SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE



Projekt iz kolegija Analiza i Razvoj Programa

Oznaka tima: T-10

Članovi tima shareSquad:

Jakov Andrić JMBAG 0016079257

Dario Filipaj JMBAG 0016077989

Lorena Tomić JMBAG 0016077809

**Varaždin, 2013/2014**

Sadržaj

[1. Uvod 1](#_Toc380614360)

[2. Odabrana metodika 2](#_Toc380614361)

[2.1. XP planiranje 5](#_Toc380614362)

[2.2. XP dizajniranje 6](#_Toc380614363)

[2.3. XP programiranje 6](#_Toc380614364)

[2.4. XP testiranje 7](#_Toc380614365)

[3. Alati korišteni za realizaciju projekta 8](#_Toc380614366)

[3.1. TFS 8](#_Toc380614367)

[3.1.1. Instalacija 8](#_Toc380614368)

[3.1.2. Korištenje 9](#_Toc380614369)

[3.2. Asana 10](#_Toc380614370)

[3.3. Microsoft Visio 10](#_Toc380614371)

[4. Projektni plan 11](#_Toc380614372)

[4.1. Korisnički zahtjevi 12](#_Toc380614374)

[4.2. Pregled funkcionalnosti besplatne i premium verzije 12](#_Toc380614375)

[4.3. Terminiranje Projekta 14](#_Toc380614376)

[5. Dijagram slučajeva korištenja mobilne aplikacije 15](#_Toc380614377)

[6. Korisnička priča 17](#_Toc380614378)

[7. Dijagram klasa 18](#_Toc380614379)

[8. ERA model 19](#_Toc380614380)

[9. Plan izrade aplikacije 21](#_Toc380614381)

[9.1. Prva faza 21](#_Toc380614382)

[9.1.1. Planiranje releasea 21](#_Toc380614383)

[9.1.2. Planiranje iteracija 22](#_Toc380614384)

[9.2. Druga faza 23](#_Toc380614385)

[9.2.1. Planiranje relesea 23](#_Toc380614386)

[9.2.2. Planiranje iteracije 24](#_Toc380614387)

[9.3. Treća faza 25](#_Toc380614388)

[9.3.1. Planiranje releasea 25](#_Toc380614389)

[9.3.2. Planiranje iteracije 26](#_Toc380614390)

[10. Početna shema i Web aplikacija 27](#_Toc380614391)

[11. Početna shema aplikacije 28](#_Toc380614392)

[Literatura 40](#_Toc380614393)

# Uvod

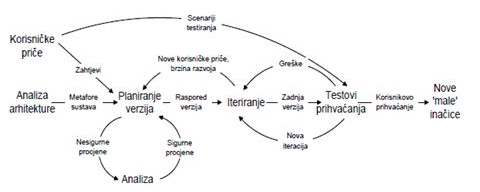
Razmjena kontakt podataka izražen je aspekt na mnogim poslovnim sastancima, konvencijama, pregovorima te mnogim drugim društvenim susretima. Posjedovanje vizitke koja sadrži osobne informacije i kontakt neizbježno je ukoliko se želi ostaviti profesionalni dojam na poslovne partnere, kolege ili suradnike. Razvojem tehnologije, odnosno povećanjem primjene mobilnih uređaja kao pomagala u poslovnom svijetu, razmjena papirnih vizitki doima se zastarjelo.

Aplikacijom shareCard želimo omogućiti kreiranje mobilnih, odnosno digitalnih vizitki koje su personalizirane i interaktivne te ih je jednostavno podijeliti. Svaki korisnik može besplatno kreirati svoju vizitku te ju povezati sa svojom vremenskom crtom na kojoj se dodaju željeni podatci o karijeri i životu pojedinca. Brza razmjena i pretraživanje vizitki omogućuju jednostavno pronalaženje kontakata i pregled svih podatka.

Aplikacijom želimo olakšati razmjenu kontakata, praćenje poslovnih karijera podijeljenih kontakata te pretraživanje podataka o kontaktima. Usredotočit ćemo se na poslovni svijet kao ciljno tržište, ali aplikaciju će moć koristiti svi koji žele svoje kontakt podatke razmjenjivati sa drugima te pohranjivati ih i pretraživati na interaktivan način. Naša vizija je promicanje shareCarda u osnovni i najbolji način razmjene kontakata te povezivanje poslovnih odnosa i veza u interaktivnu mrežu vremenskih crta karijera i poslovnog razvoja.

# Odabrana metodika

Ekstremno programiranje (XP) je agilna metodika razvoja softverskog proizvoda. XP je set najbolje provedenih praksi gdje su neke dovedene na “ekstremnu” razinu. U praksi XP se naginje prema svakodnevnom programskom razvoju. XP se često smatra komplementom metodike Scrum, zatvara pukotine gdje ih Scrum širom otvara. Menadžeri, korisnici i developeri imaju ista prava unutar razvojnog tima. Metodika ekstremnog programiranja implementira jednostavnu, ali učinkovitu okolinu omogućavajući  visoku efikasnost tima. Tim se sam organizira oko danog problema kako bi ga riješili na što efikasniji način.



Slika 1 XP metodika

Ekstremno programiranje poboljšava softverski projekt u pet osnovnih puteva, gdje se također gledaju kao vrijednosti Ekstremnog programiranja:

* komunikacija,
* jednostavnost,
* povratna veza,
* poštovanje i
* hrabrost

Ekstremni programeri konstantno komuniciraju sa svojim korisnicima i kolegama programerima. Održavaju svoj dizajn jednostavnim i čistim. Dobivaju povratnu vezu testiranjem programskog proizvoda od samog početka razvoja (prvog dana). Dostavljaju sustav korisniku što je prije moguće implementirajući promjene prema sugestijama. Svaki mali uspjeh ovisi o poštovanju prema jedinstvenom doprinosu svakog člana u timu. Radi navedenih karakteristika Ekstremno Programiranje hrabro odgovara na promjene u zahtjevima i tehnologiji [2].

Pravila Ekstremnog programiranja može biti podijeljeno u nekoliko kategorija [2]:

* Planiranje
  + Korisničke priče (zahtjevi) su zapisane
  + Planiranje distribucije kreira kalendar distribucije
  + Pravljenje učestalih malih distribucija
  + Projekt je podijeljen u iteracije
  + Svaka iteracija počinje planiranjem iteracije
* Upravljanje
  + Dodjeljivanje timu fleksibilni radni prostor
  + Postaviti održivi tempo
  + Stand up sastanak svaki dan
  + Napredak projekta (Project Velocity) se mjeri
  + Cirkulacija ljudi
  + Popravljanje XP-a kad se pokvari
* Dizajn
  + Jednostavnost
  + Odabir sistemske metafore
  + Korištenje CRC kartica za dizajn sesije
  + Kreiranje spike rješenja za smanjenje rizika
  + Funkcionalnost se ne dodaje u ranoj fazi
  + Refaktoriranje kad god je moguće
* Kodiranje
  + Korisnik je uvijek raspoloživ
  + Kod mora biti napisan po dogovorenim standardima
  + Kodiranje Unit testova prvo -> TDD
  + Svaki produkcijski kod je programiran u paru
  + Samo jedan par integrira kod
  + Česta integracija
  + Postavljanje integracijskog računala
  + Korištenje kolektivnog vlasništva
* Testiranje
  + Svaki kod mora imati Unit test
  + Svaki kod mora proći unit testiranje prije distribucije
  + Ako je bug pronađen testovi su kreirani
  + Test prihvatljivosti se često provodi te se rezultati objavljuju

Ekstremno programiranje se javlja kao odgovor na problematiku zahtjeva za izmjenom. Korisnik u početku ne treba imati solidnu ideju kako bi njegov sustav trebao izgledati, također neki sustavi zahtijevaju izmjenu funkcionalnosti svakih nekoliko mjeseci. U mnogim programskim proizvodima potreba za dinamičkim izmjenama je konstantna. Ovo su obilježja okoline gdje će Ekstremno Programiranje uspjeti dok druge metodike neće [2].

XP je također postavljen kako bi postigao rješavanje problema vezanih uz projektni rizik. Ako korisnici zahtijevaju novi sustav do određenog datuma rizik je velik. Ako je taj sustav novi izazov za grupu rizik je još veći. Ako je taj sustav novi izazov za cijelu industriju rizik je na najvećoj razini. XP praksa je postavljena za ublažavanje rizika i povećavanje vjerojatnosti uspjeha.

XP je postavljen za malu grupu programera: između 2 i 12, dok je projekt od 30 također zabilježen kao uspješan. Projektanti, odnosno developeri ne trebaju posjedovati napredno znanje programiranja kako bi radili s XP-om, ali XP se ne može koristiti na projektima s velikim brojem osoblja. Treba naglasiti da na projektima s dinamičkim zahtjevima i velikim rizicima mali timovi s XP-om daje bolje rezultate nego veliki tim.

XP je postao de-facto kad se govori o agilnom načinu razvoja programskih proizvoda. Slika 2. predstavlja uvod u Agilni razvoj programskih proizvoda [2].



Slika 2 Agilni razvoj

## XP planiranje

Planiranje se odnosi na dva glavna pitanja u procesu razvoja: što će biti napravljeno  do određenog dana te određivanje što treba sljedeće napraviti.

Kod planiranja se koriste korisničke priče, a služe za potrebe planiranja izlazaka pojedinih inačica programa. Koriste  se umjesto dugih dokumenata sa zahtjevima, a u njima korisnici opisuju što bi sustav trebao raditi u par rečenica (korisnikova, a ne tehnička terminologija). Za svaku korisnikovu priču potrebno je izraditi najmanje jedan test kojim će se potvrditi ispunjenje korisnikove priče. Svrha ovih priča je da se procijeni trajanje implementacije, tj. prije same implementacije, programeri moraju dobiti detaljnije specifikacije.

