

Mladi za napredek Maribora 2016

33. srečanje

MOBILNA APLIKACIJA FAST ORDER

Raziskovalna naloga: Računalništvo

Avtor: PATRICK KAČIČ

Mentor: ZDRAVKO KAČIČ

Šola: SREDNJA ELEKTRO-RAČUNALNIŠKA ŠOLA

Maribor, 2016

Kazalo vsebine

Kazalo slik.....	3
Kazalo tabel.....	3
Zahvala.....	4
1 Povzetek	5
2 Problem	6
3 Funkcionalnost aplikacije.....	7
3.1 Pridobivanja podatkov	7
3.1.1 NFC - Near Field Communication	9
3.1.2 Storitve v oblaku – podatkovna baza	9
3.2 Grafične vmesniške komponente mobilne aplikacije	10
3.3 Naročanje	12
3.4 Plačevanje	12
4 Platforma Android.....	13
4.1 Kaj je platforma Android	13
4.2. Zgodovina.....	13
4.3 Arhitektura platforme Android	15
4.4 Življenjski cikel aktivnosti Android aplikacije	17
4.5 Prednosti platforme Android	18
5 Zagonsko in razvojno okolje.....	18
5.1 Razvojno okolje Android Studio	19
5.2 Android SDK	19
5.3 Android API	20
5.4 Navidezna mobilna naprava (Emulator)	20
6 Razvoj aplikacije FastOrder.....	21
6.1 Zagon aplikacije	21
6.2 Glavna aktivnost.....	21
6.3 Naslednje aktivnosti	26
6.3.1 Aktivnost Meniji - izbrana	26
6.3.2 Aktivnost Jedi – izbrana.....	28
6.3.3 Aktivnost Pijače – izbrana	29

6.3.4 Aktivnost Akcije - izbrana	31
6.4 Aktivnost Naročila	31
6.5 Aktivnosti v navigacijskem predalu.....	31
7 Sklep.....	33
Viri	33

Kazalo slik

Slika 1: prikazuje potek komunikacije ob uporabi sistema hitrega naročanja FastOrder.....	8
Slika 2: pojmovni diagram računalništva v oblaku.....	10
Slika 3: prikazuje aktivnosti predal.....	11
Slika 4: prikazuje akcijsko vrstico.....	11
Slika 5: najbolj uporabljene različice Android OS.....	15
Slika 6: arhitektura operacijskega sistema Android.....	16
Slika 7: življenjski cikel aktivnosti Android aplikacije.....	18
Slika 8: logotip Android Studio.....	19
Slika 9: API za različne verzije Androida.....	20
Slika 10: diagram poteka aplikacije FastOrder.....	21
Slika 11: fragment Meniji v glavni aktivnosti.....	23
Slika 12: fragment Jedi v glavni aktivnosti.....	24
Slika 13: fragment Pijače v glavni aktivnosti.....	25
Slika 14: fragment Akcije v glavni aktivnosti.....	26
Slika 15: aktivnost s seznamom kategorij menijev.....	27
Slika 16: aktivnost s seznamom kategorij jedi.....	29
Slika 17: aktivnost s seznamom kategorij pijač.....	30
Slika 18: navigacijski predalnik.....	32

Kazalo tabel

Tabela 1: različice Android OS in njihovi dodatki/popravki.....	15/16
---	-------

Zahvala

Najprej bi se rad zahvalil mojemu mentorju, ki mi je z njegovim znanjem in izkušnjami pomagal skozi celoten proces in brez njegove pomoči ne bi naredil tako uspešne naloge. Zahvaliti se moram tudi dvema zaposlenima na FERI-ju, ki sta mi pomagala in svetovala pri izdelavi prototipa.

1 Povzetek

Navdih za izdelavo raziskovalne naloge sem dobil, ko sem se hotel naučiti večine programiranja oz. celotnega procesa izdelave aplikacije android. V osnovi je ideja sistem hitrega oz. informativnega naročanje hrane in pijače v restavracijah. Z uporabo sistema uporabniku ni potrebno čakati na natakarja, pač pa se s svojim mobilnim telefonom poveže s strežnikom, preko NFC komunikacije pa sistemu sporoči številko mize in sedeža v restavraciji, za katero se nahaja. S pomočjo sistema dobi vpogled v gostinsko ponudbo restavracije, po izbiri in naročilu hrane in pijače, lahko s svojim mobilnim telefonom izvede tudi plačilo naročila (npr. preko sistema mobilnega plačevanja Moneta), ali pa preko aplikacije pokliče natakarja. Z namenom čim uspešnejše uporabe sistema sem najprej preučil sedanji sistem naročanja v restavracijah, ga primerjal s predlaganim sistemom hitrega naročanja in določil prednosti predlaganega sistema. Odločil sem se, da bom aplikacijo izvedel za mobilne telefoni z operacijskim sistemom Android, zato sem za pisanje programske kode uporabil programski jezik Java. Pri načrtovanju sistema sem najprej definiriral diagram poteka in akcij, ki prikazuje prehode med stanji oz. zaslone, ko uporabnik uporablja aplikacijo in akcije, ki jih pri tem izvaja. Definiran diagram poteka in akcij je bil osnova za izvedbo programske kode in ustvarjanje aplikacije. Najprej sem sprogramiral funkcionalnost aplikacije, nato pa definiriral grafični izgled posameznih zaslonov in jih vnesel v kodo.

Summary

I got inspiration for this research work when I wanted to learn the virtues of programming and the process of making android apps. The idea is quick and informative ordering of food and drinks in restaurants. With sage of this system the user would not need to wait for the waiter, instead user will simply connect with his phone to the server. Via NFC communication system gets number of table and seat in restaurant in which user located. With this system the user gets a view of what this restaurant offers, after the choice and ordering of selected food and drink he can simply make a payment with his phone (e.g., through Moneta) or using the application he can call the waiter. In order to optimize the use of the system, I first had to elaborate the current system of ordering in restaurants, compare it with the proposed idea and determine the advantages of the fast ordering system. I decided that I will develop application for Android system therefore I used the Java programming language for programming. In the design of the system, I first created a flowchart showing the transitions between the states/screens when a user is using the application and actions in time he has implemented. Defined flowchart and actions were the basic for making programming code and creation of application. First, I programmed functionality of application, then I defined graphic design of each screens and converted it into code.

2 Problem

Ste že kdaj prišli v lokal oz. restavracijo, ki je bila polna gostov, se usedli za mizo in čakali natakarja, da je prišel do vas. Preden natakar utegne priti do vas, mine v takšnih primerih običajno kar nekaj časa, pogosto preveč, sploh, če se vam kam mudi. Ste pozorni na to, kaj jeste in bi radi vedeli sestavine oz. hranilne vrednosti hrane, ki jo boste zaužili v restavraciji? Morda vas zanima, kaj pravijo drugi, ki so naročili enako jed? Zelo nerodno je spraševati natakarje o mnenju gostov o posamezni jedi, prav tako pa pogosto natakarji tudi ne vedo podrobnosti o hrani, ki jo strežejo. Zato smo zasnovali mobilno aplikacijo, ki predstavlja rešitev za omenjene probleme.

Današnji svet sloni na tehnologiji in v naših življenjih so vse bolj prisotni mobilni telefoni, zato smo zasnovali mobilno aplikacijo, ki bo omogočala hitro naročanje in ponujala različne informacije o naročeni hrani ter pijači. Vključevala bo funkcionalnost že ustvarjenih aplikacij, kot so na primer funkcije za naročanje hrane od doma, aplikacije, ki omogočajo vpogled v menije oz. ponudbe različnih storitev. Predlagan sistem naročanja bo v pomoč gostincem, saj bodo lahko obvladovali večje število gostov, gostje bodo bolje informirani o naročeni hrani in pijači, prav tako pa bodo lahko tudi hitreje naročali in plačevali. Aplikacija bo še posebej uporabna za dijake in študente oz. mlade, ki pogosto hitijo in želijo hitro naročiti in plačati. Plačevanje je danes večinoma gotovinsko ali s kartico, vse pogosteje pa je tudi preko mobilnih telefonov. Vendar mora gost v vseh primerih poklicati natakarja, da pride do gosta, da lahko ta opravi plačilo. Predlagana rešitev rešuje tudi ta problem, saj je predvideno tudi plačevanje z mobilnim telefonom in sicer tako, da se preko lokalne brezžične povezave telefon poveže s strežnikom oziroma z blagajno in izvede plačilo (pri tem je lahko uporabljen kateri od že uveljavljenih sistemov plačevanja z mobilnimi telefoni – na primer Moneta). Uporabnik tako ob potrditvi naročila določi tudi način plačila. Če je izbral plačilo z mobilnim telefonom, lahko ob potrditvi naročila izvede tudi plačilo. Če bo želel plačati z gotovino, bo osebje pri blagajni ob naročilu videlo tudi način plačila in ko bo natakar prinesel naročilo, bo lahko hkrati prinesel tudi račun, ki ga bo gost nato plačal. Enako bi bilo tudi pri plačilu s kreditno kartico. Naročilo gostov je posredovano tudi direktno v kuhinjo, kar bo omogočalo šefu kuhinje pregled nad naročili gostov, ter tako pripomoglo k manjšemu številu nesporazumov med natakarji in kuharji.

V primeru, da ima gost več časa, si lahko ogleda podrobnejše informacije o hrani, ki jo želi naročiti, še posebej, če ima kakšno dieto ali pa ima prepoved uživanja kakšne sestavine. Lahko si prebere tudi mnenja in ocene drugih gostov, ki so že naročili posamezno jed ali pijačo v tem restavraciji ter jo ocenili. Aplikacija hrani informacije o tem, kaj je uporabnik naročal v posameznem lokalu v preteklosti, tako lahko uporabniku ob naslednjem obisku predlaga ponudbo, glede na njegov okus. Gostom aplikacija omogoča tudi izbiro akcijske ponudbe, prav tako tudi dodatne popuste, npr. ob 10 naročilu hrane v isti restavraciji, bi bilo naslednje naročilo zastoj ali pa z veliko večjim popustom.

3 Funkcionalnost aplikacije

Pri zasnovi aplikacije smo upoštevali, da mora biti aplikacija uporabniško prijazna, vendar mora hkrati obdržati dobro funkcionalnost. Z aplikacijo želimo gostu omogočiti predvsem hitro naročanje, osebju v restavraciji pa zmanjšati obseg dela ter jim omogočiti bolj sproščeno in manj stresno delo. Postopek od vstopa gosta v lokal, do uspešnega plačila smo razdelili na tri pomembne dele.

Da si gost lahko ogleda ponudbo lokala oz. izvede naročilo, mora aplikacija vsebovati podatke oziroma informacije o restavraciji, ki jih pridobi iz podatkovne baze, ki se v tem primeru nahaja v oblaki storitvi. V bazi je shranjenih veliko podatkov različnih restavracij. Da aplikacija ve, katere podatke mora prenesti, uporablja z identifikacijo restavracije in mize oziroma mesta, na katerem se nahaja gost, brezžično komunikacijo NFC (near field communication). Modul NFC je vgrajen v vsako mizo oziroma mesto, na katerem se lahko nahaja gost v restavraciji.

