) ter pritisnemo gumb na telefonu za nazaj. Pri določenih telefonih je ta gumb prikazan na samem uporabniškem vmesniku telefona.

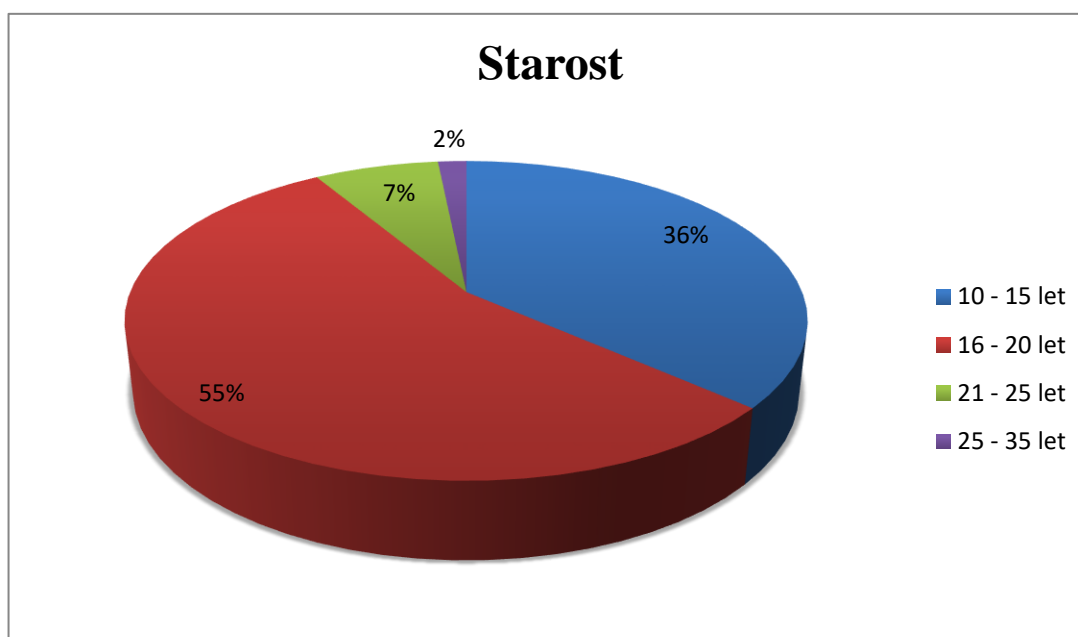
Zaenkrat izdelovanje in programiranje aplikacije ni predstavljalo prevelikih težav, vendar pričakujemo, da se bodo z določenimi izboljšavami pojavili novi izzivi.

## 7.1 ANKETA

Za lažje delo pri pridobivanju idej za aplikacijo, smo ustvarili kratek anketni vprašalnik. Zanimala nas je naša ciljna publika, ali anketiranci že poznajo osnovni koncept strani Dijaski.net, ali ga uporabljajo itd.