“Na zajedničkom sastanku kreira se raspored verzija sustava. Taj raspored služi da bi se kreirali planovi za svaku pojedinačnu iteraciju. Preporuka je često izdavanje verzija koje ne trebaju biti značajno poboljšane, samo je važno otkriti funkcionalnosti od koji korisnik ima koristi za poslovanje te ih isporučiti čim prije. Sustav se razvija u iteracijama koje traju 1 do 3 tjedna pri čemu one trebaju biti približno iste duljine. Sve programske aktivnosti nije potrebno planirati na samom početku projekta već se aktivnosti planiraju na početku pojedine iteracije. To je takozvano JIT (engl. Just In Time) planiranje iteracija koje omogućava brzo praćenje promjena korisničkih zahtjeva.

Svaki programer procjenjuje koliko mu je idealnih radnih dana potrebno za ispunjenje zadatka, a ako je procjena veća od 3 radna dana, zadatak je potrebno raščlaniti na manje zadatke. U slučaju da unutar iteracije nije moguće izvršiti zadatke, rokovi se ne smiju pomicati te stoga treba izbaciti neke od zadataka.

Potrebno je mjeriti i brzinu razvoja projekta. Pri tome se procjenjuje trajanje ispunjenja korisničkih priča u pojedinim iteracijama na temelju prijašnjih iteracija” [1].

## XP dizajniranje

Jednostavnost dizajna omogućava da se projekt brže izvrši tako da u glavne ciljeve spada određivanje dizajna na kojeg se troši malo vremena.

“Prilikom dizajniranja potrebno je koristiti prikladne metafore za konzistentno definiranje klasa i objekata. One se koriste za dizajniranje objektno orijentiranih programa i pomažu shvatiti koje su programske klase potrebne te kako će one međusobno djelovati.

U slučaju kompliciranih tehničkih ili dizajnerskih problema potrebno je kreirati potencijalna zamjenska rješenja. Cilj tomu je smanjiti rizik mogućih tehničkih problema ili povećati pouzdanost procjena korisničkih priča.

Prilikom dizajniranja potrebno je paziti da se ne dodaju nepotrebne funkcionalnosti pošto se samo 10% tako ekstra dodanih funkcionalnosti ikad koristiti. I na kraju kod dizajna je potrebno primjenjivati redizajn, odnosno odbacivati nepotrebne funkcionalnosti , te ukloniti redundancije” [1].

## XP programiranje

“Tijekom razvoja sustava važan je preduvjet dostupnost krajnjeg korisnika i to ne samo kako bi pomogao razvojnom timu već i bio član razvojnog tima, budući da sve faze XP razvoja zahtijevaju komuniciranje s korisnicima.

Preporuka je da  i da se unaprijed kreiraju osnovni testovi za provjeru funkcionalnosti, prije same izrade programskog koda. Time se ne produžuje izrada samog osnovnog programskog koda, a kasnije nije potrebno trošiti dodatno vrijeme na izradu testova.

Još jedna od preporuka, koja programerima na početku može biti čudna, jest programiranje u parovima. To znači da jedna osoba piše programski kod dok ga druga kontrolira. Na kraju to bi trebalo rezultirati mnogo višom razinom kvalitete sustava te jednakom količinom funkcionalnosti kao da su programeri radili odvojeno.

Isto tako vrlo važno je da se u uključivanje novog programskog koda  u cjelokupni sustav koristi sekvencijalna integracija. To znači da se zabranjuju slučajevi u kojima bi različiti programeri istodobno uključivali svoj programski kod i testirali ga.

Što se tiče integracije programskog koda, tu postoji pravilo , a ono je da se programski kod često integrira. Preporuka je da se programski kod uključuje u sustav svakih nekoliko sati te da se ne čeka duže od jednog dana, ali po pravilu sekvencijske integracije. Važno je da tijekom projekta postoji, tzv. zajedničko vlasništvo na programskim kodom, tj. svi programeri mogu slobodno mijenjati gotovi programski kod,  koji nisu sami napisali, kako bi dodali novu funkcionalnosti ili ispravili pogreške” [1].

## XP testiranje

“Jedna od temelja ekstremnog programiranja je korištenje osnovnih testova za provjeru ispravnosti implementiranih funkcija. Za samo testiranje potrebno je ili uzeti ili razviti vlastito programsko okruženje pomoću kojega će se moći kreirati automatizirani testovi. Testove je potrebno kreirati prije razvoja samog programskog koda sustava. Stoga  razvijeni programski sustav ne može biti integriran u cjelokupni sustav ako ga ne prate popratni testovi.

U slučajevima kada se u programskom kodu otkrije neki tip pogreške potrebno je kreirati testove za pogrešku kako se ne bi kasnije u drugim modulima ponovila. Iz definiranih korisničkih priča kreiraju se testovi za prihvaćanje sustava. Klijent definira scenarije koje je potrebno testirati na temelju definiranih korisničkih priča, a za jednu priču moguće je definirati jedan ili više testova”. [1]

# Alati korišteni za realizaciju projekta

## TFS

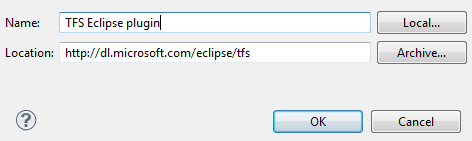
URL: <https://sharecard.visualstudio.com>

Odabrali smo Team Foundation Server (skraćeno TFS) kao glavni repozitorij našeg projekta i hosting za subverzioniranje zbog njegove efikasnosti i pristupačnosti. Također TFS je besplatan za projekte na kojima radi do 5 članova.

## Instalacija

Kako bi se TFS mogao koristiti unutar Eclipse razvojnog sučelja potrebno je instalirati dodatak (eng. Plugin).

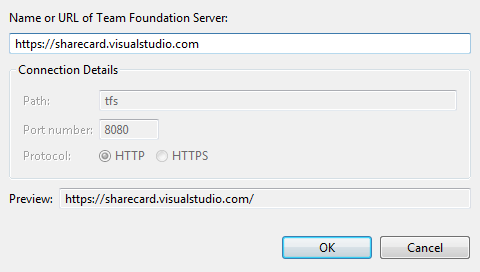
1. Unutar Eclipsa kliknemo na Help > Install New Software te zatim na „Add“
2. U novom prozoru potrebno je napisati naziv dodatka te adresu s koje se on preuzima (<http://dl.microsoft.com/eclipse/tfs>)



Slika 3 Prikaz forme iz koraka 2.

1. U sljedećem prozoru odabere se zadnja verzija dodatka i dovrši sama instalacija pritiskom na gumb „Next“.

Nakon što je instaliran dodatak potrebno je spojiti se sa serverom. Nakon što otvorimo Team Explorer prozor potrebno je kliknuti na „Connect to TFS“. U novom prozoru je potrebno kliknuti na „Servers…“ te zatim na Add. U novom prozoru potrebno je upisati pristupne podatke servera. Na sljedećoj slici vidljivi su podaci za naš projekt.



Slika 4 Upis pristupnih podataka

Nakon inicijalne provjere da li je moguće uspostaviti vezu sa server, server će zatražiti pristupne podatke korisnika (Windows Live ID). Prethodno osoba koja je kreirala server mora dodati članove tima kako bi se oni mogli spajati.

## Korištenje

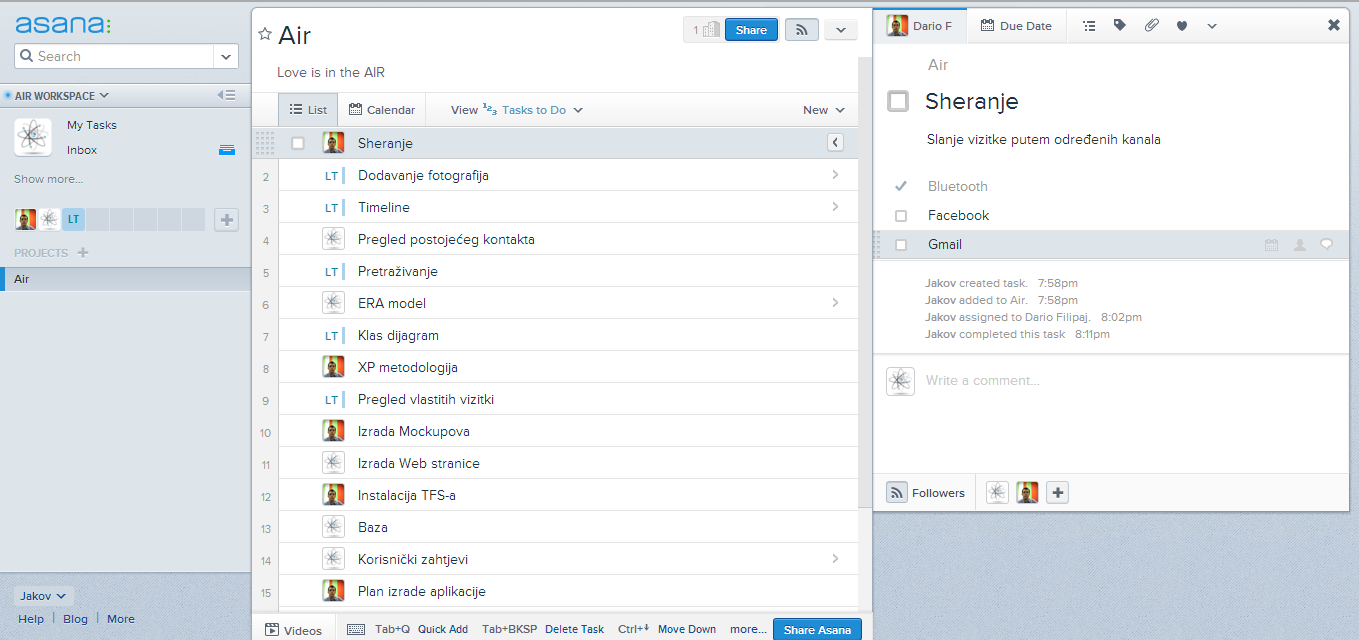
Kao i kod mnogih sličnih usluga unutar TFS projekt je podijeljen u nekoliko grana (eng. branch). Naš projekt se sastoji od glavne „grane“ i sporednih u kojima će raditi svaki član tima. Svaki član će imati svoju „granu“ kako ne bi došlo do kolizija i neplaniranih grešaka unutar koda.

