



MOBILNA APLIKACIJA ZA PROMOCIJU TURISTIČKIH SADRŽAJA

MOBILE APPLICATION FOR TOURISTIC PROMOTION

Isidora Cvetkov, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTORTEHNIKA I RAČUNARSTVO

Kratak sadržaj – U radu je predstavljena implementacija mobilne aplikacije za promociju turističkih sadržaja. Rad takođe opisuje razvoj Android aplikacije, kao i njemu praktičnu primenu u svrhu turističkih promocija određenih lokacija. Funkcionalnost i način implementacije su detaljno opisani u samom radu.

Ključne reči: *Turizam, Android, Mobilna aplikacija, Google map, SQLite, Java, GPS*

Abstract – *This paper presents the implementation of a mobile application for the promotion of tourist content. This paper also describes the development of the Android application, as well as its practical application for the purpose of tourist promotions of certain locations. The functionality and implementation method are described in detail in the paper.*

Keywords: *Tourism, Android, Mobile application, Google map, SQLite, Java, GPS*

1. UVOD

Sve bržim razvojem mobilnih uređaja i mobilne aplikacije beleže sve veći broj instalacija za različite potrebe počevši od aplikacija koje služe isključivo za zabavu, zatim aplikacije informisanja, svakodnevnog poslovanja i ostale. Zbog svoje dostupnosti, kao i zbog rasta broja prodatih mobilnih uređaja na kojima se izvršavaju i jednostavnosti korišćenja, mobilne aplikacije beleže značajan rast primene i u različitim područjima turističkog poslovanja. Mobilne aplikacije predstavljaju programsku podršku za pametne telefone, tablet računare i druge mobilne uređaje. Turističke aplikacije imaju više uloga, njihova primena je veoma širokog spektra. Neke od uloga turističkih aplikacija jesu da zamenjuju turističkog vodiča, zatim agencije za rezervaciju smeštaja, karti, iznajmljivanje automobila, takođe mogu biti vodiči za kupovinu, navigacija, prevodilac, službe servisa kao što su vremenska prognoza, red vožnje, kursna lista i još mnogo toga. Svrha turističkih aplikacije jeste da se na najlakši mogući način isplanira putovanje ili obilazak, kako bi se pomoglo korisniku. Takođe je bitno naglasiti da korisnici, prolazeći kroz sve cikluse putovanja, koriste svoje mobilne telefone tako što istražuju, rezervišu, fotografišu i snimaju određene destinacije. Zahvaljujući turističkim aplikacijam, korisnik može lakše pronaći željenu destinaciju, informirati se o željenim sadržajima

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragan Ivetić red.prof.

na toj destinaciji i još dosta toga. Mobilne aplikacije postale su jedan od bitnijih marketinških alata u razvoju i promociji turizma, pa tako imaju promotivnu ulogu u brojnim destinacijama i turističkim preduzećima [1, 2].

Kada je reč o aplikaciji koja je razvijena kao deo ovog rada ona predstavlja turistički vodič i korisnicima omogućava pronalaženje mesta u okolini i dodatne informacije o istim.

Tehnologije koje su korišćene za razvoj ove aplikacije su pre svega Android operativni sistem, jedan od vodećih operativnih sistema na tržištu mobilnih uređaja. Kao razvojno okruženje korišćen je Android Studio i programski jezik Java, korišćene su Google mape, takođe korišćen je i GIMP (engl. GNU Image Manipulation Program) za obradu slika u aplikaciji i izradu logoa aplikacija. Detaljno o tehnologijama i korišćenju aplikacije pisano je u poglavljima u nastavku.

2. ANDROID OPERATIVNI SISTEM

Android je operativni sistem otvorenog koda, zasnovan na Linux na sistemu. Napravljen je za veliki spektar mobilnih uređaja. Android je trenutno najrasprostranjeniji operativni sistem za mobilne telefone, prilagođen je tako da se može koristiti na većini mobilnih uređaja, takođe uključujući pored mobilnih telefona, i tablet računare, laptop računare, smartbook računare, čitače elektronskih knjiga i ručne satove [3][4]. Sam korisnički interfejs Androida je zasnovan na direktnoj manipulaciji objektima na ekranu, korišćenjem ulaza u vidu dodira koji odgovaraju pokretima u realnom svetu kao što su prevlačenje, pritiskanje, zumiranje (engl. Pinching), kao i unos teksta pomoću virtuelne tastature.