### 8.1. Starost anketirancev

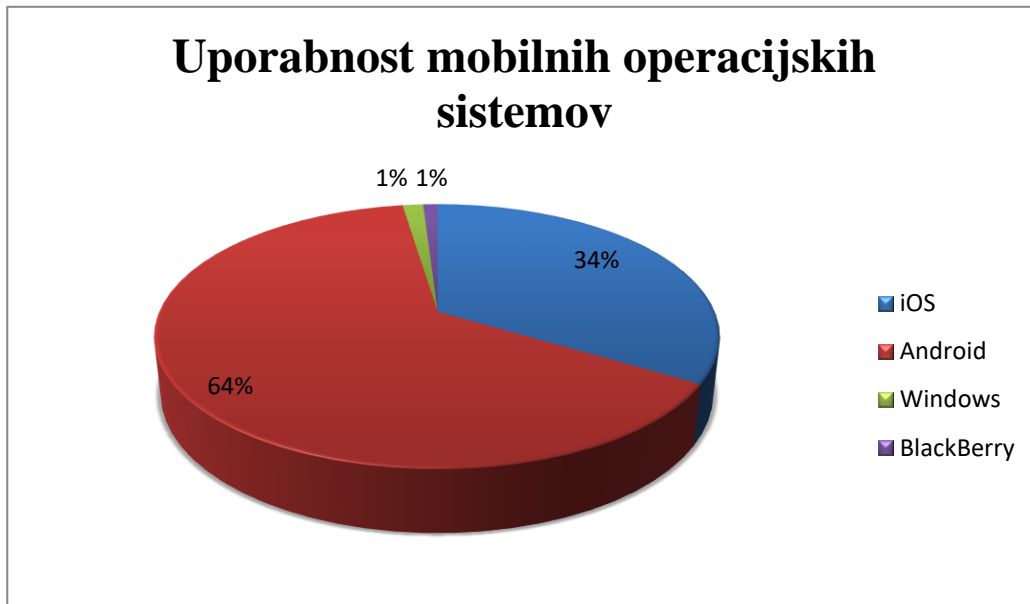
Okoli polovica anketirancev spada v starostno skupino 15 - 20 let, kar je tudi ciljna publika naše aplikacije – torej dijaki. Med ostale procente anketirancev spadajo študentje in osnovnošolci.



*Grafikon 2: Starost anketirancev*

## 8.2. Kakšni mobilni operacijski sistem uporabljate?

Več kot polovica anketirancev uporablja mobilni operacijski sistem Android. Na podlagi tega smo se odločili aplikacijo ustvariti v programu Android Studio, ki uporablja programski jezik Java. Preostanek anketirancev uporablja operacijski sistem iOS ali Windows, redki so uporabniki BlackBerry-ja.



*Grafikon 3: Uporabnost mobilnih operacijskih sistemov*

## 8.3. Ali poznate Dijaski.net?

Večina anketirancev že pozna spletno stran Dijaski.net. Stran je uporabna zaradi učnih snovi, vaj, domačih branj in seminarских nalog ki jih najdemo na njej. Predvidevamo, da je stran tako poznana zaradi njene vsebine.