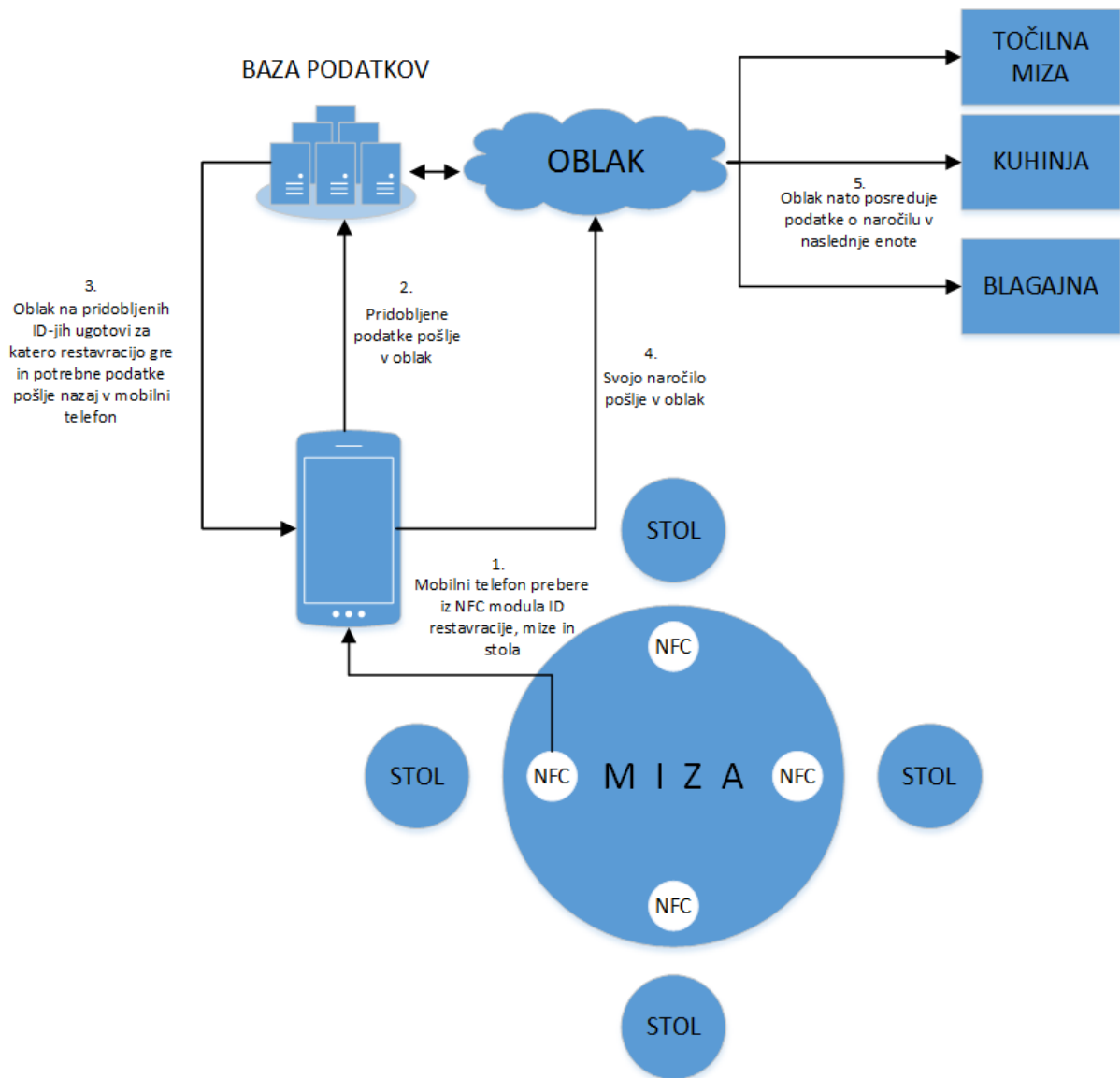
Ko se aplikacija poveže s sistemom, lahko pričnemo z uporabo aplikacije. Aplikacija je zgrajena tako, da že ob prvi uporabi sama uporaba aplikacije uporabnikom ne povzroča nobenih težav. Ne samo, da lahko gost naroča živila, lahko tudi izve sestavine hrane oziroma njene hranilne vrednosti. Pred naročilom se lahko preko komentarjev pozanima o mnenju drugih gostov, ki so želena hrano v tej restavraciji že naročili in komentirali njeno kvaliteto oziroma izvedli njeno oceno. Ima možnost izbire živil, ki so v akciji ali je deležen aktualnih popustov.

Ko izbere vso želena hrano in pijačo je naročilo pripravljeno in ga lahko odda. Pri tem določi tudi način plačila. Tega lahko izvede na 3 načine: z gotovino, plačilno kartico ali pa z uporabo plačilnih storitev preko mobilnega telefona.

Aplikacija omogoča tudi rezervacijo mize v restavraciji za v naprej določeno časovno obdobje, v kolikor restavracija ponuja možnost rezerviranja miz.

3.1 Pridobivanja podatkov

Aplikacijo smo zastavili tako, da ta v oblaki storitvi, in sicer v podatkovni bazi, shranjuje potrebne podatke o posameznih restavracijah, kot so: naslov in osnovni podatki o restavraciji, meniji, ki jih restavracija ponuja, jedi, pijače, cene, akcijske ponudbe itd. Te podatke v podatkovno bazo vnašajo zaposleni v restavraciji in jih urejajo preko grafičnega vmesnika spletnega brskalnika. Če ponudijo kakšno novost ali spremembo ponudbe, lahko podatke nadgradijo. Oblačna storitev omogoča dostop do podatkov kjerkoli po svetu, potrebna je samo internetna povezava.



Slika 1: Potek komunikacije ob uporabi sistema hitrega naročanja FastOrder

Uporabnik dostopa do podatkov o restavracijah, ki so shranjeni v oblaku, tako, da, ko zažene aplikacijo, ta najprej izvede identifikacijo lokala, v katerem se uporabnik nahaja. Pozove ga, da vključi modul za NFC komunikacijo, v kolikor je ta izklopljen, in telefon položi na oznako NFC, ki se nahaja na mizi in označuje položaj modula komunikacije NFC, vgrajenega v mizo za posamezen sedež. V mizo je vgrajenih toliko modulov NFC, kolikor je predvidenih sedežev za mizo. Vsak modul NFC vsebuje identifikacijsko številko (ID) restavracije (vsaka restavracija ima svoj unikatni ID), in številko mize ter sedeža, na katerem sedi gost, ki uporablja sistem hitrega naročanja. S prebranimi identifikacijskimi podatki lahko aplikacija prenaša iz baze samo tiste podatke, ki se nanašajo na določeno restavracijo.

Generiran ID lokacije gosta (stola in mize), ki se določi ob uspešnem prenosu podatkov med aplikacijo in modulom NFC, služi kot potreben podatek za komunikacijo s podatkovno bazo in točilno mizo oziroma blagajno, kot tudi za kuhinjo. Ko gost odda naročilo, bodo natakarji zaradi

dodeljenega ID mize vedeli, za katero mizo in na katerem mestu za mizo sedi gost, prav tako pa dobijo informacijo o naročilu tudi v kuhinji, tako da lahko takoj prično s pripravo naročila. Natakariji tudi ne bodo imeli težav z iskanjem mize za dostavo naročila. Enako pri plačevanju, še posebej pri plačevanju z mobilnim telefonom. V nadaljevanju podrobneje predstavljamo posamezne elemente, ki so pomembni za pridobivanje podatkov.

3.1.1 NFC - Near Field Communication

NFC (ali near field communication) v osnovi pomeni brez-kontaktno komunikacijo. Je vrsta prenosa podatkov med dvema napravama, ki deluje na podobnem principu, kot tehnologija RFID. NFC značke (»tagi«), kot so poimenovane različne NFC nalepke in drugi NFC izdelki, vsebujejo majhen čip, ki ne potrebuje napajanja, saj ga požene bližina NFC naprave, ki pa je lahko oddaljena le nekaj centimetrov (do približno 5cm).[1]

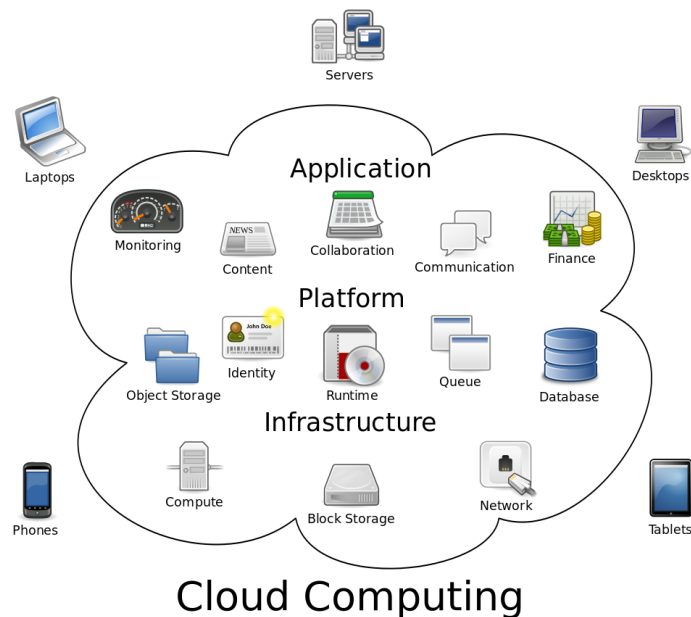
NFC ponuja v mobilni telefoniji celo vrsto novih možnosti. V značke NFC lahko zapišemo kontaktne podatke, kratka sporočila, sporočilo mobilni napravi za menjavo profila, klic osebe, pošiljanje kratkega sms sporočila, v našem primeru pa sporočanje ID restavracije, mize in sedeža mobilnemu telefonu.[1]

3.1.2 Storitve v oblaku – podatkovna baza

Storitve v oblaku oz. računalništvo v oblaku je področje računalništva, pri katerem so dinamično razširljiva in pogosto virtualizirana računalniška sredstva na voljo kot storitev preko interneta.[2]

V osnovi koncept računalništva v oblaku združuje pojme:

- programska oprema, kot storitev (*software as a service - SaaS*)
- platforma, kot storitev (*platform as a service - PaaS*)
- infrastruktura, kot storitev (*infrastructure as a service - IaaS*);



Slika 2: Pojemovni diagram računalništva v oblaku[13]

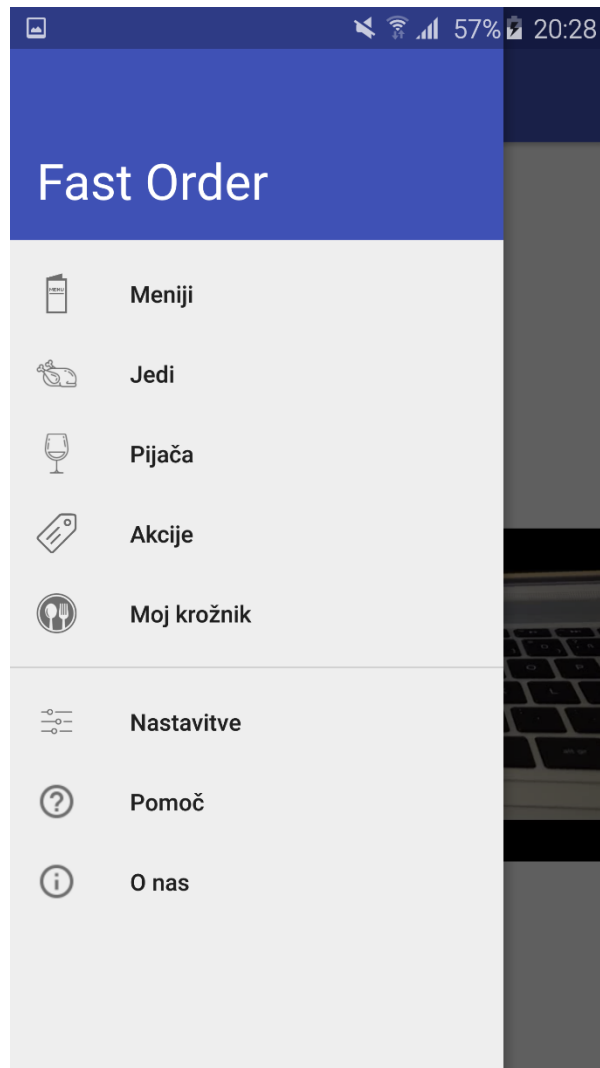
3.2 Grafične vmesniške komponente mobilne aplikacije

Glavne grafične vmesniške komponente aplikacije, ki se z njimi sreča uporabnik so zaslonske slike, aktivnosti predal in akcijska vrstica.