Nova grana se stvara klikom na glavnu granu „shareCard“ i zatim na „branch“ unutar „Source control“ prozora kojeg je moguće otvoriti iz Team Explorer prozora. Potrebno je odabrati naziv nove grane te lokalni folder u koji će se spremati. Bitno je naglasiti kako je sve promjene na TFS potrebno „Check-inati“ tj. potvrditi i uploadati na server (desni klik „Check-in Pending Changes…“). Tako je nakon stvaranja brancha potrebno taj branch check-inati kako bi se on lokalno preuzeo.

Sličan je postupak i s uploadanjem novih verzija projekta. Nakon što je član tima zadovoljan s promjenama potrebno ih je mergati tj. spojiti u glavni branch. To se radi desnim klikom na branch unutar „Source control“ prozora te odabirom opcije „merge“. Automatski je ponuđeno spajanje sporednog brancha s glavnim. Ponovno je bitno naglasiti kako je nakon odabrane opcije sve izmjene potrebno „check-inati“, odnosno potvrditi (desni klik „Check-in Pending Changes…“).

## Asana

Asana web alat nam je omogućijo da definiramo glavne funkcionalnosti projekta. Uz pomoć njega realizirali smo „kartice s pričom“, odnosno opise svojstva koji se dodaju programskoj podršci. Također alat nam je poslužio kao ToDo lista gdje se jasno može vidjeti tko je šta radio. Slika 5 prikazuje primjer „kartice s pričom“.



Slika 5 Asana web alat

## Microsoft Visio

Microsoft Visio je programski alat iz paketa Microsoft Office koji nam je poslužio prilikom kreiranja određenih dijagrama kao što su ERA model, klas dijagram i dijagram slučajeva korištenja. Kako imamo slobodan pristup Microsoftovim alatima putem faksa iskoristili smo Visio kako bi si olakšali daljnji razvoj projekta.

# Projektni plan

shareCard je aplikacija namijenjena kreiranju vizitki te životopisa u obliku vremenske crte. Korisnici kroz interaktivno sučelje unose svoje podatke i kreiraju vlastite vizitke i životopis koje mogu razmjenjivati sa poslovnim partnerima, prijateljima te svim pojedincima s kojima žele dati svoj kontakt.

Ciljna skupina na koju se orijentiramo su ponajprije mlađi poslovni ljudi koji su informatički obrazovani. Jednostavna razmjena informacija i kontakata funkcionalnost je kojom namjeravamo privući korisnike, kao i jednostavnost i korisnost aplikacije u svakodnevnoj primjeni. Besplatna verzija aplikacije sadržavat će osnovne funkcionalnosti koje korisnicima omogućavaju nesmetano korištenje aplikacije, no ukoliko žele dodatne funkcionalnosti kao što su dodatni predlošci, dodavanje loga i slika, kreiranje većeg broja vizitki, imat će mogućnost kupovanja proširene verzije aplikacije.

Marketinške aktivnosti bit će usmjerene uglavnom na našu ciljanu skupinu u obliku online promotivnih materijala. Prilikom izrade aplikacije, izradit će se i web stranica na kojoj će korisnici moći dobiti sve potrebne informacije i podršku te davati povratne informacije o kvaliteti aplikacije. Prema našim istraživanjima, trenutno ne postoji aplikacija koja omogućava iste ili slične funkcionalnosti te samim time ostvarujemo stratešku prednost na tržištu. Osnovna prijetnja je neprihvaćanje i neprepoznavanje aplikacije od strane korisnika, a tu prijetnju ćemo nastojati umanjiti istraživanjem tržišta i prilagodbom aplikacije potrebama korisnika.

Za izradu aplikacije koristit ćemo open source razvojno okruženje i aplikacije sa studentskim licencama te za taj aspekt razvoja nećemo morati izdvajati financijska sredstva. Posjedujemo vlastita računala i android uređaje za testiranje aplikacije što uvelike umanjuje troškove razvoja i testiranja. Resurs koji ćemo najviše ulagati je vrijeme i rad, odnosno kroz vremenski period od 2 mjeseca koristit ćemo stečeno znanje kako bi napravili funkcionalnu aplikaciju. Financijska sredstva uložit ćemo u zakupljivanje servera i domene te izradu promotivnih materijala i oglašavanje.



## Korisnički zahtjevi

U narednim točkama definirani su glavni korisnički zahtjevi na temelju čega se definira daljnji razvoj projekta, terminiranje istog i razvijanje programskog koda. Korisnički zahtjevi:

* Kreiranje vlastite vizitke koja će sadržavati osobne podatke i sliku.
* Mogućnost djeljenja vlastite vizitke.
* Mogućnost primanja i pregleda tuđih vizitki.
* Mogućnost izmjene postojeće vlastite vizitke.
* Mogućnost postavljanja vizitki na web stranicu.
* Mogućnost kreiranja i editiranja životne crte.
* Mogućnost registriranja premium korisničkog računa.

## Pregled funkcionalnosti besplatne i premium verzije

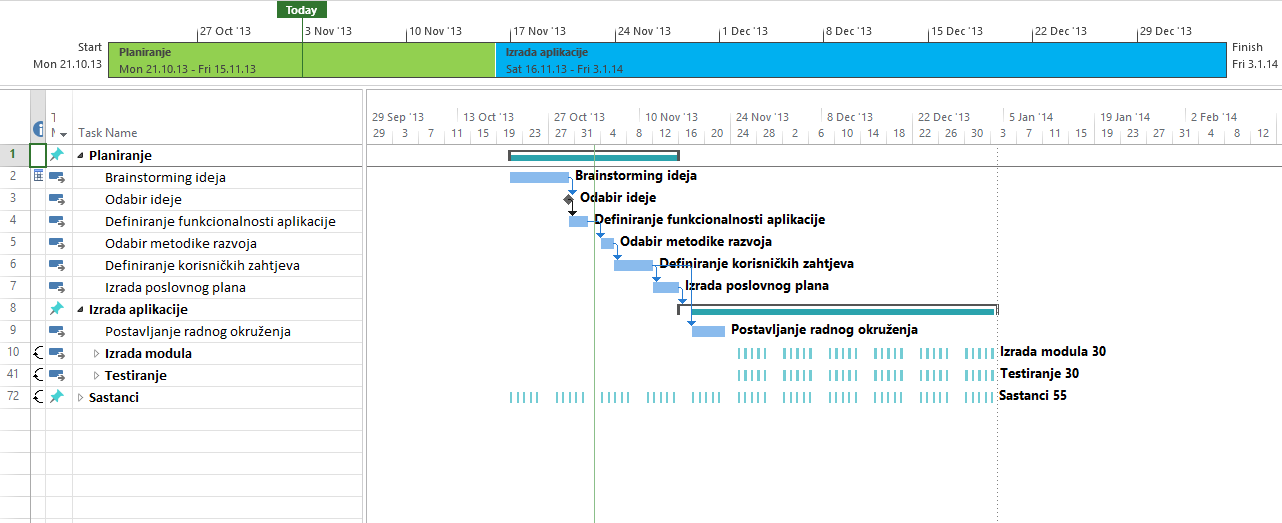
Besplatna verzija

* Kreiranje jedne pojedinačne i jedne grupne vizitke
  + Korisnik u besplatnoj verziji ima mogućnost kreiranja samo jedne osobne vizitke i jedne grupne vizitke
* Tri ponuđena predloška za kreiranje vizitke
  + Korisnik u besplatnoj verziji, pri kreiranju vizitke, može odabrati samo između tri preddefinirana predloška.
* Povezivanje s društvenim mrežama (Facebook, LinkedIn...)
  + Povezivanje korisnika s društvenim mrežama radi popunjavanja potrebnih osobnih podataka i radi djeljenja vizitki.
* Dijeljenje i pregled vizitki
  + Mogućnost djeljenja vizitki s korisnicima ili potencijalnim korisnicima (nije potrebno biti korisnik da bi se vizitka primila).
* Pretraživanje baze vizitki
  + Korisnik u besplatnoj verziji ima pristup samo onim vizitkama koje su od strane korisnika postavljenji na „public“.
* Posjeta na web stranicu shareCard
  + Potencijalni Premium korisnik ima mogućnost pristupa web stranici samo kao „posjetitelj“. Posjetitelj može pretraživati i upoznavati se s web sučeljem, informirati se o mobilnoj aplikaciji, web aplikaciji i pogodnostima premium računa.
  + Posjetitelj se može registrirati kao normalan ili Premium korisnik.