2.1. Arhitektura Android operativnog sistema

Najbolji način za razumevanje Android operativnog sistema je prikaz njegovog rasčlanjenja na sekcije i slojeve. Android je grubo podeljen na šest sekcija i pet slojeva [3]. Linux jezgro predstavlja osnovu Android sistema. Ovaj sloj sadrži sve drajvere uređaja definisane na niskom nivou za različite hardverske komponente svakog pojedinačnog Android uređaja. Apstraktni sloj hardvera (engl. Hardware abstraction layer – skr. HAL) – definiše standardni interfejs koji omogućava korišćenje hardverskih komponenti uređaja na višem nivou pomoću Java API (engl. Application Programming Interface) okvira. API je aplikacioni programski interfejs koji definiše načine na koje aplikacije mogu da zahtevaju usluge od operativnog sistema ili biblioteka. Biblioteke sadrže ključne komponente i servise Android sistema, kao što su ART (Android okruženje za izvršavanje) i HAL

(Apstraktni sloj hardvera) napisane u izvornom kodu koji za izvršavanje traže biblioteke napisane u C/C++ programskom jeziku. Operativni sistem Android pruža okvir Java API za korišćenje nativnih biblioteka u aplikacijama. Na primer, biblioteka WebKit obezbeđuje funkcije koje se odnose na pregledanje veb sadržaja ili biblioteka SQLite koja obezbeđuje podršku za korišćenje baza podataka. Android okruženje za izvršavanje (engl. ART) nalazi se na istom nivou kao i biblioteke, obezbeđuje skup osnovnih biblioteka koje programerima omogućavaju da pišu Android aplikacije korišćenjem programskog jezika Java. Radni okvir aplikacija (Java API okvir) predstavlja celokupan skup funkcionalnosti Android sistema preko više API okvira napisanih u Java programskom jeziku. Dok skup API okvira čini gradivne jedinice potrebne za kreiranje aplikacija za Android sistem, pojednostavljivanjem ponovnog korišćenja jezgarnih servisa i sistemskih komponenti.

2.2. Karakteristike Android aplikacija

Android aplikacije uglavno su pisane u programskom jeziku Java. Prevedeni Java kod zajedno sa svim podacima i datotekama resursa potrebnim za aplikaciju se arhivira u datoteku sa ekstenzijom *.apk*. Ova datoteka služi za distribuciju i instaliranje aplikacija na mobilnim uređajima. Datoteku korisnici preuzimaju na svoje uređaje. Kod koji se nalazi u jednoj *.apk* datoteci smatra se jednom aplikacijom [3]. Svaka aplikacija živi nezavisno od drugih, odnosno sistem svaku aplikaciju izvršava u posebnom procesu. Android sistem počinje proces kada bilo koji deo koda aplikacije treba da se izvrši i isključuje proces kada više nije potreban. Svaki proces ima svoju specijalnu Java virtualnu mašinu – ART, pa kod aplikacije radi u izolaciji od kodova svih ostalih aplikacija. Svakoj aplikaciji je dodeljen jedinstveni ID korisnika. Dozvole su postavljene tako da su datoteke aplikacije vidljive jedino samoj aplikaciji, iako postoje načini kako se datoteke mogu izvoziti i u druge aplikacije.

2.3. Osnovne kategorije

Sve Android aplikacije se mogu podeliti na više različitih kategorija. Međutim, većina aplikacija se može svrstati u jednu od sledećih kategorija [3][5]:

- Aktivnost koja se izvršava je u prvom planu (engl. Foreground activity) – aplikacija bez pozadinske aktivnosti. U ovu vrstu aplikacija se ubrajaju igre i mape. Kad su izvan fokusa aplikacije ove vrste se uglavnom privremeno zaustavljaju.
- Pozadinska usluga (engl. Background service) – aplikacije su ograničene interakcije sa korisnikom koje se uglavnom izvršavaju u pozadini. Ovoj vrsti aplikacija pripadaju aplikacije za upravljanje dolaznim pozivima.
- Aktivnosti sa prekidima – aplikacije koje podrazumevaju određeni stepen interakcije sa korisnikom, ali se uglavnom odvijaju u pozadini.