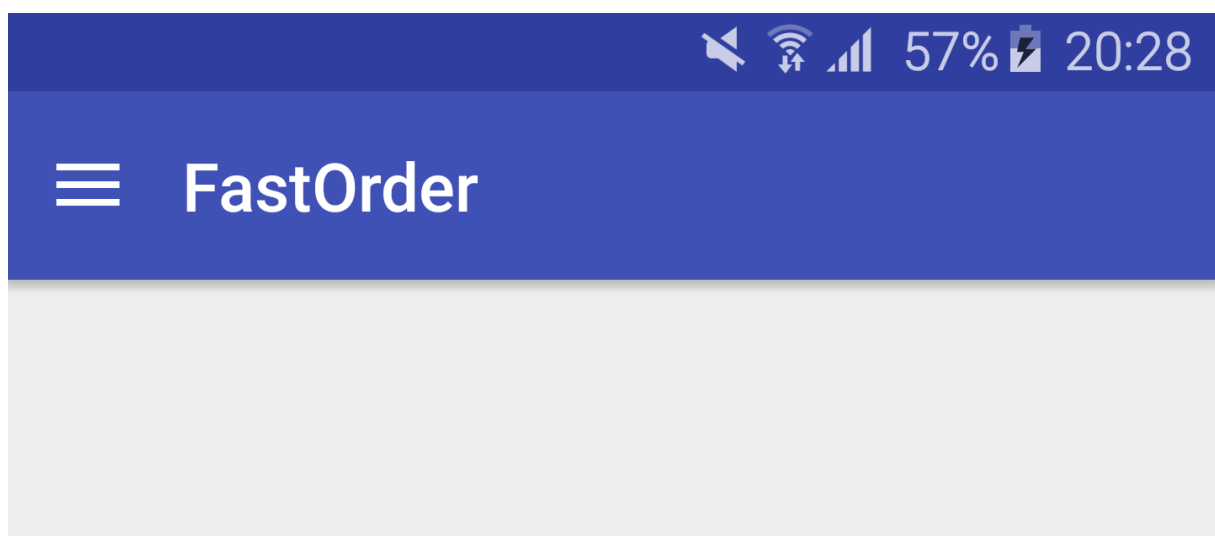
Zaslonska slika oziroma Fragment predstavlja vedenje ali del uporabniškega vmesnika v aktivnosti. V eni aktivnosti lahko uporabimo več različno oblikovanih fragmentov, ki predstavljajo več-okenski uporabniški vmesnik. Fragment ima svoj življenjski cikel, prejem svoje dogodke vnosa, ki jih lahko dodajaš ali uničuješ med tem ko aktivnost normalno deluje, je kot nekakšna pod-aktivnost.

Aktivnosti predal je nekakšen »drop-down« seznam, ki lahko vsebuje dialoge in slike. Pogosto se nahaja v zgornjem desnem kotu zaslona in ima znano ikono treh vodoravnih črt. Ko se aktivira prekrije večji del zaslona z svojim seznamom razen levi del je zamegljen, če pa se odpira iz desne strani pa obratno (slika 3).

Akcijska vrstica je ena izmed najbolj pomembnih elementov v aplikaciji, prisotna je v večini android aplikacij. Nahaja se v zgornjem delu zaslona in vsebuje najpomembnejše elemente, s katerimi se uporabniki orientirajo po aplikaciji, kot na primer aktivnosti predal, nastavitve, iskanje po aplikaciji ipd. (slika 4).



Slika 3: Aktivnostni predal



Slika 4: Akcijska vrstica

3.3 Naročanje

Zasnova aplikacije je sestavljena iz treh glavnih delov. Ko aplikacija pridobi vse potrebne podatke za prikazovanje, lahko pričnemo z uporabo aplikacije.

Pričnemo v prvem delu oziroma na glavni strani. Na njej lahko s pomiki zaslonских slik levo ali desno izbiramo med štirimi kategorijami. Če želimo pregledati menije, ki jih ponuja restavracija, izberemo aktivnost Meniji tako, da pritisnemo na ikono Meniji. Enako storimo, če želimo izbrati jedi, pijače in akcije. Vsaka aktivnost ima svojo ikono in napis. Če ne želimo listati zaslonских slik oziroma želimo izbirati po hitrejšem načinu, nam tukaj pomaga aktivnosti predal, ki se nahaja na vseh zaslonih oziroma aktivnostih in sicer v zgornjem levem kotu oziroma v akcijski vrstici, ki je prav tako zmeraj prisotna. Iz navigacijskega predala lahko prehajamo tudi na nastavitve, se pozanimamo za pomoč ali pa lahko pregledamo informacije o restavraciji. Če izberemo eno izmed kategorij, se izpiše seznam kategorij izbrane zvrsti (če smo izbrali Meniji, se izpiše seznam kategorij raznih menijev). Če nato izberemo eno izmed kategorij, se pokaže zaslon, s seznamom živil pod kategorijo, ki smo jo izbrali. Na zaslonu lahko preverimo, katera živila so bila največkrat izbrana.

Drugi del aplikacije vključuje aktivnost, ki omogoča podrobnejši pregled podatkov za živilo, ki smo ga izbrali. Za hrano lahko pogledamo hranile vrednosti in sestavine. Na zaslonu je predstavljeno tudi slikovno gradivo, tako da lahko vidimo, kako bo jed oziroma pijača izgledala. Če smo s ponudbo zadovoljni, jo lahko dodamo na seznam naročil, ki je del zadnjega, tretjega dela zasnove aplikacije.

V aktivnosti naročilo se zbirajo vse izbrane ponudbe. Seznam nam na koncu prikaže skupno ceno, ki jo moramo plačati. Kako izvedemo plačilo in kako poteka posamezna oblika izvedbe plačila predstavljamo v naslednjem poglavju.

3.4 Plačevanje

Zaradi dodelane identifikacijske številke (ID) gosta, lahko blagajna lažje izvede izdajo računa. Ko oddamo naročilo, osebje pri blagajni natančno ve, kdo je oddal naročilo in posledično tudi pri kateri mizi je bilo oddano. Ko smo se odločili, da bomo plačali naročilo, imamo možnost izbire med tremi načini plačevanja. Prvi način plačila je plačilo z gotovino. Ko potrdimo ta način, osebje pri blagajni dobi potrebne podatke o naročilu, kot tudi končno ceno naročenega. Tako imajo vse potrebne podatke, da nam prinesejo naročilo. Imamo tudi možnost, da blagajni sporočimo ob katerem naročilu bomo plačali, tako da natakarju ni potrebno ob vsakem naročilu zraven prinesiti posebej račun.

Naslednja možnost plačila je plačilo s plačilno kartico. Enako kot pri plačilu z gotovino, ob izvedbi naročila aplikacija na blagajno sporoči, da želimo plačati s plačilno kartico. Ob izbranem naročilu natakar s seboj prinese POS terminal, s katerim izvedemo plačilo s kartico.

Zadnja možnost plačila je plačilo z mobilnim telefonom, seveda če restavracija nudi to storitev. Enako kot pri plačilu s plačilno kartico natakar s seboj prinese POS terminal, ki podpira

plačevanje z mobilnim telefonom. V Sloveniji je najbolj razširjen način plačevanja z mobilnim telefonom Moneta. Ta omogoča, da preko POS terminala plačamo s svojim telefonom. Težava je v tem, da lahko Moneto uporabljajo samo uporabniki, ki so naročniki Mobitela, Debitela in Simobila. Kmalu se bo tudi v Sloveniji razvila tehnologija plačevanja z NFC, kot je to primer v Ameriki in Veliki Britaniji. V tem primeru izvedemo plačilo z uporabo aplikacije ponudnika, pri katerem imamo odprt svoj račun (na primer Google Wallet) in modula NFC mobilnega telefona.[3]

4 Platforma Android

Preden smo začeli z izdelavo aplikacije, smo morali določiti, na kateri platformi bo delovala. Odločitev ni bila težka, saj je **Android** po statistiki Statista[14] leta 2015 pokrival 81.2% uporabnikov.

4.1 Kaj je platforma Android

Platforma Android je operacijski sistem za pametne mobilne telefone ter ostale prenosne naprave, ki temelji na Linux jedru. Razvija ga Google v sodelovanju s podjetjem združenja Open Handset Alliance (OHA). Android je v prvi vrsti namenjen mobilnim telefonom, začena se tudi opremljanje drugih naprav, kot so pametne ure, za avtomobile, televizije itd. Večina ljudi, ki so že slišali za platformo Android, zmotno misli, da je platforma Android namenjena zgolj zahtevnejšim uporabnikom. Vendar temu ni tako. Platforma Android je namenjena tudi uporabnikom, ki nimajo veliko predznanja in izkušenj z napravami, kot so npr. namizni/prenosni računalniki, mobilni telefoni itd., ki omogočajo uporabo aplikacij preko različnih kompleksnih interaktivnih programskih vmesnikov.[4]

4.2. Zgodovina

Android Inc. so ustanovili Andy Rubin, Rich Miner, Nick Sears in Cris Whiteeta, leta 2003 v Kaliforniji, v mestu Palo Alto. Leta 2005 je podjetje prevzelo podjetje Google, ki je želelo vstopiti v mobilni trg. Takrat so pričeli z razvojem operacijskega sistema za mobilne naprave, ki bi naj temeljil na Linux jedru.[5]