Premium verzija (sadrži sve što i Besplatna verzija)

* Kreiranje neograničenog broja vizitki
  + Premium korisnik ima mogućnost kreiranja neograničenog broja vizitki
* Veliki broj predložaka i kreiranje vlastitog predloška
  + Premium korisnik ima pristup bazi podataka predložaka, također ima mogućnost kreiranja vlastitog predlopka na web stranici
* Pretraživanje baze vizitki
  + Premium korisnik ima pristup onim vizitkama koje su od strane korisnika postavljene na „public“, „group“ (određena grupa), „registered“ (samo registriranim korisnicima).
* Prijava na web stranicu shareCard
  + Korisnik se putem mobilne aplikacije može registrirati na Premium račun i putem tih podataka omogućen mu je pristup (putem prijave) na web stranicu aplikacije shareCard be potrebe dodatne registracije putem Web applikacije. Premium korisnik unutar web aplikacije ima mogućnost kreiranja novih predložaka za vizitku kao i mgučnost kreiranja „n“ broja vizitki

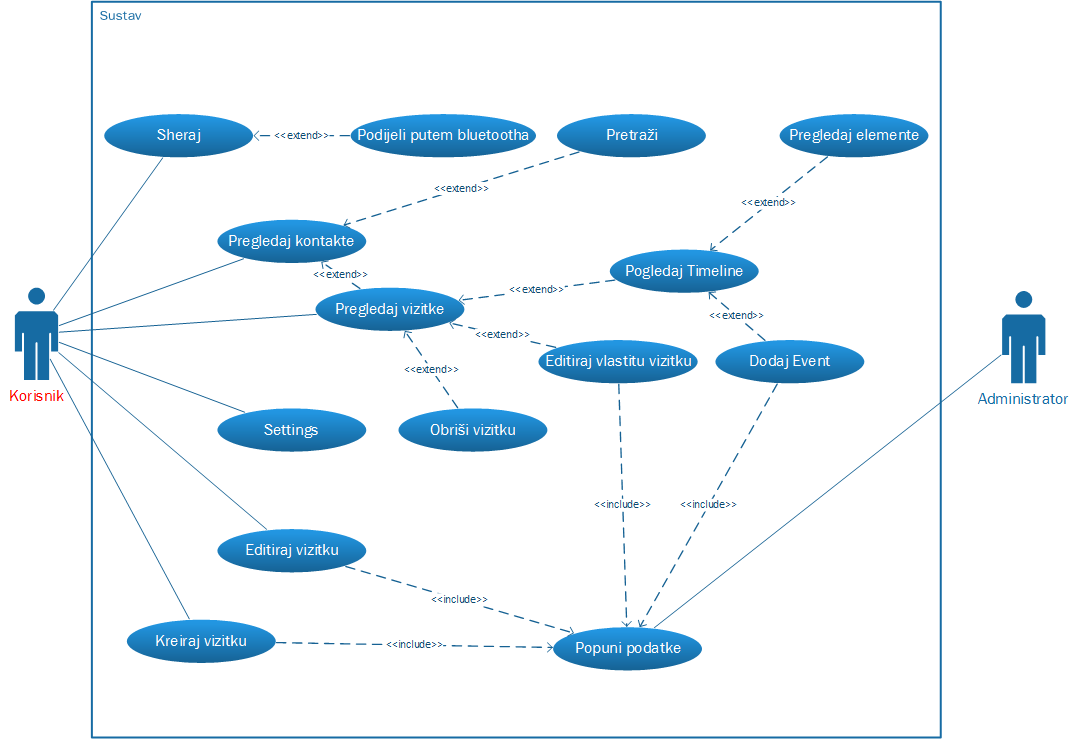
## Terminiranje Projekta



Slika 6 Gantogram projekta

Slika 6 prikazuje gantogram projekta, odnosno proces izrade aplikacije shareCard od samog nastajanja ideje do implementacije i završnog testiranja. Projekt je podijeljen u dvije faze, planiranje i izradu aplikacije. Prva faza sadrži procese kreiranja ideja, odabira jedne od alternativa, definiranje funkcionalnosti i metodike razvoja te definiranje korisničkih zahtjeva i izradu poslovnog plana. Faza planiranja traje kraće od faze izrade aplikacije koja sadrži postavljanje radnog okruženja te izradu modula i testiranje. Pošto smo odabrali XP metodu, izrada modula i testiranje odvijaju se paralelno kao ponavljajući zadatci. Za metodiku je također karakteristično održavanje svakodnevnih sastanaka tima što je također realizirano ponavljajućim zadatkom. Svi navedeni procesi grafički su prikazani desno na slici 1. Većina procesa ima određene preduvjete koji moraju biti zadovoljeni kako bi sljedeći procesi mogli započeti s radom (naznačeno strelicama na gantogramu).

# Dijagram slučajeva korištenja mobilne aplikacije



Slika 7 Dijagram slučajeva korištenja

Dijagram slučajeva korištenja (Slika 7) prikazuje pojedini scenarij kada se korisnik nalazi u interakciji sa sustavom. Bitno je naglasiti da Dijagram slučajeva korištenja (Use case dijagram) naglašava što treba učiniti, a ne na koji način ostvariti. Gledajući aspekt korisnika, dozvoljene su mu funkcionalnosti poput:

* **Sheraj** - djeljenje vlastite vizitke putem raznovrsnih kanala
  + Djeljenje vizitki moguće je realizirati pomoću bluetootha i ostalih popratnih kanala
* **Pregledaj vizitke** - pregled svih postojećih vizitki (primljenih i kreiranih)
  + Pregled vizitke omogućuje niz funkcionalnosti kao što su brisanje određene viztke, editiranje vlastite vizitke te pregled životne crte (Timeline).
  + Timeline omogućuje dodavanje novih evenata koji karakteriziraju korisnika (certifikati, poslovi, volontiranje itd.) te pregled istih.
* **Editiraj vizitku** - editiranje vlastitih vizitki (mjenjanje osobnih podataka) i promjena istih u mobilnoj bazi
* **Pregled kontakta** - pregled svih kontakta preko kojih se može pretraživati određeni kontakti upisujući određena slova.
  + Preko pregleda moguće je također pregledati sve vizitke klikom na određeni kontakt. Ukoliko je vlastita vizitka, korisnik je u mogućnosti editirati istu.
* **Kreiraj vizitku** - korisnok je u mogućnosti kreirati novu vizitku koju može dalje proslijeđivati.
  + Prilikom kreiranja vizitke popunjavaju se osobni podaci popraćeni slikom koji se na kraju spremaju u mobilnu bazu

# Korisnička priča

Cilj korisničke priče je predstaviti način i razlog korištenja aplikacije u obliku privlačnom korisnicima ili ulagačima. Korisnička priča se često bazira na razgovoru razvojnog tima koji ih je doveo do same ideje ili glavnoj, najvažnijoj funkcionalnosti aplikacije. Korisničke priče su često glavni element prezentiranja tj. pitchanja projekata pred publikom. One su uspješne ako u njoj publika prepozna korist aplikacije ili se poistovjeti s njom. S obzirom na odabranu metodiku korisničku priču je potrebno kreirati tako da bude sažeta i jednostavna, ali istovremeno mora pratiti realne funkcionalnosti aplikacije. Detaljnije upoznavanje s korisničkom pričom planiramo provesti komunikacijom „licem u lice“, odnosno tijekom prezentacije projekta. Ovakav pristup nam omogućava realniju procjenu uspješnosti projekta, ali i kvalitetnije povratne informacije u realnom vremenu. Povratne informacije su nam važne zbog procjene broja potencijalnih korisnika, troškova, mogućih problema i uočavanja eventualnih nepredviđenih benefita aplikacije. Korisnička priča u jednoj rečenici:

*„Kao korisnik na jednostavan način kreiram i dijelim svoje kontaktne podatke s potencijalnim partnerima uz minimalne troškove.“*

Proširena korisnička priča:

Korisnik A:

*„Kao korisnik shareCard aplikacije na jednostavan način kreiram i dijelim svoje kontaktne podatke s potencijalnim partnerima uz minimalne troškove.. Klasične posjetnice mogu potrošiti, a izrada novih posjetnica mi oduzima vrijeme i resurse. Kreiranjem shareCard posjetnica u digitalnom obliku rješavam navedene probleme.“*

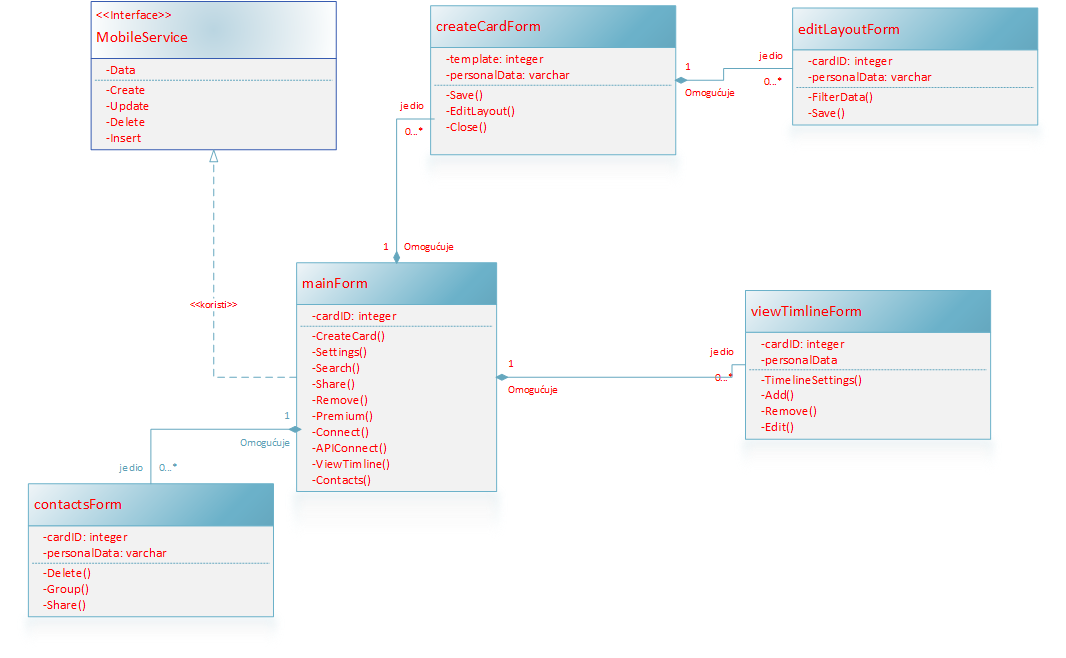
Korisnik B:

*„Kao korisnik shareCard aplikacije na brz i jednostavan način uređujem sve svoje posjetnice i čuvam posjetnice podijeljene sa mnom. Kreiranje više posjetnica specijaliziranih za neki od mojih projekata omogućuje mi ostavljanje profesionalnog dojma pred potencijalnim partnerima. Kopiju svih posjetnica čuvam i detaljnije uređujem putem shareCard Web mjesta.“*

Korisnik C:

*„Kao korisnik shareCard kreiram svoj životopis u obliku vremenske crte na nov, efektivan i brz način. Kreiran životopis dijelim s potencijalnim partnerima putem weba i mobilne aplikacije bez dodatnih troškova.*