2.4. Java

Java je objektno-orjentisan programski jezik zasnovan na klasama dizajniran sa ciljem da ima što manje zavisnosti od implementacije. Jezik je opšte namene namenjen programerima aplikacija sa ciljem da pišu kod jednom, a pokreću bilo gde (engl. Write once, run anywhere – skr.

WORA), što znači da se kompajlirani Java kod može pokretati na svim platformama koje podržavaju Javu bez potrebe za ponovnom kompilacijom. Java aplikacije se obično kompajliraju u bajt kod koji se može pokrenuti na bilo kojoj Java virtualnoj mašini (skr. JVM) bez obzira na osnovnu računarsku arhitekturu. Pored toga, Java jezik ima sintaksu sličnu jezicima C i C++, dizajniran je tako da bude nezavisan od platforme i sa pojednostavljenim upravljanjem memorijom. Java pripada skupu viših programskih jezika i danas je uz Kotlin najpopularniji jezik za razvoj Android aplikacija [8][9], za razvoj ove aplikacije korišćen je Java programski jezik.

3. KOMPONENTE I STRUKTURA APLIKACIJE

Prilikom razvoja aplikacije korišćene su komponente opisane u nastavku ovog poglavlja, kako bi se omogućio lakši i brži razvoj aplikacije, kao i bolje funkcionalnosti.

3.1. Google Maps API

Google Maps API omogućava korišćenje Google Maps servisa na Android platformi, prikaz mape, rukovanje kamerom, pomeranje, zumiranje mape, postavljanje oznaka, linija, poligona na mapi itd. [7].

Implementacija Google mapa se odvija u nekoliko koraka. Prvi korak je preuzimanje Google Play Service SDK (engl. Software Development Kit), zatim preuzimanje ključa za korišćenje Google Maps API-a, unošenje vrednosti ključa u *AndroidManifest.xml*, deklarisanje mapu kao pogled u odgovarajućem rasporedu I na kraju definisanje aktivnosti koja prikazuje raspored. Kada je reč o tipovima Google Mapa koje se mogu koristiti postoje: *normal* – uobičajna autokarta, *hybrid* – satelitski snimak na kome su vidljivi i putevi, *satellite* – satelitski snimak, *terrain* – topografska karta i *non* – bez pločica. U okviru aplikacije implementirana terrain tj. topografska karta.

3.2. SQLite baza podataka

Android aplikacije mogu da koriste ugrađen sistem za upravljanje bazama podataka – SQLite. Za razliku od većine sistema za upravljanje bazama podataka, SQLite se izvršava u istom procesu kao i aplikacija koja koristi njegove usluge. Obezbeđuje referencijalni integritet i omogućava rad u transakcijama. Za pravljenje, izmenu i otvaranje baze podataka koristi se SQLiteOpenHelper klasa i potrebno je implementirati neke od sledećih metoda:

- void onCreate (SQLiteDatabase database),
- void onOpen (SQLiteDatabase database),
- void onUpgrade (SQLiteDatabase database, int old_ver, int new_ver),
- void onDowngrade (SQLiteDatabase database, int old_ver, int new_ver).

SQLite je relacioni sistem upravljanja bazama podataka (skr. RDBMS – engl. Relational Database Management System) koji se nalazi u C biblioteci. Za razliku od mnogih drugih sistema za upravljanje bazama podataka, SQLite nije mehanizam baze podataka klijent-server. Umesto toga, on je ugrađen u krajnji program. SQLite je kompatibilan sa ACID (atomic, consistent, isolated, durable) i primenjuje većinu SQL standarda, uglavnom

prateći [6]. SQLite je popularan izbor kao ugrađeni softver za baze podataka za lokalno tj. klijentsko skladištenje u aplikacionom softveru kao što su veb pregledači ili mobilne aplikacije. Veže se za mnoge programske jezike.