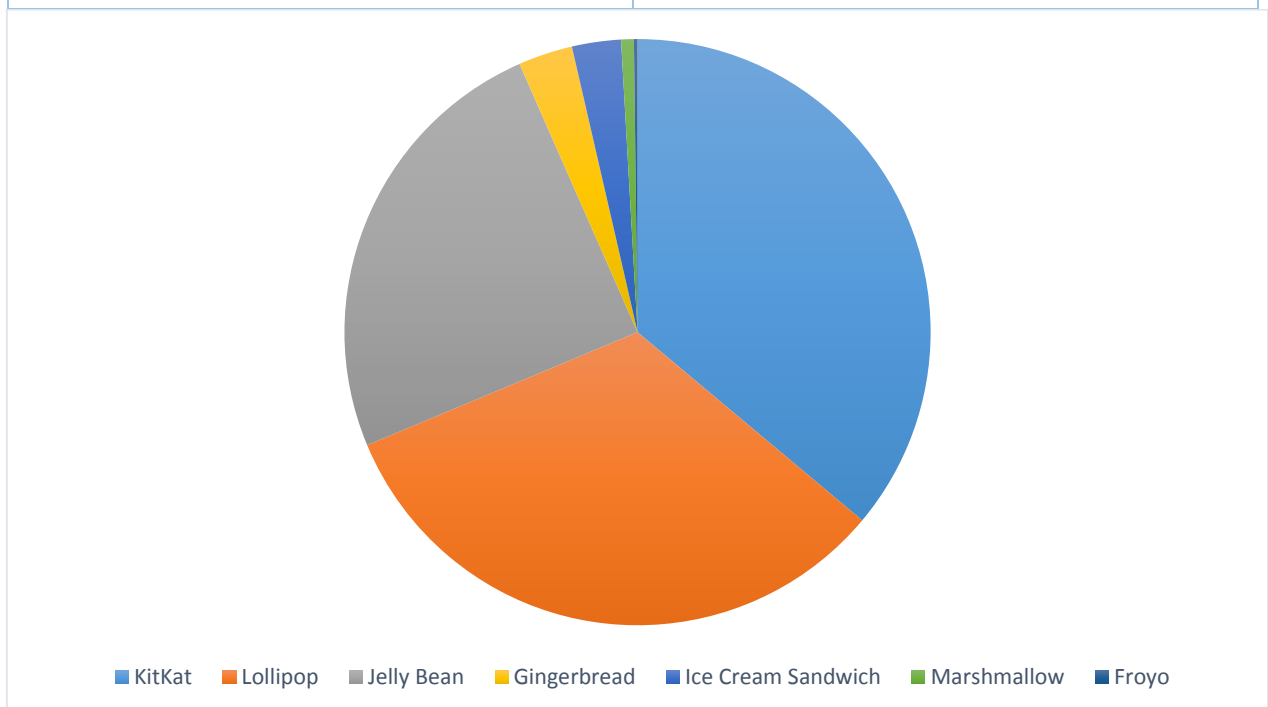
Prihod odprtokodne mobilne platforme Android so objavili novembra 2007 z ustanovitvijo združenja Open Handset Alliance (OHA). V to združenje so danes vključena velika podjetja s področja proizvodnje računalniških naprav, programske opreme, mobilni operaterji in ponudniki mobilnih telefonov. Leta 2007 je sledila tudi prva izdaja programskega orodja. Avgusta 2008 je bil napovedan trg aplikacij Android Market, na katerem je danes vse več aplikacij in iger.[7]

Platforma Android je od nastanka pa do danes doživela veliko nadgradnji, vsaka novejša različica prinaša večjo podporo širokemu naboru strojne opreme in številne izboljšave, ki so podrobneje prikazane v spodnji tabeli 1, odstotna uporaba pa je prikazana s tortnim grafikonom na sliki 5.[5]

Tabela 1: Različice Android OS in njihovi dodatki/popravki

VERZIJA	DODATKI/POPRAVKI
Android 1.6 – Donut	Android Market omogoči lažje iskanje, pretvorba besedila v govor, podporo za zaslon WVGA resolucije.
Android 2.1 - Eclair	E-pošta (več računov hkrati), Bluetooth 2.1, SMS, fotoaparati (digitalni zoom, flash etc.), virtualna tipkovnica, brskalnik (HTML5), Google Maps 3.1.2.
Android 2.2 - Froyo	Hitrost, spomin, integriranje Chrome z JavaScript, izboljšana podpora Microsoft Exchange, USB povezovanje, preklapljanje med jeziki, klicanje ob podpori Bluetootha, Adobe Flash, namestitve aplikacij na pomnilniško kartico, WiFi hotspot, Android Market samostojno posodabljanje aplikacij.
Android 2.3 - Gingerbread	Podpora za veliko ločljivost (WXGA), SIP VoIP, izboljšana virtualna tipkovnica, kopiranje in lepljenje besed, Bass Boost, upravljanje porabe baterije, Download Manager, video format VP8. Dodana nova senzorja giroroskop in barometer.
Android 3.0 - Honeycomb	Optimiziran uporabniški vmesnik, hiter dostop do obvestil, večopravnost, poenostavljeno kopiranje in lepljenje, fotoaparati (fokus, flash, zoom, sprednja kamera), galerija v celozaslonskem načinu, videoklepet Google Talk, izboljšana strojna oprema, podpora za večjedrne procesorje.
Android 4.0 – Ice Cream Sandwich	Obilica popravkov in dodatnih funkcionalnosti sistema; izstopajo strojno pospešeni grafični vmesnik, prenova grafičnega vmesnika, odklepanje s prepoznavo obraza, novi spletni brskalnik, izboljšana aplikacija za kamero, nadzor uporabe podatkov itd.
	Omejen dostop do aplikacij – starševski nadzor, Bluetooth smart support, avtomatsko predlaganje

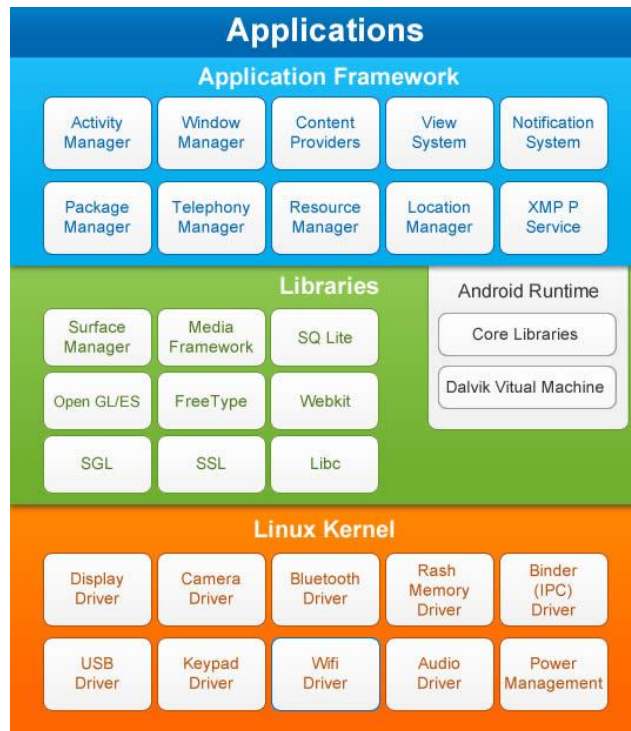
Android 4.1 – Jelly Bean	pri pisanju, Support za Hebrejski in arabski jezik, izboljšan zvok.
Android 4.4 - KitKat	Glasovno upravljanje »Ok Google«, Celozaslonski način dela, pametni imenik, izboljšana optimizacija podatkov in hitrost sistema, ID klicanje, nov način SMS, Dodani Emoji, povezava med printerji, Quickoffice.
Android 5.0 - Lollipop	Novi Material Design, upravljanje na različnih napravah, obvestila na zaklenjenem zaslonu, izboljšan način uporabe baterije, dodatni jeziki, Tap & play itd.



Slika 5: Najbolj uporabljene različice Android OS (1.4.2015)

4.3 Arhitektura platforme Android

Odprtokodno platformo Android v grobem sestavlja pet slojev (slika 6). Vsak sloj lahko uporablja storitve, ki jih ponuja sloj pod njim.[8]



Slika 6: Arhitektura operacijskega sistema Android_[15]

Celotni operacijski sistem Android je zgrajen na podlagi Linux jedra, ki predstavlja osnovni sloj z nekaterimi dodatnimi arhitekturami spremembami, ki jih je naredil Google. Jedro deluje kot vmesna plast med strojno opremo in programskim delom. Med pomembnejše osnovne funkcije, ki jih Android uporablja z jedrom, spadajo upravljanje s pomnilnikom, upravljanje procesov, mrežna podpora, gonilniki in varnost.[8]

Naslednji sloj so knjižnice, ki omogočajo obdelavo različnih vrst podatkov. Knjižnice so napisane v programskem jeziku C/C++ in so specifične za določeno strojno opremo. Med knjižnico spadajo Surface Manager, Media framework, SQLite, SGL, SSL. V sloju knjižnic je tudi zagonsko okolje.[8]

Glavni del zagonskega okolja predstavlja ART, ki izboljšuje splošno učinkovitost izvajanja in s tem zmanjšuje porabo energije, prinaša pa hitrejšo izvajanje aplikacij ter izboljšavo delovanja pomnilnika.[8]

Naslednji sloj je aplikacijsko ogrodje, ki omogoča razvijalcem, da bodo razvite aplikacije prikazane enako na vseh mobilnih napravah s podporo operacijskega sistema Android. Omogoča tudi dostop do knjižnic iz naslednje ravni. Nekaj blokov, ki sestavljajo aplikacijsko ogrodje:

- **Upravljanje aktivnosti (ang. Activity Manager)** upravlja aktivnost življenjskega cikla posameznih aplikacije in omogoča medsebojno komunikacijo med njimi,
- **Ponudnik vsebin (ang. Content Provider)** aplikacijam omogoča dostop do deljenih podatkov, kot so npr. stiki, in omogoča deljenje podatkov med aplikacijami,

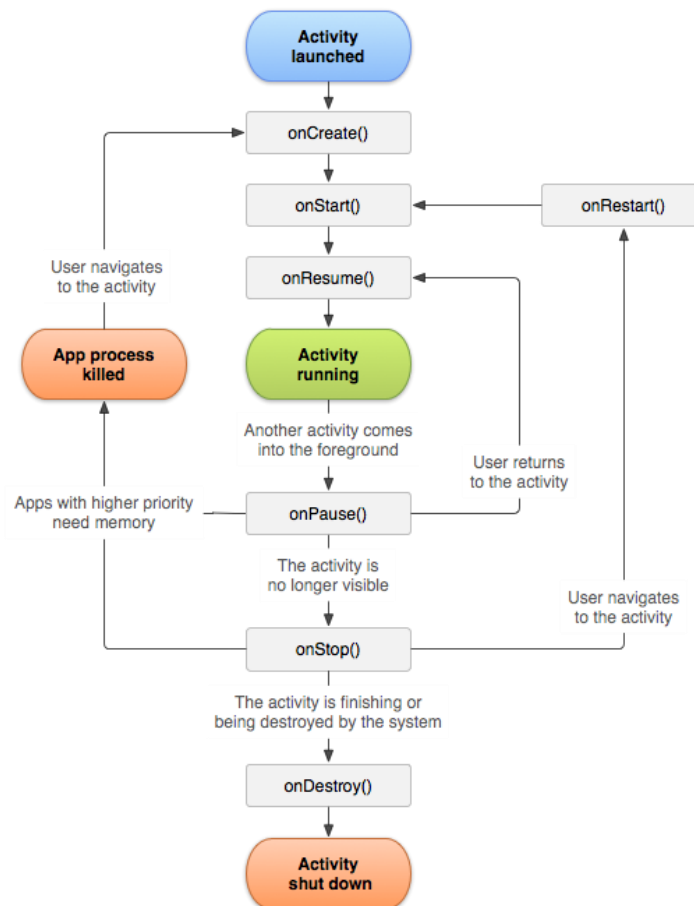
- **Upravitelj telefonije (ang. Telephony Manager)** vsebuje programske vmesnike, namenjene telefoniji,
- **Upravitelj virov (ang. Resource Manager)** zagotavlja dostop do virov, ki niso del aplikacije, kot so lokalni nizi, grafični elementi, razporejevalci vsebin,
- **Upravitelj virov (ang. Notification Manager)** skrbi za prikazovanje obvestil v vrstici stanja sistema,
- **Pogledi (ang. Views)** so osnovni gradniki uporabniškega vmesnika. To so razni gumbi, tekstovna polja, mreže in navsezadnje tudi brskalniki.