# Dijagram klasa

****

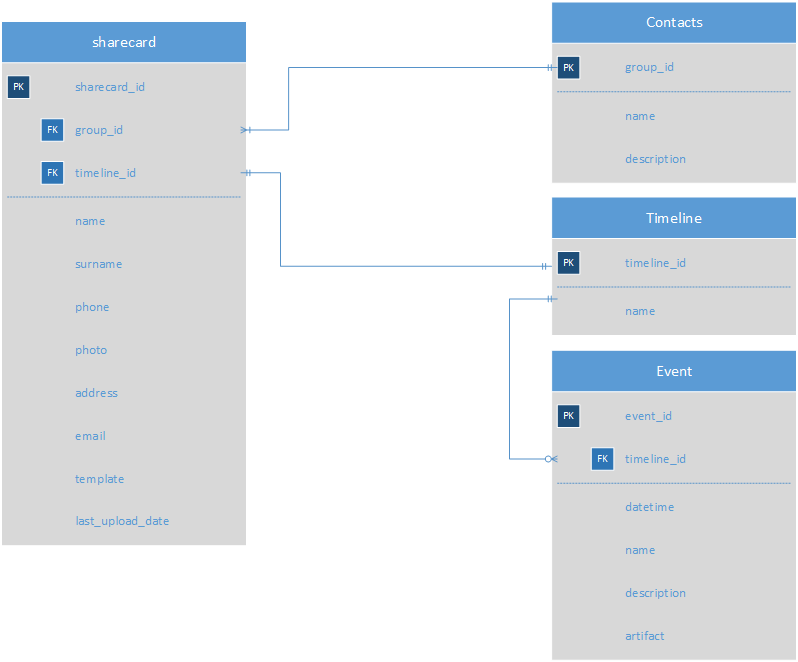
Slika 8. Dijagram Klasa

Slika 7. prikazuje dijagram klasa naše aplikacije koji opisuje unutarnju strukturu i realizaciju svih procesa., atributa operacija itd. Prikazane su moguće veze između postojećih klasa. Kako je sami projekt još u ranoj fazi razvoja, moguće su izmjene, popravci ili proširenja samog dijagrama. Dijagram se sastoji od četiri klase (mainForm [početni ekran], contactsForm [prikaz svih primljenih vizitki], viewTimelineForm [prikaz vremenske crte pojedine vizitke], createCardForm [izrada nove vizitke], editLayoutForm [editiranje postojeće vlastite vizitke]), te jednog interfacea (MobileService [sinkronizacija podataka s web bazom]).

# ERA model

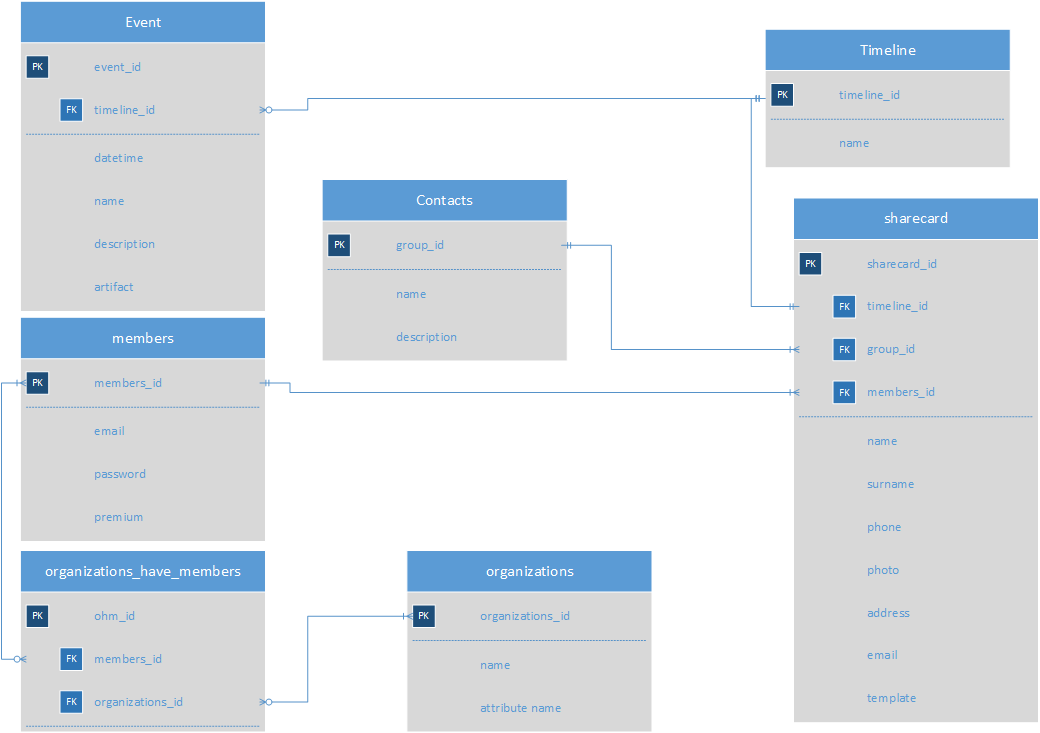
Projekt se sastoji od dvije baze podataka. Prva baza podataka nalazi **se** na mobilnom uređaju te služi za pohranu podataka kreiranih unutar mobilne aplikacije. Druga baza podataka nalazi se u sklopu Web mjesta te služi kao mjesto za pohranu kopije podataka iz mobilne aplikacije te pohranu dodatnih podataka generiranih a webu.

Sljedeće slike prikazuju razlike u modelu baze podataka. U osnovni web baza podataka je proširena verzija mobilne baze podataka te služi kao sigurnosna kopija. Sinkronizacija podataka obavljat će se prilikom spajanja mobilnog uređaja na internet. U slučaju da se mobilni uređaj ne spaja na internet sve funkcionalnosti će biti dostupne unutar mobilne aplikacije, ali napredne web funkcionalnosti neće biti aktivne. Sljedeća slika prikazuje era dijagram mobilne aplikacije.



Slika 9. ERA mobilne aplikacije

Uz mobilnu, postojat će i web aplikacija čiji je era model prikazan na slici 9.



Slika 10. ERA web aplikacije

# Plan izrade aplikacije

## Prva faza

### Planiranje releasea

#### Faza istraživanja

U pvoj fazi istraživanja definirali smo zahtjeve i procjenili potrebnu radnu dinamiku kako bi ostvarili projektne ciljeve za prvu fazu. Definirani sati su u intervali od početka projekta do predaje za prvu fazu.

4 - sata (definiranje korisničkih zahtjeva, definiranje početnog izgleda, definiranje osnovnih koncepata baze, klas dijagrama )

4 - sata (popunjavanje dokumentacije)

2 - sata (postavljanje okoline, određivanje potrebnih alata)

#### Faza opredjeljenja

Fazom opredjeljenja smo definirali cijenu, korist i opterećenost obaveza, gdje smo definirali i posložili glavne elemente po važnosti Tablica 1.

Procjena izrade mobilne aplikacije (podjeljeno na dvije osobe):

----------------

10 - sati (samoučenje, istraživanje, najbolje prakse)

10 - sati (izrada glavnih funkcionalnosti)

5 - sati (izrada grafičkih elemenata)

----------------

Ukupno: 25 radnih sati

Procjena izrade Web aplikacije (samostalni rad):

----------------

5 - sati (definiranje i početak izrade glavnih funkcionalnosti)

5 - sati (izrada grafičkog sučelja)

5 - sati (samoučenje, istraživanje, najbolje prakse)

-----------------

Ukupno: 15 radnih sati

Tabela 1 Važnost elemenata projekta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Elementi | Index završenst  0 - završen  1 -ne završen  2 - nepoznao | Index promjenjivosti  0 - nizak  1 - srednji  2 - visok | Index težine  0 - jednostavan  1 - standardan  2 - komplexan | Važnost  x>=5 - vrlo važan  x<=5 i x>=3 - važan  x<3 - normalan |
| Izrada Mobilne aplikacije | 2 | 1 | 2 | 5 |
| Izrada Web aplikacije | 1 | 2 | 1 | 4 |
| Izrada dokumentacije | 1 | 2 | 0 | 3 |
|  | | | | |

#### Faza upravljanja

U fazi upravljanja imali smo male promjene što se tiče korisničkih zahtjeva. Dodali smo mogučnost registracije na Web aplikaciji. U satnici nije bilo velikih odstupanja.

### Planiranje iteracija

#### Faza istraživanja

U fazi istraživanja kod planiranja iteracije definirali smo glavne točke izrade mobilne i web aplikacije Tablica 2 .

Tabela 2 Glavne točke izrade aplikacija

|  |  |
| --- | --- |
| **Mobilna aplikacija** | **Web aplikacija** |
| 1. Postavljanje radne okoline | 1. Postavljanje radne okoline |
| 1. Kreiranje glavne forme apliacije | 2. Kreiranje početne stranice |
| 1. Kreiranje početne baze mobilne aplikacije | 3. Kreiranje glavnih funkcionalnosti |
| 1. Kreiranje dodatnih formi | 4. Kreiranje web baze |
| 1. Prvi test | 5. Testiranje |
| 1. Procjena testa | 6. Procjena testa |
|  |  |

**Procjena testa mobline aplikacije:**

* Stabilna funkcionalnost.
* Dovršenost - 10%

**Procjena testa web aplikacije:**

* Stabilna funkcionalnost
* Dovršenost - 5%

#### Faza opredjeljenja

U fazi predjeljenja kod planiranja iteracije definirani su korisnički zahtjevi.

#### Faza upravljanja

Nastavak izrade mobilne i web aplikacije po zadanim kriterijima.

## Druga faza

### Planiranje relesea

#### Faza istraživanja

Uzevši u obzir prvu fazu i uspješnost izrade, definirane su novi ciljevi za drugu fazu.