Glavne značajne karakteristike SQLite baze kao i razlozi njenog korišćenja pri razvoju aplikacije su:

- Transakcije funkcionišu po principu ACID (atomic, consistent, isolated, durable). Podaci otaju postojani pri padu sistema ili nestanku električne energije.
- Nulta konfiguracija nema potrebu za prilagođavanjem i administracijom.
- Podržava većinu SQL-92 standarda.
- Cela baza smestena u jednoj datoteci na disku.
- Podaci iz baze podataka se mogu slobodno deliti između više računara.
- Podržava nekoliko terabajta velike baze i nekoliko gigabajta velike nizove podataka.
- Brža izvedba standardnih operacija od ostalih popularnih baza podataka.

3.3. Android Manifest datoteka

Svaki projekat mora da sadrži `AndroidManifest.xml` datoteku, sa tačno tim nazivom. Ova datoteka pruža osnovne podatke o aplikaciji alatima za prevođenje i kompajliranje, operativnom sistemu i Google Play prodavnicima. Uglavnom se `Android Manifest` datoteka kreira automatski pri kreiranju projekta i nalazi se u korenom direktorijumu. U toku razvoja ova datoteka se može menjati automatski ili po potrebi od strane programera.

3.4. Struktura aplikacije

Android projekat u svojoj strukturi sadrži nekoliko foldera i fajlova, a neki od njih su:

- `src` – koji sadrži `.java` fajlove projekta. U `src` fajlu se nalazi sav kod aplikacije,
- `Android biblioteka` – ovaj folder sadrži jedan `android.jar` fajl, koji sadrži biblioteke klasa koje su potrebne za Android aplikaciju,
- `gen` – sadrži `R.java` fajl. On predstavlja generisani fajl koji referencira sve resurse koji se nalaze u projektu,
- `assets` – ovaj folder sadrži resurse koje aplikacija koristi, neki od njih su HTML fajlovi, baze podataka i ostali,
- `res` – ovaj folder sadrži sve resurse koji se koriste u aplikaciji. Takođe, sadrži nekoliko drugih podfoldera, a to su `drawable`, `layout` i `values`,
- `AndroidManifest.xml` – je manifest fajl Android aplikacije. Ovde su registrovane sve aktivnosti koje aplikacija ima, kao i svi servisi, takođe definisane su permisije koje su potrebne aplikaciji.

3.5. Komponente aplikacije

Za razliku od većine aplikacija drugih operativnim sistemima, aplikacije Android sistema nemaju samo jednu pokretnu tačku. Što konkretno znači da nemaju samo jednu `main()` funkciju [3].

Što se tiče osnovnih komponenti koje čine jednu Android aplikaciju postoji njih četiri, a to su:

- **Aktivnosti** (engl. `Activity`) – predstavlja komponentu aplikacije koja se uglavnom poistovećuje sa jednim konkretnim prozorom aplikacije u kojem korisnik može izvršavati određenu radnju. Aplikacija može da sadržati jednu ili više definisanih aktivnosti, pri čemu je jedna od aktivnosti uvek definisana kao primarna aktivnost. Iako više aktivnosti stvara jedan kompaktni korisnički interfejs treba imati na umu da su one međusobno nezavisne jedna od druge. Svaka aktivnost implementira se kao posebna klasa koja nasleđuje klasu `Activity`, te je i sama odgovorna za čuvanje svog stanja u životnom ciklusu aplikacije.
- **Namera** (engl. `Intent`) – su poruke koje omogućavaju komponentama aplikacije da zahtevaju funkcionalnost od drugih komponenti. Omogućavaju komunikaciju sa komponentama iz iste aplikacije, ali i komponentama drugih aplikacija. Namera je u osnovi pasivna struktura podataka koja sadrži apstraktni opis akcije.
- **Usluga** (engl. `Service`) – predstavlja proces bez vidljive korisničke interakcije. S tim da usluga nema korisnički interfejs, ona nije vezana za sam životni ciklus neke aktivnosti. Uglavnom se izvršava u pozadini za nedefinisani period vremena. Usluge se koriste za ponavljajuće i potencijalno dugačke operacije, kao što su preuzimanje sa interneta, proveru novih podataka, obradu podataka, ažuriranje dobavljača sadržaja i slično. Svaka usluga nasleđuje klasu `Service`.
- **Prijemnik** (engl. `BroadcastReceiver`) – služi kako bi prihvatio poruke koje mogu slati druge aplikacije ili čak sam sistem. Te poruke mogu da idu od sistemskih događaja kao što je uključivanje telefona ili kraj preuzimanja podataka sa interneta.
- **Dobavljač sadržaja** (engl. `Content Providers`) – pomaže aplikaciji prilikom pristupa sopstvenim podacima, podacima drugih aplikacija i prilikom deljenja podataka sa drugim aplikacijama. Oni su standardni interfejs koji povezuje podatke iz jednog procesa sa kodom koji se izvršava u drugom procesu i nasleđuju klasu `ContentProvider`. Svaka Android aplikacija ne mora sadržati sve navedene komponente, isto tako može sadržati i neke druge komponente koje ovde nisu navedene.