Aplikacije so najvišji sloj arhitekture operacijskega sistema Android. Te aplikacije so vključene že v osnovni različici Androida ob nakupu pametnega mobilnega telefona. To so npr. sporočanja, e-pošta, budilka, koledar, stiki, zemljevid in podobno. Vse aplikacije so napisane v programskem jeziku Java.[8]

4.4 Življenjski cikel aktivnosti Android aplikacije

Koncept aktivnost predstavlja posamezni zaslon aplikacije z uporabniškim vmesnikom, na primer seznam prejetih sporočil. Druga aktivnost lahko vsebuje vsebino sporočil, tretja pa na primer izbiro prejemnika posameznega sporočila. Aktivnost ima štiri pomembna stanja življenjske dobe. Ko Med preklopimo na drugo aktivnosti se aktivnost, ki je trenutno delovala, se neha izvajati in jo sistem potisne v zgodovinski sklad, s tem ohrani podatke za kasnejše nadaljnjo izvajanje. Aktivnost lahko iz sklada tudi odstranimo če je več ne potrebujemo ali pa nočemo po ne potrebi polniti pomnilnik. Življenjski cikel aktivnosti je prikazan na sliki 7. Štiri pomembnejša stanja življenjske dobe aktivnosti so:

- **Aktivna aktivnost (ang. Active activity)** aktivnost je prikazana na zaslonu v ospredju in se odziva na uporabnikove akcije (dotik zaslona, pritiske tipk ...),
- **Prekinjena aktivnost (ang. Paused activity)** aktivnost se ne odziva na uporabnikove akcije, vendar je še vedno vsaj delno vidna na zaslonu (izbira opcij, potrjevanje dialogov ...),
- **Ustavljena aktivnost (ang. Stopped activity)** aktivnost na zaslonu ni več vidna, a je še vedno na zgodovinskem skladu,
- **Uničena aktivnost (ang. Destroyed activity)** aktivnost ni na zaslonu in je tudi ni več na zgodovinskem skladu.



Slika7: Življenjski cikel aktivnosti Android aplikacije[16]

4.5 Prednosti platforme Android

Največja prednost platforme Android v primerjavi z ostalimi je v tem, da je odprtokodna in brezplačna. To je dobro tudi za uporabnike, saj imajo na razpolago mobilne naprave s še več funkcionalnostmi in večje število aplikacij, od tega večina brezplačnih. Domači proizvajalci lahko sedaj ceneje in lažje realizirajo svoje ideje in jih ponudijo na trg. Razvijalci lahko koristijo vse možnosti strojnih platform, kot so npr. zaslon na dotik, GPS, kamera, mikrofonski, kompas ipd. Aplikacije lahko vključujejo Java, JavaScript, HTML in CSS. Za to je na voljo tudi brezplačno razvojno okolje SDK (Software development kit). Vse naprave se samodejno sinhronizirajo z Googlovimi storitvami in omogočajo enostavno posodobitev, ko je na voljo novejša različica.[10]

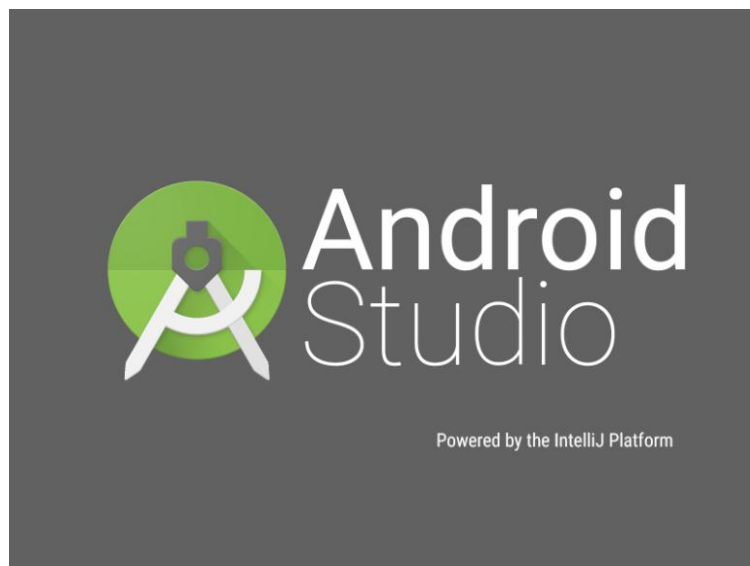
5 Zagonsko in razvojno okolje

Za razvoj Android aplikacij potrebujemo zagonsko in razvojno okolje. Med razvojna okolja spadata tudi Eclipse, z vtičnikom ADT in Android Studio. Za testiranje aplikacij lahko

uporabljamo navidezno napravo, ki je vključena v programskem paketu Android SDK, ali pa uporabljamo kar dejanske naprave z operacijskim sistemom Android.

5.1 Razvojno okolje Android Studio

Razvojno okolje Android studio (slika 8) je odprtokodni in brezplačni program, ki ga je ustvarilo podjetje Google. Namenjen je razvoju aplikacij na platformi Android. Prvič je bil predstavljen 13.5.2013 na Google I/O konferenci. Temelji na programski opremi JetBrains IntelliJ IDEA in je namenjen samo za razvoju Android aplikacij. Prva stabilna različica je bila »Android Studio v1.0«, ki je bila izdana v decembru 2014, trenutna različica pa je »Android studio v2.0 Preview 5« in je bila izdana 14 januarja 2016. Razvojno okolje Android Studio je na voljo za Windows, Linux in Mac OS X.[11]



Slika 8: Logotip Android Studio[17]

5.2 Android SDK

Android SDK je programski paket, ki vsebuje celovit paket orodji za razvijalce. Vsebuje izvorno kodo, razvojno orodje, navidezno napravo (Emulator), dokumentacijo, razhroščevalnik, knjižnice, ki so ključnega pomena pri izgradnji Android aplikacij. Knjižnice so napisane v programskem jeziku Java, poganja pa jih navidezni stroj ART. Navidezna naprava (Emulator) omogoča izvajanje aplikacije, ki jo razvijamo na navidezni ciljni napravi in je v veliko pomoč pri testiranju aplikacij. V programskem paketu so na začetku samo osnovna orodja, v nadaljevanju pa moramo dodajati komponente, ki jih potrebujemo pri razvoju kompleksnejših aplikacij. Orodja SDK dodajamo preko managerja SDK, ki ga vključuje program Android Studio. Android SDK vsebuje tudi starejše verzije platforme Android, tako da lahko razvijalci razvijajo aplikacije na starejših platformah.[12]

5.3 Android API

API je ogrodje, ki ima unikatno oznako za določeno verzijo platforme Android. Operacijski sistem razvijalci ves čas razvijajo in posodablajo, prav tako pa tudi ogrodje API. Vsak novejši API je združljiv s predhodnim, vendar le do določene verzije. Razvijalci morajo pri razvoju določiti najnižjo in ciljno številko (slika 9) različice API.[12]

Version	Codename	API
2.2	Froyo	8
2.3.3 - 2.3.7	Gingerbread	10
4.0.3 - 4.0.4	Ice Cream Sandwich	15
4.1.x	Jelly Bean	16
4.2.x		17
4.3		18
4.4	KitKat	19
5.0	Lollipop	21
5.1		22
6.0	Marshmallow	23

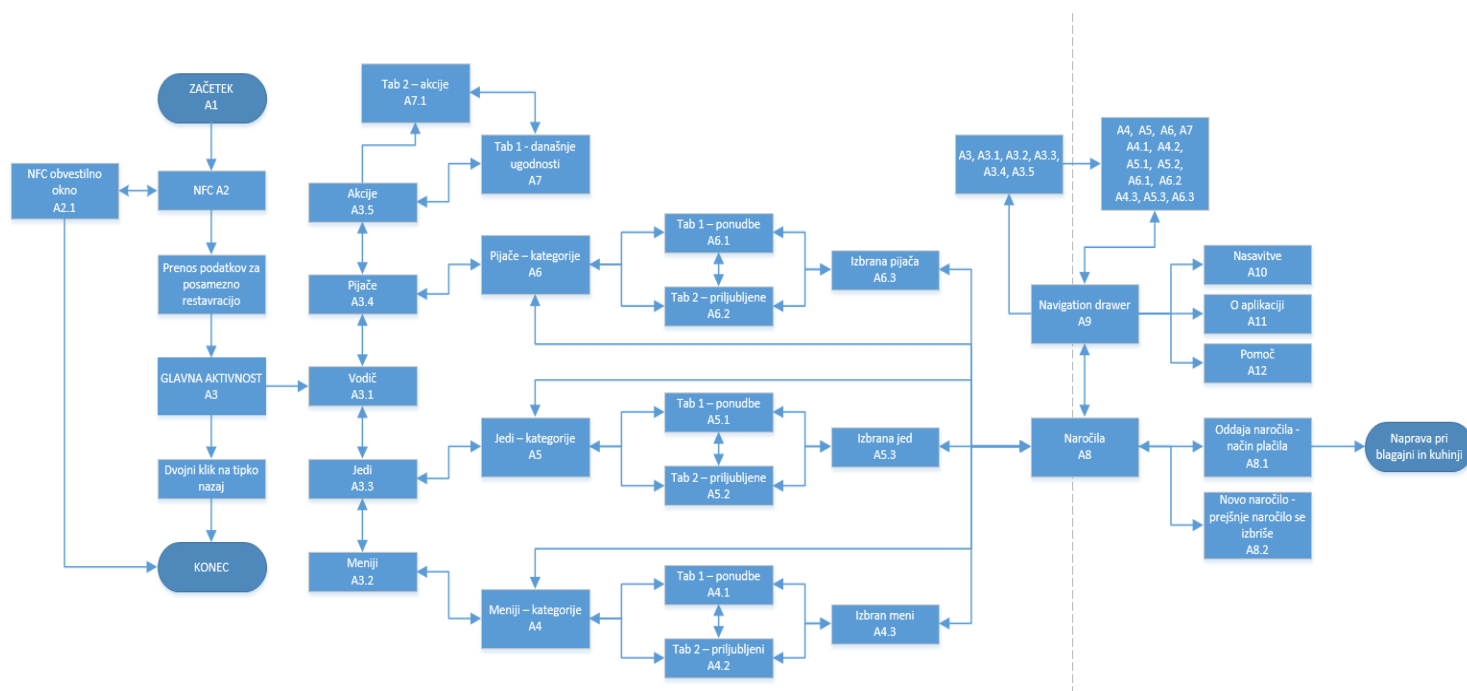
Slika 9: API za različne verzije Androida[18]

5.4 Navidezna mobilna naprava (Emulator)

V paket Android SDK je vključena navidezna mobilna naprava, ki omogoča sprotno testiranje aplikacije in iskanje napak z razhroščevalnikom. Priporočljivo je, da razvijano aplikacijo testiramo na čim več različnih mobilnih napravah, saj je na ta način mogoče, da najdemo napako, ki se morda pojavi smo na nekaterih napravah. Navidezna naprava ima možnost predstavljanja različnih naprav, saj ji lahko določimo velikost in ločljivost zaslona, različico ogrodja API, procesorjev in nastavljanje velikosti delovnega in notranjega pomnilnika. Lahko pa uporabimo tudi dejanske naprave, ki morajo imeti operacijski sistem Android in ki jih priklopimo na računalnik. Izkazalo se je, da je testiranje na dejanskih napravah boljše ter hitrejše, saj navidezne naprave potrebujejo več časa za zagon.[12]

6 Razvoj aplikacije FastOrder

Zaradi problema, ki je podrobneje opisan v poglavju 2, smo se odločili, da razvijemo aplikacijo, ki deluje na pametnih mobilnih napravah z operacijskim sistemom Android. Aplikacijo smo razvijali v razvojnem okolju Android Studio, ki je opisan v poglavju 5. Aplikacija še ni imela nobenih nadgradnji in je trenutno še prototip. Pri razvoju aplikacije smo uporabili dva programska jezika, to sta Java, za funkcionalnost aplikacije in XML, ki zajema grafični vmesnik aplikacije.