4 - sata (korekcija dokumentacije)

2- sata (modifikacija klas dijagrama, modifikacija era dijagrama)

4 - sata (definiranje početnog user interfacea)

#### Faza opredjeljenja

Uzevši u obzir rezultate prve faze definiramo procjena za nastavak razvoja mobilne i web aplikacije

Procjena izrade mobilne aplikacije (podjeljeno na dvije osobe):

----------------

10 - sati (samoučenje, istraživanje, najbolje prakse)

10 - sati (modifikacija i dodavanje novih funkcionalnosti)

10 - sati (izrada grafičkih elemenata)

----------------

Ukupno: 30 radnih sati

Procjena izrade Web aplikacije (samostalni rad):

----------------

5 - sati (izrada glavnih funkcionalnosti)

5 - sati (izrada grafičkog sučelja)

5 - sati (samoučenje, istraživanje, najbolje prakse)

-----------------

Ukupno: 15 radnih sati

#### Faza upravljanja

Nakon definiranja korekcija naspram prve faze uočili smo i napravili potrebne izmjene. Korisnički zahtjevi nisu se promjenili.

### Planiranje iteracije

#### Faza istraživanja

U fazi istraživanja kod planiranja iteracije definirali smo dodatne točke za mobilnu i web aplikaciju Tablica 3.

Tabela 3 Dodatne točke aplikacija

|  |  |
| --- | --- |
| **Mobilna aplikacija** | **Web aplikacija** |
| 1. Dodavanje podataka | 1. Cijelovita funkcionalnost |
| 1. Kreiranje interaktivnih elemenata | 2. Povezanost stranica |
| 1. Pisanje i čitanje iz baze | 3. Pregled elemenata |
| 1. Popunjavanje obrazaca | 4. Pisanje i čitanje iz baze |
| 1. Testiranje | 5. Testiranje |
| 1. Procjena testa | 6. Procjena testa |
|  |  |

**Procjena testa mobline aplikacije:**

* Nestabilna funkcionalnost
* Dovršenost - 40%

**Procjena testa web aplikacije:**

* Stabilna funkcionalnost
* Dovršenost - 45%

#### Faza upravljanja

Uočeni su propusti u mobilnoj aplikaciji, početak definiranja korekcije grešaka. Potrebno sinkroniziranje mobilne i web aplikacije. Nedostatak opreme (server). Web aplikacija se radi lokalno što sprječava razvoj sinkroniziranja s mobilnom aplikacijom

## Treća faza

Osvrt na ocjenu druge faze:

Nedostatak popračenosti metodike, nedovoljno razvijena funkcionalnost, nedostatak komentiranja koda.

### Planiranje releasea

#### Faza istraživanja

Uzevši u obzir prvu i drugu fazu i uspješnost izrade, definirane su novi ciljevi za treću fazu.

4 - sata (korekcija dokumentacije)

2- sata (modifikacija funkcionanosti, optimizacija koda)

4 - sata (definiranje fiinalnog izgleda kompletne aplikacije)

#### Faza opredjeljenja

Uzevši u obzir rezultate prve i druge faze definiramo procjenu za nastavak razvoja mobilne i web aplikacije

Procjena izrade mobilne aplikacije (podjeljeno na dvije osobe):

----------------

10 - sati (samoučenje, istraživanje, najbolje prakse)

20 - sati (modifikacija i dovršetak svih glavnih funkcionalnosti)

10 - sati (izrada grafičkih elemenata)

----------------

Ukupno: 40 radnih sati

Procjena izrade Web aplikacije (samostalni rad):

----------------

10 - sati (dovršavanje funkcionalnosti)

4 - sati (izrada grafičkog sučelja)

5 - sati (optimizacija)

-----------------

Ukupno: 19 radnih sati

#### Faza upravljanja

Svi ključni elementi su realizirani. Glavni problem sinkronizacija web i mobilne aplikacije

### Planiranje iteracije

#### Faza istraživanja

U trećoj fazi istraživanja kod planiranje iteracije sve glavne točke su pokrivene, radi nedostatke određene opreme nije moguće u potpunosti realizirati sve funkcionalnosti

**Procjena testa mobline aplikacije:**

* Polovično stabilna funkcionalnost
* Dovršenost - 70%

**Procjena testa web aplikacije:**

* Stabilna funkcionalnost
* Dovršenost - 65%

#### Faza opredjeljenja

U potpunosti su realizirani korisnički zahtjevi

#### Faza upravljanja

Uočeni su nedostatci u mobilnoj aplikaciji. Radi nedostatka servera u nemogučnosti smo sinkronizirati podatke s mobilne aplikacije na web aplikaciju.

Zaključni komentar: mobilna i web aplikacija imaju stabilnu funkcionalnost ali su u potpunosti odvojeni.

# Početna shema i Web aplikacija

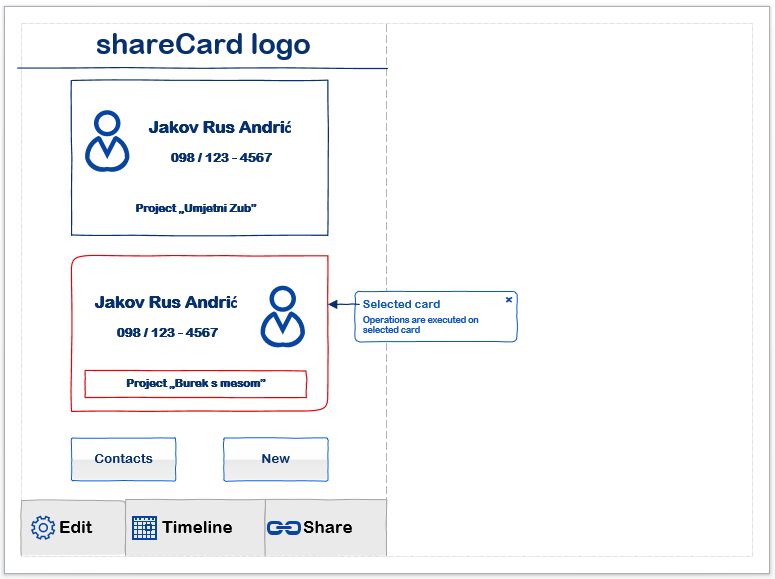
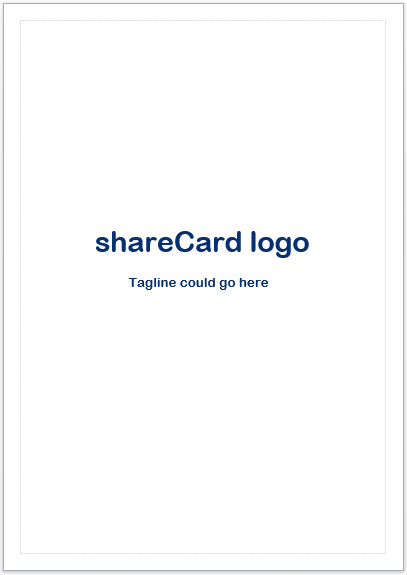
U nastavku se nalazi početna shema mobilne aplikacije koja predstavlja samu funkcionalnost aplikacije. Cilj je pridržavati se određenih ključnih točaka kako bi uspijeli realizirani zamišljeni koncept aplikacije. Također u priloženoj datoteci se nalazi trenutačna verzija mobilne aplikacije.

Trenutna Web aplikacija je još u izradi, kako bi ju dovršili potrebna nam je kompletna Android mobilna aplikacija kako bi izvršili sinkronizaciju potrebnih podataka. Također u priloženom se nalazi djelomično gotova Web aplikacija s formama:

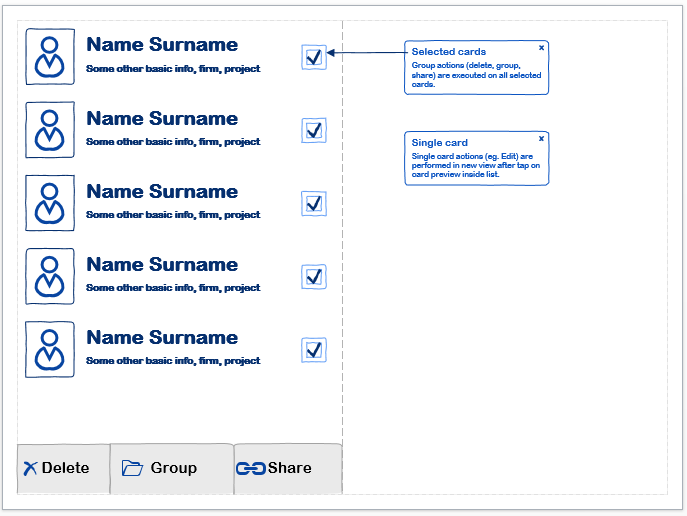
* Login Forma (Slika 15)
* Registracijska Forma (Slika 16)
* Početna Forma (Slika 17)
* Learn more Forma (Slika 18)
* Ispis iz baze (Slika 19)
* Individualna vizitka (Slika 20)
* Prikaz Timelina (Slika 21)

Web aplikacija je namjenjena za kreiranje novih vizitki odnosno predložaka te detaljni pregled Baze vizitki svih odobrenih od strane korisnika. Web aplikacija je rađena uz pomočć Microsoftovog alata Web Matrix. Također napominjem da je trenutačna faza rađena lokalno te da se stranice još uvijek ne nalazi na serveru. Trećom fazom očekujemo punu funkcionalnost Web aplikacije i postavljenjena server.