4. IMPLEMENTACIJA APLIKACIJE ZA PROMOCIJU TURISTIČKIH SADRŽAJA

U ovom poglavlju biće reči o samoj funkcionalnosti aplikacije, organizaciji i glavnim karakteristikama aplikacije

4.1. Pregled i organizacija aplikacije

Android projekati su organizovani po paketima koji su smešteni unutar jednog glavnog paketa koji uglavnom ima naziv isti kao i sama aplikacija. Međutim, fajlovi koji se ne nalaze unutar njega su skripte i fajlovi koji se koriste za pokretanje aplikacije, kao i druge uvezene biblioteke i moduli.

4.2. Modeli podataka aplikacije

Model podataka korišćen tokom razvijanja aplikacije je zadovoljava funkcionalnosti implementirane aplikacije. Kao što je u predhodnom delu naglašeno aplikacija koristi SQLite ugrađenu bazu podataka za čuvanje podataka o svim turističkim atrakcijama. Model ove baze sadrži sledeće tabele: *Korisnik* (Id, Ime, Prezime, Broj, Grad, Email, Lozinka), *Lokacija* (Id, Grad, Naziv, Adresa, Lokacija, Opis, Slika) i *Slika* (Id, Id_Lokacije, Path, Naziv). Svaka lokacija može imati više slika koje su vezane za nju, dok svaka slika koja se nalazi u bazi može pripadati samo jednoj lokaciji.

4.3. Opis funkcionalnosti aplikacije

Turistička aplikacija namenjena je korisnicima koji žele da bliže istraže mesta u svojoj okolini, kao i da pogledaju šta od turističkih atrakcija mogu posetiti. Pre pokretanja aplikacije neophodno je da korisnik uključi GPS (engl. Global Positioning System) kako bi se osigurao ispravan rad aplikacije, praćenje korisničke lokacije u realnom vremenu i preuzimanje ispravnih informacija. Nakon pokretanja aplikacije korisniku će se prikazati ekran sa prijavom na kom može uneti svoju e-mail adresu i lozinku. Unošenjem ispravnih podataka, email adrese i lozinke korisnik se prijavljuje u aplikaciju i nakon toga mu se na ekranu pojavljuje ekran sa glavnim menijem na kome ima opcije koje aplikacija nudi. Ukoliko korisnik nema kreiran nalog mora se registrovati kako bi pristupio aplikaciji. U slučaju registracije korisnika pojaviće se ekran sa poljima za registraciju koja korisnik mora popuniti. Aplikacija izbacuje validacione poruke o greškama za sva polja, na primer ukoliko korisnik unese e-mail adresu u pogrešnom formatu. Takođe registrovanje korisnika nije moguće ukoliko korisnik nije popunio sva polja, ali i ukoliko neko od polja nije popunjeno validnim podacima, na primer ukoliko korisnik unese e-mail adresu koja nije u validnom formatu registracija neće biti moguća, ukoliko umesto slova u poljima koja su tekstualnog formata unese brojeve. Nakon uspešne registracije korisniku se takođe kao i nakon prijavljivanja otvara prozor početnog ekrana aplikacije. Opcije koje korisnik vidi na svom početnom ekranu nakon prijave su: *mapa* – omogućava prikaz svih turističkih atrakcija koje korisnik može posetiti u svojoj okolini, takođe prikazuje i trenutnu lokaciju na kojoj se korisnik nalazi, *detalji* – prikaz u okviru kog korisnik može videti sve informacije vezane za određenu lokaciju, *kamera* – pokretanje kamere korisnikovog telefona, *lista lokacija* – prikaz liste koja sadrži lokacije koje se nalaze u okolini korisnika, *podešavanja* – opcije sa podešavanjima aplikacije, *korisnički profil* – prikaz detalja svog korisničkog profila i *korisničke slike* – opcija koja otvara galeriju slika koje je korisnik napravio.

