Oblikovanje programske potpore

Ak. god. 2018./2019.

*ParkirajMe*

Dokumentacija, Rev. 1

Grupa: *ferovci*

Voditelj: *Hrvoje Ditrih*

Datum predaje: *<dan>. <mjeseca> <godine>.*

Nastavnik: *Nikolina Frid*

Sadržaj

[1.](#_3fwokq0) Dnevnik promjena dokumentacije 3

[2.](#_3znysh7) Opis projektnog zadatka 5

[3.](#_tyjcwt) Pojmovnik 7

[4.](#_1t3h5sf) Funkcionalni zahtjevi 8

[4.1 Opis obrazaca uporabe 10](#_4d34og8)

[5.](#_17dp8vu) Ostali zahtjevi 11

[6.](#_26in1rg) Arhitektura i dizajn sustava 12

[6.1.](#_35nkun2) Svrha, opći prioriteti i skica sustava 13

[6.2.](#_44sinio) Dijagram razreda s opisom 14

[6.3.](#_z337ya) Dijagram objekata 15

[6.4.](#_1y810tw) Ostali UML dijagrami 16

[7.](#_2xcytpi) Implementacija i korisničko sučelje 17

[7.1.](#_3whwml4) Dijagram razmještaja 17

[7.2.](#_qsh70q) Korištene tehnologije i alati 18

[7.3.](#_1v1yuxt) Isječak programskog koda vezan za temeljnu funkcionalnost sustava 19

[7.4.](#_4f1mdlm) Ispitivanje programskog rješenja 20

[7.5.](#_2u6wntf) Upute za instalaciju 21

[7.6.](#_19c6y18) Korisničke upute 22

[8.](#_3o7alnk) Zaključak i budući rad 23

[9.](#_ihv636) Popis literature 24

[Chapter 2 Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda) 25](#_1hmsyys)

[Chapter 3 Dodatak B: Dnevnik sastajanja 26](#_41mghml)

[Chapter 4 Dodatak C: Prikaz aktivnosti grupe 27](#_2grqrue)

[Chapter 5 Dodatak D: Plan rada / Pregled rada i stanje ostvarenja 29](#_vx1227)

*Sadržaj bi se trebao automatski osvježavati prema tekstu (desni klik, „Update Field“) ako se bude držalo zadanih formata poglavlja.*

# **Dnevnik promjena dokumentacije**

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev.** | **Opis promjene/dodatka** | **Autor(i)** | **Datum** |
| 0.1 | Napravljen predložak. | Ditrih | 16.10.2018. |
| 0.2 | Opis projektnog zadatka | Požeg, Bilanović | 17.10.2018. |
| 0.3 | Funkcionalni zahtjevi | Bilanović,  Požeg | 18.10.2018. |
| 0.7 |  |  | 26.08.2013. |
| 0.8 |  | Ivošević | 28.08.2013. |
| 0.82 | Opisi obrazaca uporabe | Jović | 07.09.2013. |
| 0.88 | Preveden uvod | Jović | 08.09.2013. |
| 0.9 | Sekvencijski dijagrami | Žužak | 09.09.2013. |
| 0.91 | Započeo dijagrame razreda | Horvat | 10.09.2013. |
| 0.92 | Nastavak dijagrama razreda | Horvat | 11.09.2013. |
| 1.0 | Verzija samo s bitnim dijelovima za 1. ciklus | Ivošević | 11.09.2013. |
| 1.1 | Uređivanje teksta – funkcionalni i nefunkcionalni zahtjevi | Grudenić | 14.09.2013. |
| 1.11 | Manje izmjene: Timer > Brojilo vremena | Grudenić | 15.09.2013. |
| 1.12 | Popravljeni dijagrami obrazaca uporabe | Jović | 15.09.2013. |
| 1.5 | Generalna revizija strukture dokumenta | Ivošević | 19.09.2013. |
| 1.55 | Manja revizija (dijagram razmještaja) | Jović | 20.09.2013. |
| 2.0 | Konačni tekst predloška dokumentacije | Ivošević | 28.09.2013. |
| 3.0 | Nadopuna poglavlja o arhitekturi | Jović | 25.11.2013. |
| 4.0 | Nadopune po pitanju praćenja aktivnosti u grupi | Ivošević | 01.10.2015. |
| 4.1 | Definirani Dodaci C i D | Ivošević | 05.10.2015. |