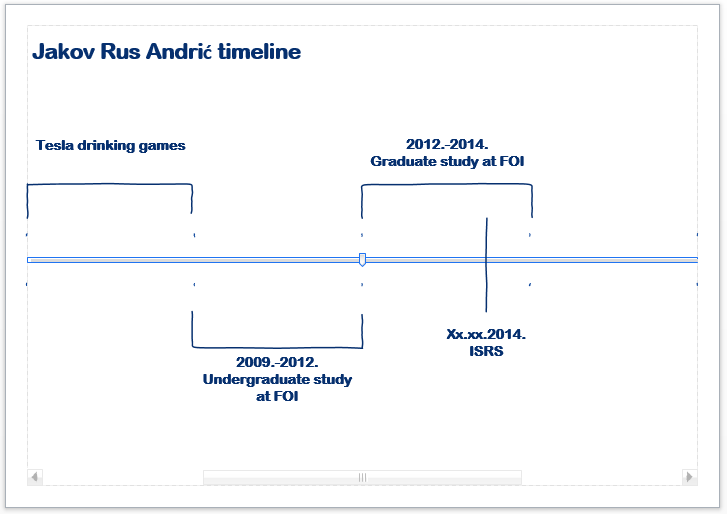
# Početna shema aplikacije



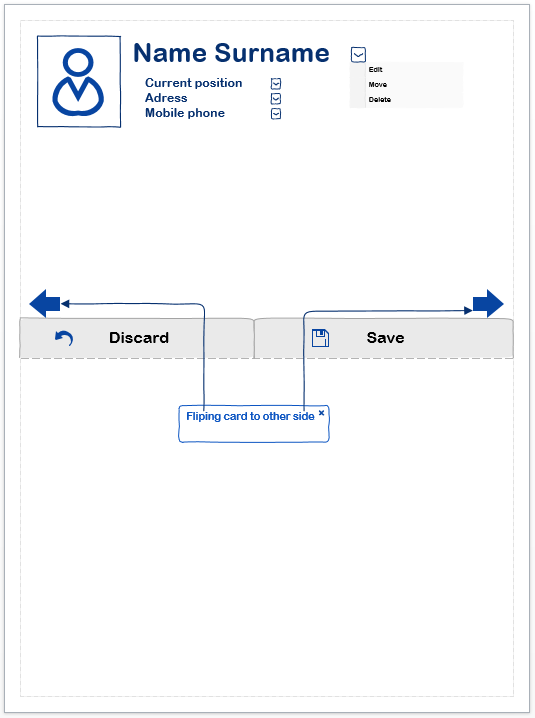
Slika 11. shareCard ekran učitavajuči Slika 12. shareCard početni ekran



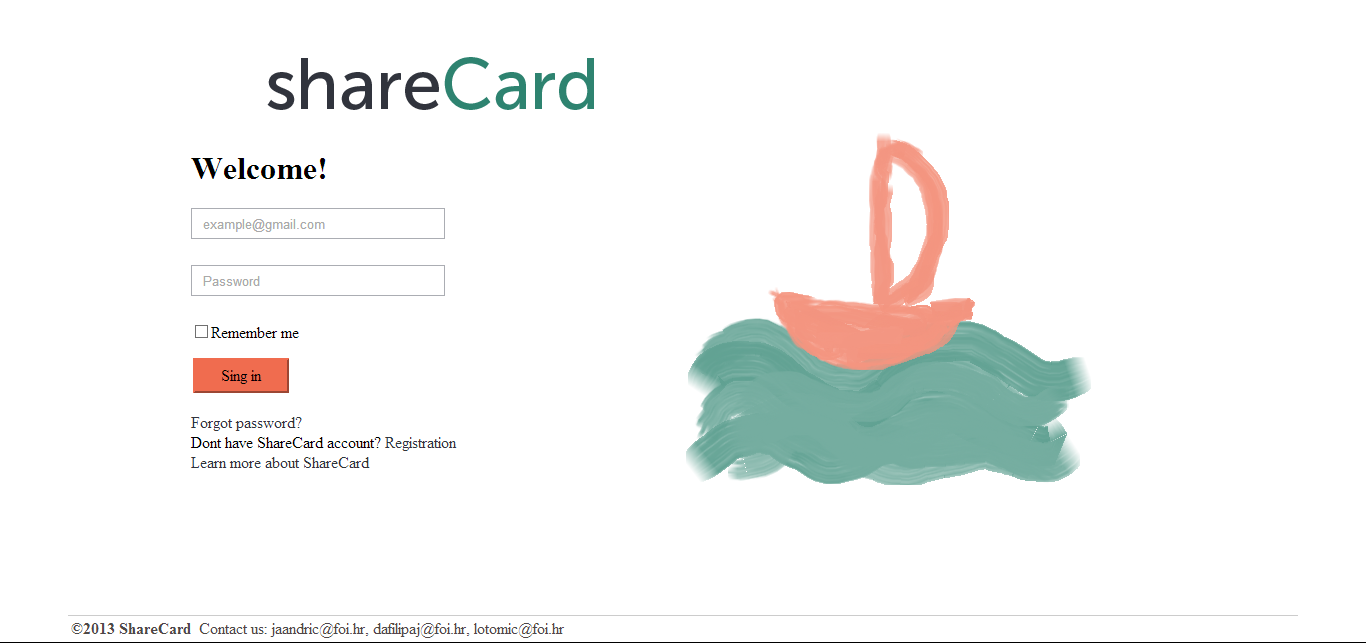
Slika 13. shareCard Lista vizitki



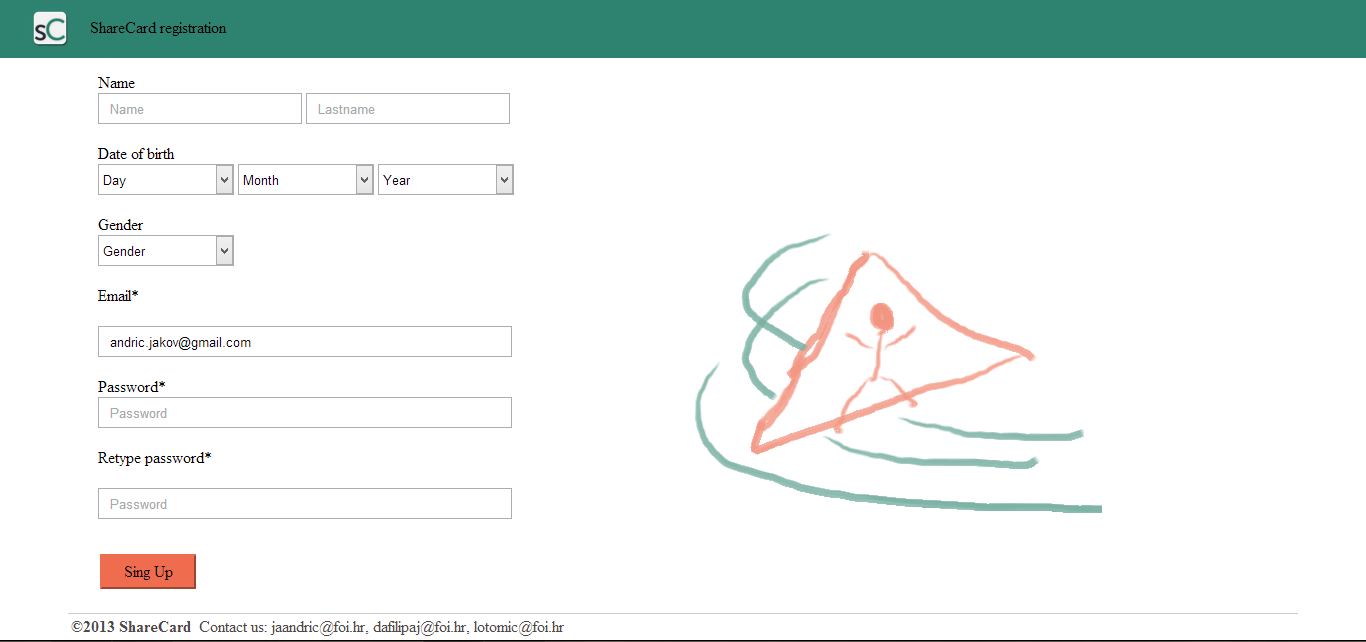
Slika 14. shareCard (pogled) Vremenska crta