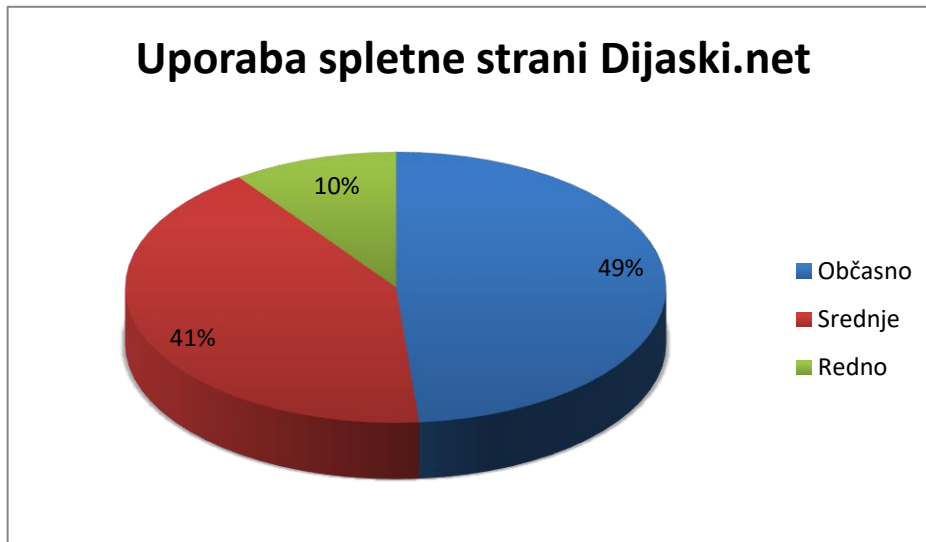


*Grafikon 4: Poznavanje strani Dijaski.net*



#### 8.4. Koliko uporabljate Dijaski.net?

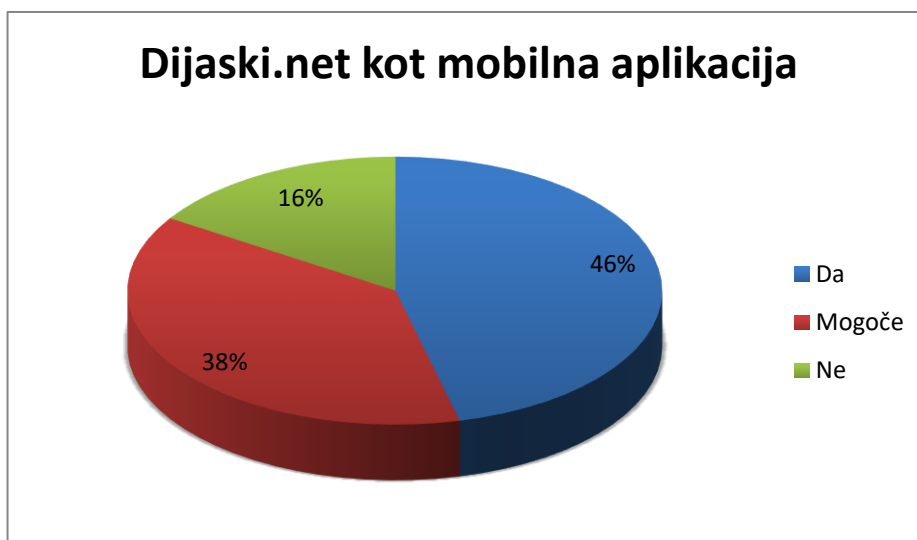
Zastavljeno vprašanje nam je posedovalo odgovore, da strani ne uporablja veliko ljudi. Približno 50 % anketirancev Dijaski.net uporablja malo. Najverjetneje zaradi določenih neustreznih vsebin, ki se nahajajo na njej. Preostali uporabljajo stran pogosteje ali redno.



*Grafikon 5: Uporaba spletne strani Dijaski.net*

#### 8.5. Bi vam koristil Dijaski.net v obliki mobilne aplikacije?

Večina anketirancev meni, da bi jim Dijaski.net v obliki mobilne aplikacije bolj ustrezal kot sama spletna stran. Menimo, da zaradi predstavljenih izboljšav aplikacije in slabe dostopnosti spletne strani preko telefona. Preostali delež meni, da bi aplikacija morda bila uporabna, nekateri pa se s samo idejo ne strinjajo.



*Grafikon 6: Dijaski.net kot mobilna aplikacija*

## 7.2 ZAKLJUČEK

Na začetku raziskovalne naloge smo predstavili orodja, v katerih smo izdelovali aplikacijo. Med njih spadata Adobe Illustrator in okolje Android Studio. Programirali smo v jeziku Java, ki se uporablja za izdelavo aplikacij v operacijskem sistemu Android. Za ta jezik smo se odločili na podlagi ankete, saj smo ugotovili, da večina dijakov in študentov uporablja telefone z operacijskim sistemom Android.

Prav tako smo predstavili tudi samo delovanje aplikacije, kako naložiti datoteke na strežnik ali telefon in kako se vpisati v samo aplikacijo, saj omogoča dostop le uporabnikom.

Kar se tiče praktičnega dela, smo se osredotočili na razvoj same aplikacije Bubble, ustvarjene v omenjenem programskem jeziku Java. Ugotovili smo, da bi aplikacija zelo koristila številnim dijakom in študentom ter tudi osnovnošolcem, ki si želijo večjega in razširjenega znanja, ali morda le pomoč pri učenju.