5. ZAKLJUČAK

Tržište pametnih mobilnih telefona i tableta je u konstantnom porastu, dok prisustvo konkurencije sve više dovodi do usavršavanja i poboljšanja performansi navedenih uređaja. Virtuelna, augmentovana realnost, veštačka inteligencija, grafički zahtevne video igre, turizam samo su neka od polja gde su mobilni uređaji našli svoju

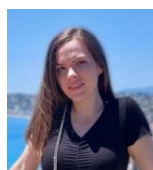
primenu. Raznovrsnost uređaja, svakodnevni napredak pametnih telefona, kao i broj korisnika u velikoj meri uslovljavaju izmene u samom Android operativnom sistemu i procesu razvoja aplikacije. Programerima su na raspolaganju nove metode i mogućnosti za implementaciju u aplikacijama, ali se i povećava potrebno znanje kako bi razvijana aplikacija pratila trendove napretka tehnologije. Korisnički interfejs predstavlja sistem sa kojim korisnici intereaguju sa aplikacijom, predstavlja jako bitan deo pri razvoju aplikacije.

Dobro definisan i funkcionalan korisnički interfejs doprinosi boljoj prihvaćenosti i većem zadovoljstvu između korisnika i aplikacije. Glavni doprinos ovog rada je bliže upoznavanje sa principima, pravilima, konceptima, ali i tehnologijama koje su potrebne za definisanje. Sve ovo je ostvareno kroz upoznavanje sa Android operativnim sistemom, njegovom arhitekturom, osnovnim principima programiranja korisničkog interfejsa, pregledom najnovijih tehnologija, ali i kroz savete i preporuke date u radu. Cilj implementacije Android aplikacije za turiste jeste prikazivanje velikog broja turističkih mesta u korisnikovoj okolini, kao i korisnih informacija koje turistima mogu biti od značaja pri odabiru i obilaženju željenih lokacija, pored toga aplikacija je veoma intuitivna i laka za korišćenje. Implementirana aplikacija se može koristiti i u edukativne svrhe kao primer zbog određenih metoda i obrazaca koji su korišćeni u toku njenog razvoja. Ideja aplikacije se može razvijati u smeru AR vodiča, ali i unapređivanju trenutnih funkcionalnosti iste, kao i povećanja lokacija koje se mogu pratiti.

6. LITERATURA

- [1] <https://aseestant.ceon.rs> (pristupljeno u januaru 2021.)
- [2] D.Buhalis, "Information technology for strategic tourism management", Pearson Publication, 2003
- [3] <https://www.android.com> (pristupljeno u januaru 2021.)
- [4] [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)) (pristupljeno u januaru 2021.)
- [5] <https://developer.android.com/index.html> (pristupljeno u januaru 2021.)
- [6] <https://en.wikipedia.org/wiki/SQLite> (pristupljeno u martu 2021.)
- [7] https://en.wikipedia.org/wiki/Google_Maps (pristupljeno u martu 2021.)
- [8] [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)) (pristupljeno u aprilu 2021.)
- [9] https://en.wikipedia.org/wiki/Android_software_development (pristupljeno u januaru 2021.)

Kratka biografija:



Isidora Cvetkov rođena je u Vranju 1996. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – primenjene računarske nauke i informatika, odbranila je 2022.god.
kontakt: isidora.cvetkov96@gmail.com