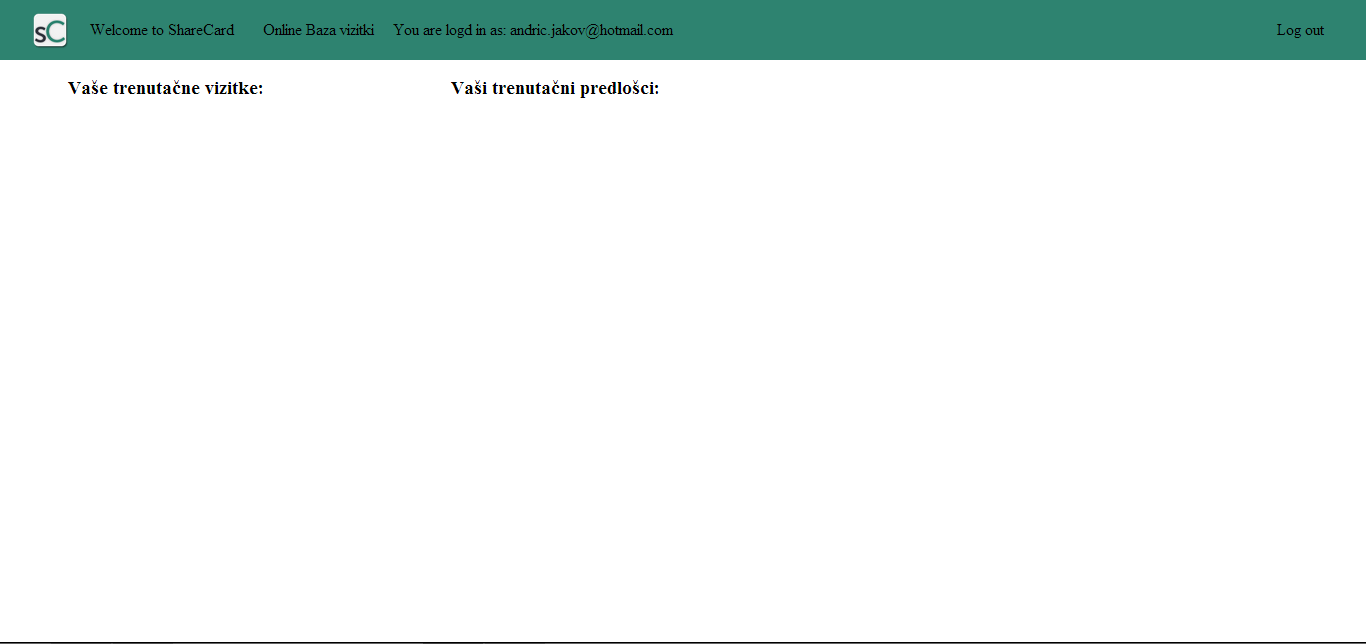
Slika 15. shareCard editiranje vizitk



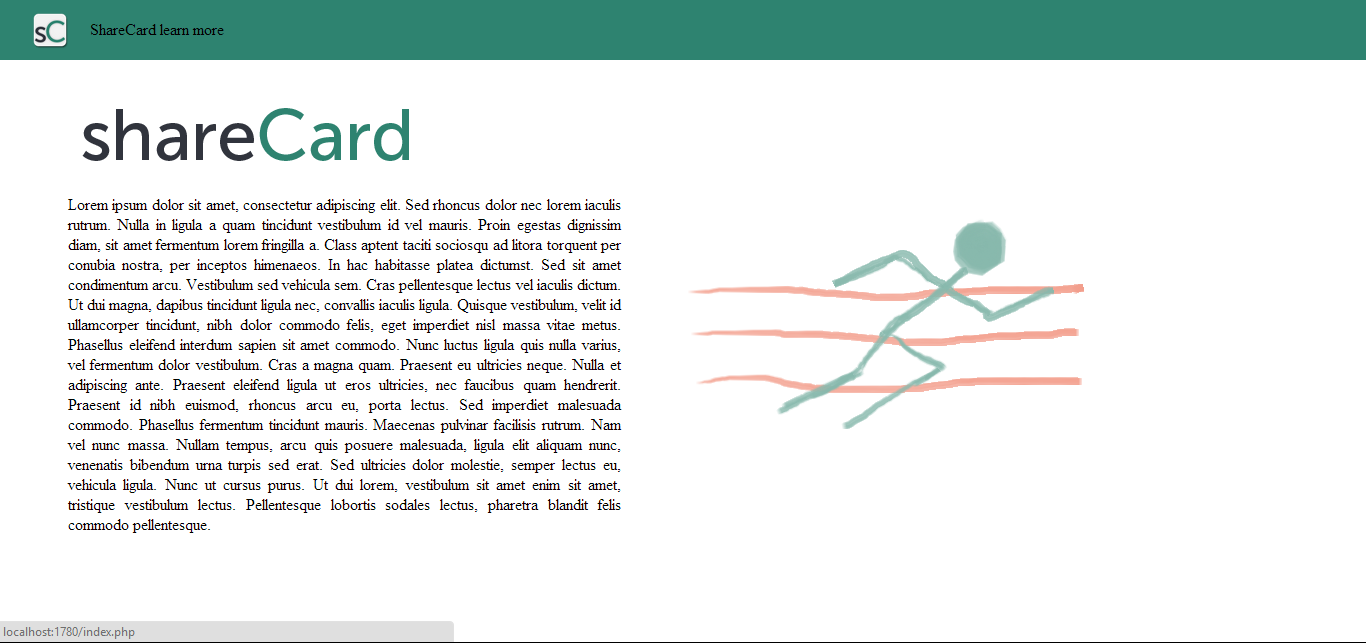
Slika 16 Login forma Web aplikacije



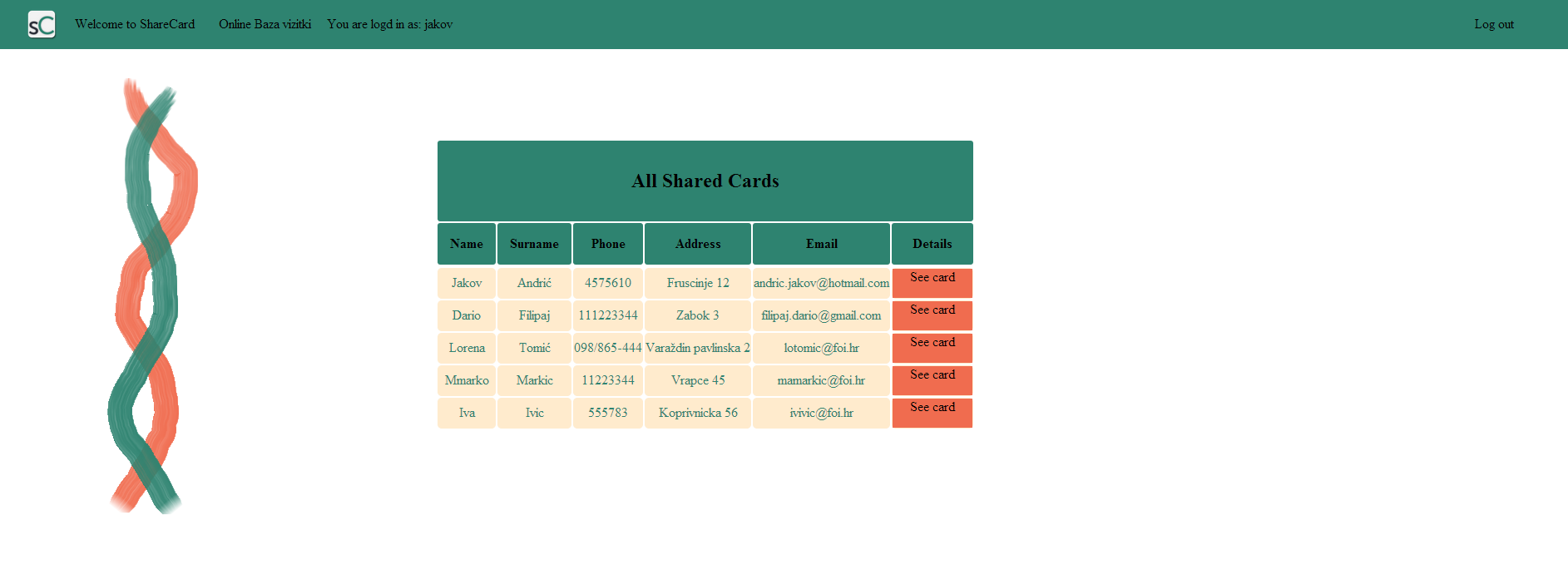
Slika 17 Registracijska forma Web aplikacije



Slika 18 Početna forma Web aplikacije



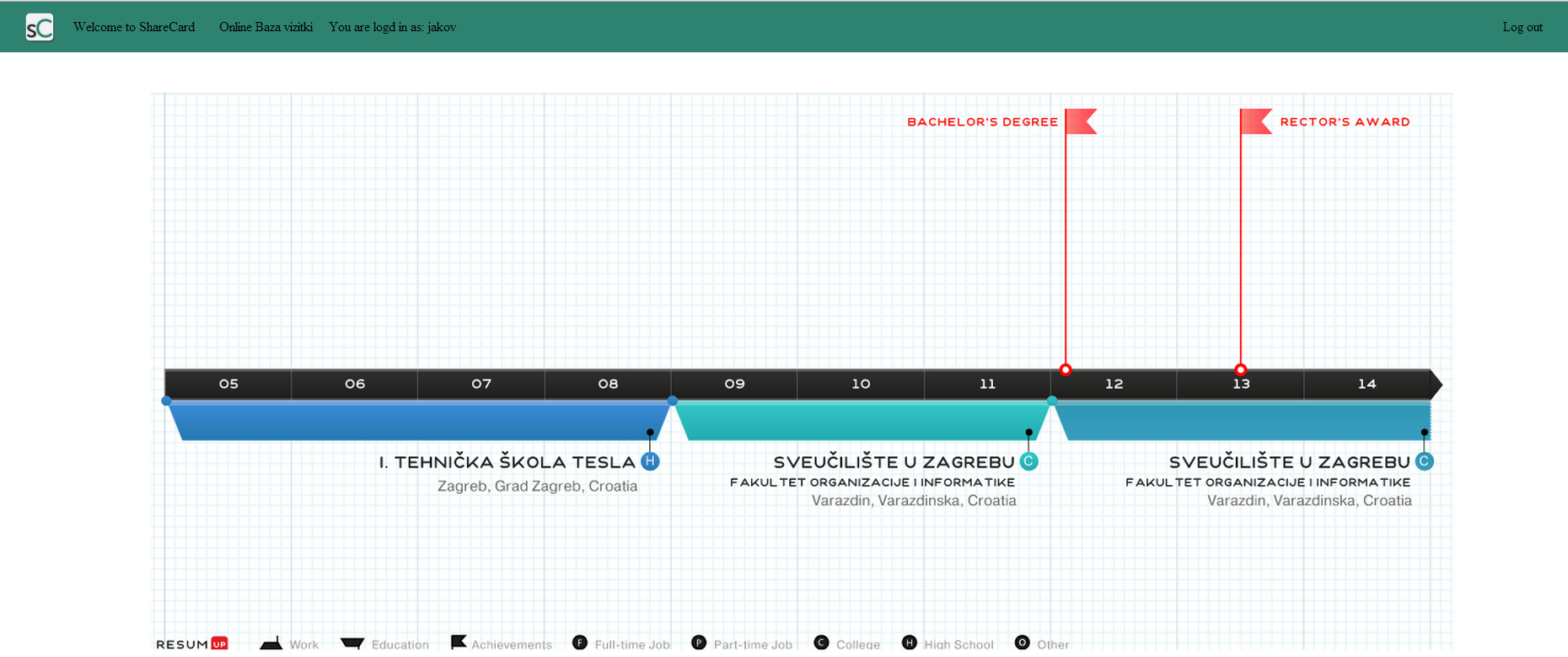
Slika 19 Learn more forma Web aplikacije



Slika 20 Ispis iz baze



Slika 21 Individualna vizitka



Slika 22 Prikaz Timelina

# Literatura

* [1] Info Novitas: Ekstremno Programiranje: <http://www.info-novitas.hr/o-nama/metodologije-rada/ekstremno-programiranje-xp/>
* [2] Course on Software Analysis and Development, Methodologies for development of mobile applications, Zlatko Stapić, M.A.