### 7.3 DRUŽBENA ODGOVORNOST

Raziskovalna naloga pripomore k razvijanju skupnega dela in lažjega učenja dijakov ter študentov. Trenutno naša aplikacija pomaga ljudem pri nalaganju in prenašanju poljubnih dokumentov po želji. Glede na to, da bi naj imela še nekaj dodatkov, kot so klepetalnice in učilnice, bi prav tako pripomogli k medsebojnemu sodelovanju in komuniciranju dijakov in študentov. Trenutno v ideji naše aplikacije ne vidimo nobene pomanjkljivosti razen aplikacije same, ki jo moramo vizualno in programsko spremeniti.

## VIRI

- [1] XAMPP. Wikipedija [online]. 2017. [31.1.2017]. Dostopno na: <https://sl.wikipedia.org/wiki/XAMPP>
- [2] LAMP. Wikipedija [online]. 2017. [3.2.2017]. Dostopno na: <https://sl.wikipedia.org/wiki/LAMP>
- [3] Statistika Android. Statista. [online]. 2016. [4.2.2017]. Dostopno na: <https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/>
- [4] Kaj je Android?, OHA, Prednosti. Slo-Android. [online]. 2011. [2.2.2017]. Dostopno na: <https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/>
- [5] Zgradba operacijskega sistema Android. Wikipedija. [online]. 2016. [4.2.2017]. Dostopno na: [https://sl.wikipedia.org/wiki/Android\\_\(operacijski\\_sistem\)](https://sl.wikipedia.org/wiki/Android_(operacijski_sistem))
- [6] Stanja aktivnosti in prehodi med njimi. Monitor. [online]. 2012. [4.2.2017]. Dostopno na: <http://www.monitor.si/clanek/sam-svoj-programer3/125026/>
- [7] Slika. Android. Mobilitaria. [online]. 2017. [1.2.2017]. Dostopno na: <http://www.mobilitaria.com/tag/android>
- [8] Slika, Grafikon. Zgodovina. Developer-Android. [online]. 2016. [2.2.2017]. Dostopno na: <https://developer.android.com/about/dashboards/index.html>
- [9] Slika, Zgradba operacijskega sistema. CodeAid. [online]. 2013. [4.2.2017]. Dostopno na: <http://www.codeaid.in/android/architecture-of-android.html>
- [10] Slika, Življenski cikel aktivnosti. Monitor. [online]. 2012. [4.2.2017]. Dostopno na: <http://www.monitor.si/clanek/sam-svoj-programer3/125026/>
- [11] Android SDK. Wikipedija. [online]. 2016. [5.2.2017]. Dostopno na: [https://en.wikipedia.org/wiki/Android\\_software\\_development](https://en.wikipedia.org/wiki/Android_software_development)
- [12] Struktura projekta, Uporabniški vmesnik. Developer-Android. [online]. 2016. [5.2.2017]. Dostopno na: <https://developer.android.com/studio/intro/index.html>
- [13] Slika, Android Studio logotip. itsandroidhelp. [online]. 2016. [5.2.2017]. Dostopno na: <http://itsandroidhelp.blogspot.si/>
- [14] Slika, Prikaz projekta v Android Studiu, Izbira pogleda. itsandroidhelp. [online]. 2016. [5.2.2017]. Dostopno na: <https://developer.android.com/studio/intro/index.html>

- [15] Verzije programa Adobe Illustrator. Wikipedija. [online]. 2016. [4.2.2017]. Dostopno na: [https://en.m.wikipedia.org/wiki/Adobe\\_Illustrator](https://en.m.wikipedia.org/wiki/Adobe_Illustrator)
- [16] Časovna preglednost programa Adobe Illustrator. Adobe. [online]. 2016. [5.2.2017]. Dostopno na: <http://www.adobe.com/si/products/illustrator/features.html>
- [17] Funkcije programa Adobe Illustrator. Poslovno-kreativnasola. [online]. 2015. [6.2.2017]. Dostopno na: <http://poslovno.kreativnasola.si/programi/adobe-illustrator/>