Slika 10: Diagram poteka aplikacije FastOrder

6.1 Zagon aplikacije

Ob Zagonu aplikacije, aplikacija najprej preveri internetno povezavo. Če internetne povezave ni, se pokaže obvestilo, ki uporabnika opozori, da nima internetne povezave in ga pozove, naj vključi internetno povezavo ter poskusi ponovno ali pa zapre aplikacijo. Če je internetna poveza vzpostavljena z brezžičnim omrežjem, ki ga ponuja restavracija, v kateri se gost nahaja, se aplikacija normalno zažene. Ko uporabnik prvič zažene aplikacijo, se najprej prenese datoteka s podatki o gostilni, zatem se odpre glavna aktivnost, ki prikazuje prvi fragment – Vodič. Ko gost tudi naslednjič pride v restavracijo se mu podatki po enakem postopku prenesejo in ko restavracijo zapusti se mu izbrišejo, da ne zasedajo dodatnega prostora.

6.2 Glavna aktivnost

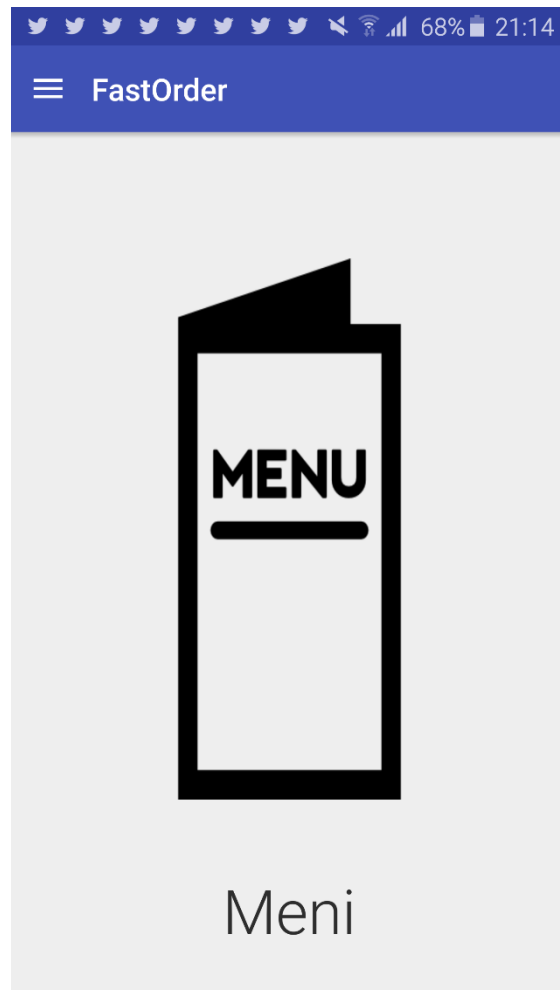
V glavni aktivnosti se nahaja pet fragmentov, katerih funkcionalnosti so opisane v nadaljevanju:

- Fragment Vodič
- Fragment Meniji
- Fragment Jedi
- Fragment Pijače
- Fragment Akcije

Nad vsemi fragmenti smo vstavili akcijsko vrstico. Na njej se nahaja navigacijski predal, preko katerega lahko dostopamo do štirih fragmentov: Meniji, Jedi, Pijače, Akcije. Dostopamo lahko tudi do aktivnosti Naročila, Nastavitvev, Pomoči in Podatkov o aplikaciji. Iz Glavne aktivnosti (kar pomeni tudi za vseh pet fragmentov) se z dvojnimi klikom na tipko nazaj aplikacija zapre.

Prvo okno, ki se pojavi, je fragment vodič. Na njem si lahko preko videoposnetka ogledamo informacije o aplikaciji oziroma način uporabe aplikacije. Če želimo izvedeti več navodil, je na voljo tipka »Več informacij«, ki odpre internetno stran restavracije. Na njej najdemo podrobnejša navodila o uporabi aplikacije. Imamo tudi možnost izklopa Vodiča. V tem primeru se ob naslednjem zagonu aplikacije vodič ne pokaže več. Da ga lahko ponovno uporabimo, ga moramo pod nastavitvami aplikacije spet vklopiti.

V glavni aktivnosti upravljamo z aplikacijo tako, da podrsavamo po zaslonu v desno ali v levo. Če iz vodiča podrsamo v desno, preidemo v novi fragment Meniji (slika 11). Do fragmenta lahko dostopamo tudi preko navigacijskega predala, ki se nahaja v akcijski vrstici. Fragment Meniji na sredini prikazuje veliko ikono, ki ima funkcionalnost na klik. Če pritisnemo na ikono, se ustvari nova aktivnost, ki bo prikazovala seznam Menijev oz. kategorije.



Slika 11: Fragment Meniji v glavni aktivnosti

Če iz fragmenta Meniji podrsnemo na desno, preidemo v naslednje okno oz. fragment Jedi (slika 12). Tudi ta fragment vsebuje ikono, ki jo lahko pritisnemo in deluje kot tipka. Ob pritisku na ikono se ustvari nova aktivnost, ki prikazuje kategorije jedi.



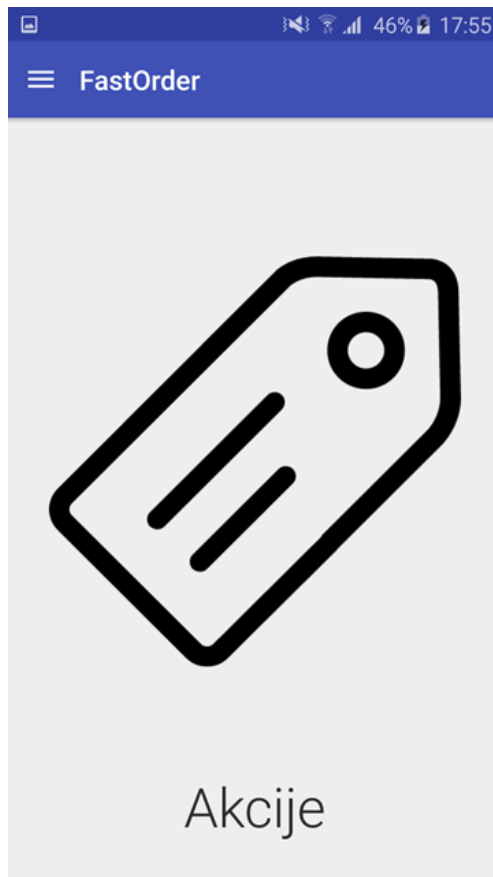
Slika 12: Fragment Jedi v glavni aktivnosti

Naslednji fragment Pijače (slika 13), ki je zgrajen na enakem principu, ima na sredini zaslona ikono, ki je tudi tipka, s katero preidemo v naslednjo aktivnost, ki jo bomo predstavili kasneje. V njej se nahaja seznam kategorij o jedeh. Vsebuje tudi navigacijski predal. Ta fragment se, enako kot vsi ostali, nahaja v isti aktivnosti – Glavna aktivnost.



Slika 13: Fragment Pijače v glavni aktivnosti

Zadnji je Fragment Akcije (slika 14), ki se nahaja desno od fragmenta Pijače. Je popolnoma enak, kot prej opisani fragmenti. Razlika je le v tem, da ima drugačno ikono oziroma tipko, ki pa ne odpre nove aktivnosti z seznamom, ampak novo aktivnost, ki vsebuje dva zavihka. Oba zavihka bomo podrobneje predstavili pozneje.



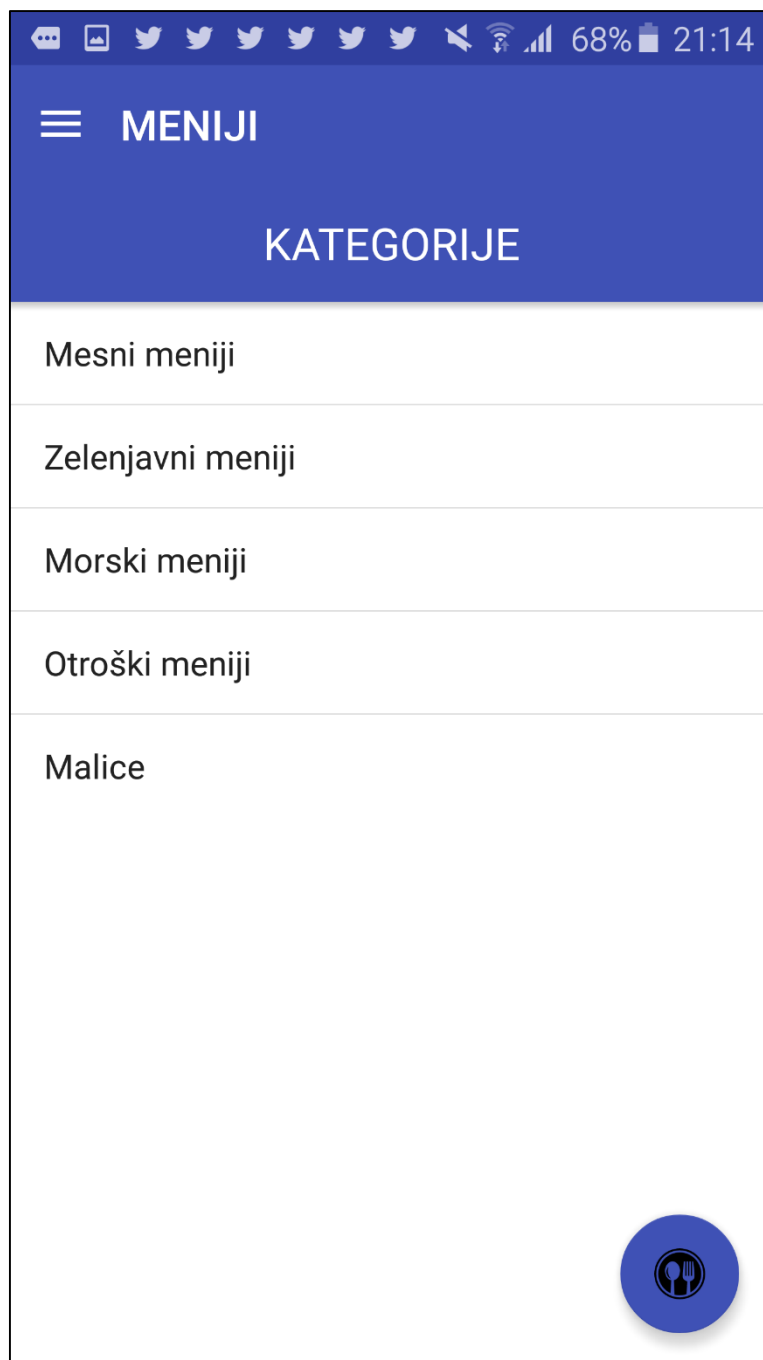
Slika 14: Fragment Akcije v glavni aktivnosti

6.3 Naslednje aktivnosti

6.3.1 Aktivnost Meniji - izbrana

Ko pri fragmentu Meniji (ki se nahaja v glavni aktivnosti) pritisnemo na ikono oziroma gumb, se s pomočjo **Intenta** ustvari nova aktivnost »MENIJI_IZBRANO« (slika 15). V njej se nahaja seznam kategorij, ki smo jih z adapterjem vnesli v ta seznam. Podatke oziroma kategorije je aplikacija prebrala ali prenesla iz podatkovne baze na spletu, vsaka restavracija ima svojo podatkovno bazo. Kategorije so lahko npr.

- Mesni meniji
- Zelenjavni meniji
- Morski meniji
- Otroški meniji
- Malice



Slika 15: Aktivnost s seznamom kategorij menijev

Vsaka posamezna kategorija omogoča, da lahko na njo pritisnemo in posledično sprožimo novo aktivnost, v kateri sta dva zavihka. Prvi zavihek »PONUDEBE« ima svoj fragment, v katerem je seznam ponudb, ki jih restavracija ponuja. Na vsakem seznamu je prikazana slika določenega menija, njegova cena in krajši opis. Ko uporabnik pritisne na enega izmed menijev, se ustvari nova aktivnost »IZBRANA PONUDBA«. Drugi zavihek »PRILJUBLJENI« vsebuje seznam ponudb, ki so bile največkrat izbrane. Seznam je zgrajen enako, kot prejšnji. Tudi tukaj lahko izberemo posamezni meni in ustvarila se bo nova aktivnost »IZBRANA PONUDBA«. V aktivnosti je tudi akcijska vrstica, ki omogoča dostop do navigacijskega predala. Zavihka se

med seboj povezujeta z vmesnikom za premikanje po zaslonu, ko podrsnemo v desno preidemo v naslednji zavihek, lahko pa tudi tako, da pritisnemo na zavihek, ki ga želimo izbrati.

Na obeh zavihkih je prisoten tudi FAB (Floating Action Button), ki ga lahko kadarkoli pritisnemo in ustvari se nova aktivnost »NAROČILA«, ki jo bomo podrobneje predstavili pozneje.

Aktivnost »IZBRANA PONUDBA« podrobneje prikazuje posamični meni, ki je bil izbran. Aktivnost vsebuje slike, ki uporabniku prikažejo izgled menija, levo pod sliko je gumb, ki omogoči, da dodamo ta meni v aktivnost »Naročila«. Tudi to aktivnost bomo predstavili kasneje. Desno od gumba je podana cena menija, pod gumbom je podrobnejši opis za izbrani meni, ki prikazuje sestavine ipd. Pod opisom lahko meni ocenimo z zvezdicami od 1 do 5 in pod ceno si lahko preberemo komentarje, ki so jih napisali gostje.

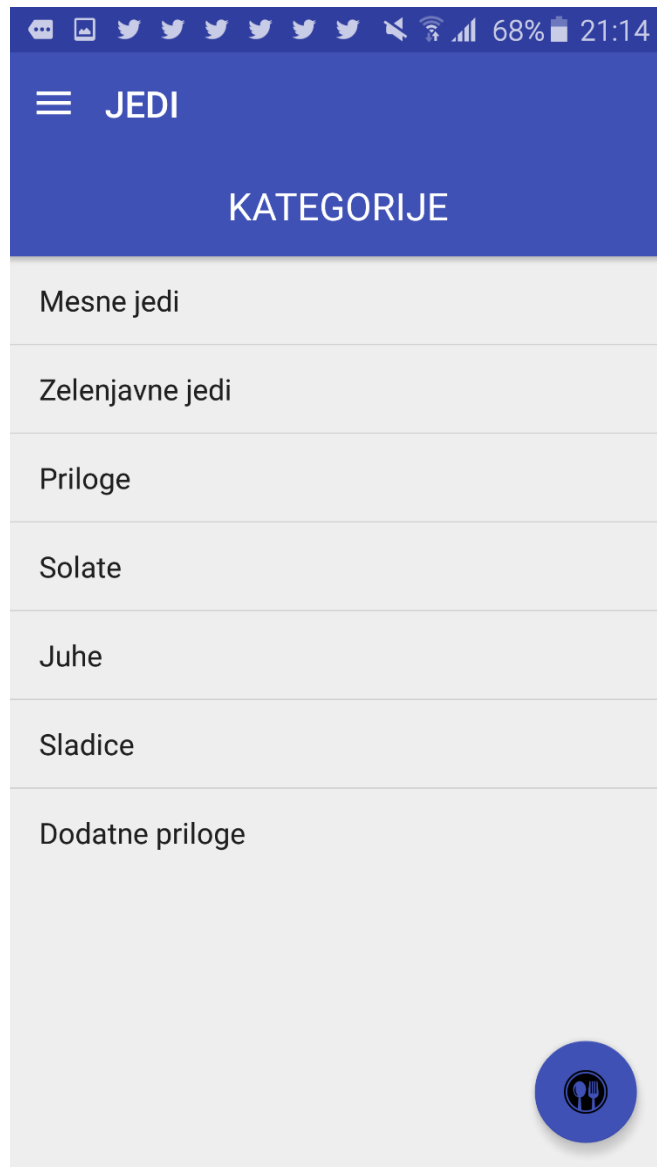
6.3.2 Aktivnost Jedi – izbrana

Do aktivnosti Jedi dostopamo enako, kot pri prejšnji aktivnosti. Ko pritisnemo na gumb v fragmentu Jedi, ustvarimo novo aktivnost »JEDI_IZBRANO« (slika 16). Ta aktivnost prikazuje seznam kategorij za jedi, ki jih izbiramo. Seznam je sestavljen enako, kot pri aktivnosti Menij izbran. Enako tudi podatki, kot na primer:

- Mesne jedi
- Zelenjavne jedi
- Priloge
- Solate
- Juhe
- Sladice
- Dodatne priloge

Vsaka posamezna kategorija omogoča, da lahko na njo pritisnemo, kar pa sproži novo aktivnost, v kateri se na zaslonu prikažeta dva zavihka. Prvi zavihek »PONUDBE« ima svoj fragment, v katerem je seznam ponudb, ki jih gostilna ponuja. Na vsakem seznamu je prikazana slika določene jedi, v vsakem seznamu je tudi prikazana cena posamezne jedi. Tukaj ni opisanih sestavin oz. hranilnih vrednosti, prikazane so v aktivnosti izbrana ponudba. Ko pritisnemo na eno izmed jedi, se ustvari nova aktivnost »IZBRANA PONUDBA«. Drugi zavihek »PRILJUBLJENI« vsebuje seznam ponudb, ki so bile največkrat izbrane. Seznam je zgrajen enako, kot prejšnji. Tudi tukaj lahko posamezne jedi izberemo in ustvarila se bo aktivnost »IZBRANA PONUDBA«. Prisoten je tudi FAB.

Aktivnost »IZBRANA PONUDBA« podrobneje prikazuje posamično jed, ki je bila izbrana. Aktivnost vsebuje slike, levo pod sliko je gumb za dodajanje te jedi v aktivnost »Naročila«, desno od gumba je podana cena jedi, pod gumbom je podrobnejši opis hranilnih vrednosti za izbrano jed, pod opisom lahko jed ocenimo in pod ceno si lahko preberemo komentarje. Tu so tako spremenjeni le podatki, aktivnost je enaka.



Slika 16: Aktivnost s seznamom kategorij jedi

6.3.3 Aktivnost Pijače – izbrana

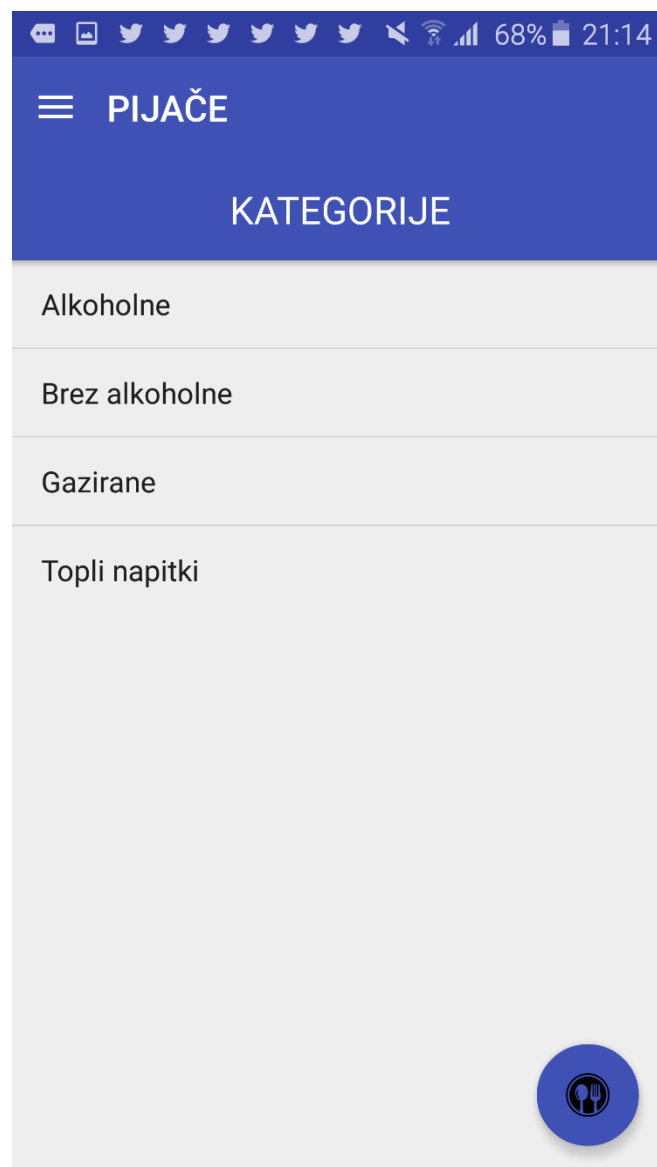
Enako je zgrajena tudi aktivnost »PIJAČE_IZBRANO« (slika 17). Do nje pridemo, če pritisnemo na ikono oz. gumb v fragmentu Pijače v glavni aktivnosti. Tudi ta aktivnost vsebuje seznam kategorij pijač, odvisno pa je od tega, kaj restavracija ponuja. Tako je lahko seznam na primer naslednji:

- Alkoholne
- Brez alkoholne
- Gazirane
- Topli napitki

Vsaka posamezna kategorija omogoča, da lahko na njo pritisnemo, kar pa sproži novo aktivnost, v kateri sta dva zavihka. Prvi zavihek »PONUDBE« ima svoj fragment, v katerem je

seznam ponudb, ki jih restavracija ponuja. Na vsakem seznamu je prikazana slika določene pijače in cena. Tu ni prikazanih sestavin oziroma hranilnih vrednosti. Te so prikazane v aktivnosti izbrana ponudba. Ko pritisnemo na eno izmed pijač, se ustvari nova aktivnost »IZBRANA PONUDBA«. Drugi zavihek »PRILJUBLJENI« vsebuje seznam ponudb, ki so bile največkrat izbrane. Seznam je zgrajen enako, kot seznam IZBRANE PONUDBE. Tudi tukaj lahko posamezne pijače izberemo in ustvarila se bo aktivnost »IZBRANA PONUDBA«. Prisoten je tudi FAB.

Aktivnost »IZBRANA PONUDBA« podrobneje predstavi posamično pijačo, ki je bila izbrana. Aktivnost vsebuje slike, levo pod sliko je gumb za dodajanje te pijače v aktivnost »Naročila«, desno od gumba je podana cena pijače, pod gumbom je podrobnejši opis hranilnih vrednosti za izbrano pijačo, pod opisom lahko pijačo ocenimo in pod ceno si lahko preberemo komentarje. Tudi v tem primeru se spremenijo le podatki, aktivnost je enaka.



Slika 17: Aktivnost s seznamom kategorij pijač

6.3.4 Aktivnost Akcije - izbrana

Ko pri fragmentu Akcije kliknemo na ikono oziroma gumb, se ustvari nova aktivnost »AKCIJE_IZBRANO«. Ta aktivnost vsebuje dva zavihka, oba z enim fragmentom. Prvi zavihek »ZNIŽANE PONUDBE« prikazuje seznam menijev, jedi in pijač, ki so dnevno znižani. Posamezna ponudba je v seznamu ilustrirana s sliko ponudbe, če je to meni, je prikazana slika menija. Enako je predstavitev izvedena za posamezne jedi oziroma pijače. Posamezna ponudba na seznamu deluje tudi kot gumb. Če izberemo ponudbo, se bo ustvarila aktivnost »IZBRANA PONUDBA«. V njej je so enak način, kot pri prejšnjih primerih predstavljeni meniji, jedi ali pijače. Drugi zavihek »AKCIJE« prikazuje dialog opisa akcij, ki jih gostilna ponuja, npr. če naročimo 10 naročil, je naslednja zastoj.

6.4 Aktivnost Naročila

Aktivnost Naročila se ustvari vedno, ko aktiviramo gumb FAB, ki se nahaja v aktivnostih Meniji_izbrano, Jedi_izbrano, Pijače_izbrano in aktivnostih znotraj njih. Nahaja se tudi v aktivnosti Akcije_izbrano. V tej aktivnosti se prikazujejo ponudbe, ki smo jih izbrali preko gumba »Dodaj v naročila«. Ta gumb se nahaja v aktivnosti IZBRANA PONUDBA. V aktivnosti Naročila se naročila prikazujejo kot seznam. Na levi strani posameznega seznama je ime naročila, na desni strani pa se nahaja cena. Na koncu vsakega posameznega seznama, desno od cene se nahaja gumb v obliki križca, ki na potrditev izbriše določen seznam, tako da če smo si premislili glede posamezne izbire, jo lahko enostavno izbrišemo in dodamo novo. Pod seznamom se na desni strani nahaja gumb Oddaj naročilo. Ob pritisku na gumb se pojavi obvestilo, s pomočjo katerega izberemo način plačila. Na voljo so tri možnosti. Prva je »Gotovina«. S to funkcijo natakarju sporočimo, da smo izbrali možnost plačila – gotovina in naj ob prinesenem naročilu prinese tudi račun. Druga možnost plačila je »Plačilo s kartico«. Ob izbiri te funkcije bo natakar vedel, da mora ob naročilu prinesiti s seboj tudi brezžični POS terminal, preko katerega bomo s plačilno kartico plačali naročilo. Zadnji način je plačilo preko mobilnega telefona. Natakar bo vedel, da bomo plačali s telefonom. Odvisno od uporabljanega sistema mobilnega plačevanja bo potekal tudi sam način plačila. Desno od gumba je prikazana končna cena vseh naročil, ki smo jih dodali na seznam. Pod gumbom za oddajo naročila je gumb za ponovno naročilo. Ko ga aktiviramo, se seznam z naročili izbriše in imamo možnost izbire novega. Za izhod iz te aktivnosti pritisnemo tipko za nazaj in vrnili se bomo tja, kjer smo nazadnje pritisnili na FAB.

6.5 Aktivnosti v navigacijskem predalu

Ko se nahajamo v navigacijskem predalu (slika 18), imamo možnost izbire naslednjih aktivnosti:

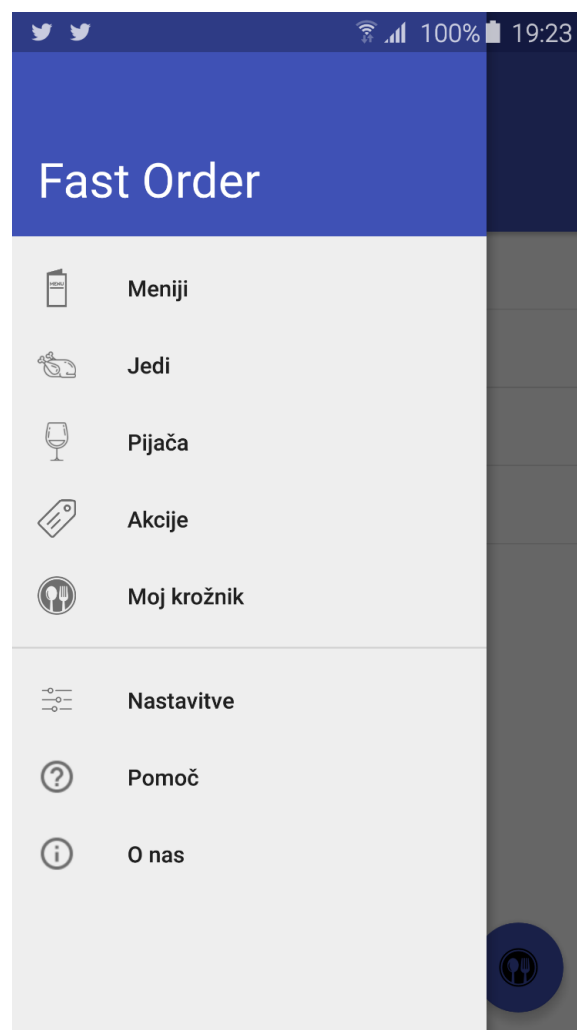
- Meniji

- Jedi
- Pijače
- Akcije
- Moj krožnik (aktivnost naročila)

Pod njimi pa imamo še možnost izbire aktivnosti:

- Nastavitve
- Pomoč
- O nas

Aktivnost »Nastavitve« omogoča nastavljanje različnih nastavitvev v aplikaciji, kot je jezik, barva aplikacije, izklop oz. vklop fragmenta za pomoč. S klikom na tipko nazaj, se vrnemo v aktivnost, v kateri smo aktivirali nastavitve. Aktivnost »Pomoč« prikazuje dialog, v katerem imamo podrobneje opisana navodila za uporabo aplikacije, pa tudi priloženo povezavo za internetno stran restavracije, na kateri so navodila še boljše opisana. Na spletnih straneh lahko dobimo tudi več informacij o gostilni. Zadnja aktivnost »O nas« prikazuje dialog, ki opisuje proizvajalca aplikacije.



Slika 18: Navigacijski predalnik

7 Sklep

Na začetku raziskovalne naloge smo najprej predstavili funkcionalnost aplikacije in njeno delovanje od povezave mobilnega telefona s strežnikom, do plačila naročene hrane in pijače. Predstavili smo tudi tehnologije, potrebne za razvoj zastavljene aplikacije. V nadaljevanju smo podrobno predstavili mobilno platformo Android ter prikazali trenutno najbolj uporabljene različice operacijskega sistema Android. Ugotovili smo, da je najbolj uporabna različica 4.4 KitKat. Nato smo razčlenili sloje arhitekture operacijskega sistema in prikazali stanja življenjskega cikla aktivnosti aplikacije Android.

V nadaljevanju smo predstavili uporabljeno prosto dostopno razvojno okolje za razvijanje aplikacij na mobilni platformi Android. To okolje je Android Studio. Poleg tega smo predstavili še dodatke, brez katerih v programskem jeziku Java ne moremo programirati Android aplikacij.

V praktičnem delu raziskovalne naloge smo se osredotočili na razvoj mobilne aplikacije FastOrder, ki smo jo razvili za mobilne naprave z nameščenim operacijskim sistemom Android. Aplikacija je namenjena vsem, ki si želijo hitrega in zanimivega naročanja hrane in pijače, kot tudi za gostince, ki si želijo bolj tehnološko napredne ponudbe. Najprej smo predstavili načrt funkcionalne zasnove aplikacije, nato pa podrobneje opisali pomembne funkcionalnosti. Na koncu smo izvedli preizkušanje aplikacije na fizičnih napravah, kar je pripomoglo k uspešnejšemu ugotavljanju napak v sistemu. Skozi celoten proces razvoja so bili v pomoč prosto dostopni primeri na spletu.

Viri

[1] NFC tehnologija [online]. Slo - Android.si. 2013. [7.2.2016]. Dostopno na: <http://slo-android.si/forum/viewtopic.php?f=55&t=10242>

[2] Računalništvo v oblaku [online]. Wikipedija. 2015. [7.2.2016]. Dostopno na: https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1tvo_v_oblaku

[3] Kaj je moneta? [online]. Moneta. [s. d.]. [7.2.2016]. Dostopno na: <https://www.moneta.si/predstavitev/>

[4] Platforma [online]. FRI - Android Wiki. 2012. [11.1.2016]. Dostopno na: <http://android.fri.uni-lj.si/index.php/Platforma>

[5] Android (operating system) [online]. Wikipedija. 2016. [11.1.2016]. Dostopno na: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))

[6] Dashboards [online]. Android Developers. 2016. [11.1.2016]. Dostopno na: <http://developer.android.com/about/dashboards/index.html>

- [7] Open Handset Alliance [online]. Wikipedija. 2016. [11.1.2016]. Dostopno na: https://en.wikipedia.org/wiki/Open_Handset_Alliance
- [8] Android architecture [online]. Tutorials point. 2016. [14.1.2016]. Dostopno na: http://www.tutorialspoint.com/android/android_architecture.htm
- [9] Activities [online]. Android Developers. 2016. [14.1.2016]. Dostopno na: <http://developer.android.com/guide/components/activities.html>
- [10] Kaj je Android? [online]. Slo – Android.si. 2011. [15.1.2016]. Dostopno na: <http://slo-android.si/prispevki/kaj-je-android.html>
- [11] Android Studio [online]. Wikipedija. 2016. [15.1.2016]. Dostopno na: https://en.wikipedia.org/wiki/Android_Studio
- [12] Android Software development [online]. Wikipedija. 2016. [15.1.2016]. Dostopno na: https://en.wikipedia.org/wiki/Android_software_development
- [13] Slika. pojmovni diagram računalništva v oblaku[online]. Dostopna na: https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1tvo_v_oblaku#/media/File:Cloud_computing.svg
- [14] The Platform War Is Over and Android Won[online]. STATISTA. 2016. Dostopno na: <https://www.statista.com/chart/4112/smartphone-platform-market-share/>
- [15] Slika. arhitektura operacijskega sistema Android[online]. Dostopna na: https://www.google.si/search?q=Android+-+Architecture&espv=2&biw=1745&bih=859&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiY7761k-7KAhVrIpoKHcYkB3MQ_AUIBigB#imgrc=mCDn3xy3vm3cJM%3A
- [16] Slika. življenjski cikel aktivnosti Android aplikacije[online]. Dostopno na: <http://developer.android.com/guide/components/activities.html>
- [17] Slika. logotip Android Studio[online]. Dostopno na: https://www.google.si/search?q=android+studio+logo&espv=2&biw=1745&bih=859&tbm=isch&imgil=XziquvMF6h-tnM%253A%253B0TA6ebUmqLf32M%253Bhttp%25253A%25252F%25252Fcodecanyon.net%25252Fitem%25252Fblappy-bird-source-code-android-studio-project%25252F11230906&source=iu&pf=m&fir=XziquvMF6h-tnM%253A%252C0TA6ebUmqLf32M%252C &usg=__JN18DAP6HJa3VDGUL5UGBPNQR0Y%3D&ved=0ahUKEwiNmOv6lO7KAhXBApoKHfS5Co4QyjcIMA&ei=wq67Vo2vDMGF6AT086rwCA#imgrc=XziquvMF6h-tnM%3A
- [18] Slika. API za različne verzije Androida[online]. Dostopno na: <http://developer.android.com/about/dashboards/index.html>