*Moraju postojati glavne revizije (verzije, inačice) dokumenata 1.0 i 2.0 na kraju prvog i drugog ciklusa. Između tih revizija mogu postojati manje revizije već prema tome kako se dokument bude nadopunjavao. Očekuje se da nakon svake značajnije promjene (dodatka, izmjene, uklanjanja dijelova teksta i popratnih grafičkih sadržaja) dokumenta se to zabilježi kao revizija. Npr., revizije unutar prvog ciklusa će imati oznake 0.1, 0.2, …, 0.9, (ukoliko bude više od toga nastavlja se sa 0.10, 0.11…) sve do konačne revizije prvog ciklusa 1.0. U drugom ciklusu se nastavlja s revizijama 1.1, 1.2, itd. Ukoliko se želi naglasiti veličina promjene/dodatka unutar revizije može se napraviti različita „granulacija“ u označavanju revizija. Npr. nakon 0.1 može biti 0.11, 0.12, 0.15 pa tek onda 0.2. Ili se za veće promjene/zahvate može nakon 0.1 odmah skočiti na 0.5. itd. Bitno je da na krajevima ciklusa revizije koje se predaju budu s oznakama* ***1.0*** *i* ***2.0*** *(Preporuka je da te revizije budu podebljane unutar liste).*

# **Opis projektnog zadatka**

Cilj ovog projekta je izraditi web aplikaciju koja će korisnicima pružati informacije o slobodnim parkirnim mjestima te njihovu rezervaciju na području grada Zagreba. Raspoloživa parkirališta bit će sva ona koje će zainteresirane tvrtke registrirati putem iste aplikacije. Aplikacija će raditi u stvarnom vremenu te koristiti korisnikovu lokaciju pomoću koje će se odrediti najbliže parkiralište sa slobodnim mjestom. Nakon određivanja parkirnog mjesta korisniku se šalje informacija o lokaciji parkinga.

Razlog pokretanja projekta je olakšavanje i ubrzavanje procesa traženja parkirnog mjesta u gradu. Želja je korisniku ponuditi individualni odabir te cijene odabranog parkirališta. Također pružamo lakši način zainteresiranim tvrtkama za promidžbu svojih parkirališta te lakšu digitalnu naplatu.

Glavni aktori uz korisnike su tvrtke koje nude svoja parkirna mjesta. Kako bi koristio usluge aplikacije korisnik se prvo mora registrirati na web stranici, što je omogućeno svakoj punoljetnoj osobi. Korisnik tijekom regristracije unosi sljedeće podatke: OIB, ime, prezime, adresa e-pošte, broj registracije svog automobila te broj kreditne kartice. Registrirani korisnik ima na raspolaganju sljedeće opcije:

1. Traženje najbližih slobodnih parkirališta

Korisnik odabire gumb za pretragu najbližih slobodnih mjesta temeljenih na njegovoj lokaciji. Zatim dobiva na odabir nekoliko najboljih opcija zajedno sa informacijom o cijeni te on na temelju toga odabire parking.

1. Rezervacija parkirališnih mjesta

Korisnik odabire gumb za rezervaciju, željeno parkiralište te način rezervacije:

* 1. Jednokratna

-za vremenski period kraći od 24 sata

-obavlja se minimalno 6 sati unaprijed

-naplaćuje se odmah u trenutku rezervacije izravnim terećenjem kreditne kartice

* 1. Ponavljajuća

-mora trajati najmanje 1 sat i ponavljati se barem jednom tjedno tijekom mjesec dana ili trajno (0-24h svaki dan na neodređeni period)

-trajna rezervacija naplaćuje se svakih 30 dana

Tvrtke koje žele nuditi svoja parkirališta unutar aplikacije također se trebaju registrirati putem aplikacije. Za registraciju ovlašteni zaposlenik unosi sljedeće podatke: OIB, ime, adresa sjedišta, adresa e-pošte. Nakon registracije unose se podatci o određenim parkiralištima koje ta tvrtka nudi. Kada korisnik izvrši rezervaciju on plaća izravno aplikaciji koja ta sredstva raspodjeljuje tvrtkama vlasnicima parkinga.

# **Pojmovnik**

WEB (World Wide Web) – najkorištenija usluga interneta koja omogućava dohvaćanje hipertekstualnih dokumenata.

# **Funkcionalni zahtjevi**

Dionici:

* Klijenti (registrirani korisnici)
* Tvrtke (registrirani vlasnici parkirališta)
* Administratori

Aktori:

* Svi posjetitelji WEB stranice
  + Mogu se registrirati kao korisnik ili tvrtka
* Registrirani klijent
  + Može se prijaviti u sustav
  + Može zatražiti informacije o slobodnim parkirnim mjestima
  + Može zatražiti cijenu određenih parkirnih mjesta
  + Može odabrati određeni parking od ponuđenih
  + Može rezervirati parkirno mjesto jednokratno
  + Može rezervirati ponavljajuće parkirno mjesto
  + Može izvršiti uplatu rezervacije
* Registrirana tvrtka
  + Može se prijaviti u sustav
  + Može ponuditi informacije o svojim parkiralištima
  + Može primati uplatu rezervacija
* Administratori
  + Može upravljati korisnicima sustava
  + Može u slučaju brisanja korisničkog računa tvrtke obrisati rezervacije korisnika koji su rezervirali parkirno mjesto u vlasništvu te tvrtke
  + Može vratiti novac korisniku u slučaju preostalog neiskorištenog rezerviranog vremena

Sudionici koji sudjeluju u sustavu su:

* Baza podataka:
  + Sprema podatke o registriranim korisnicima i tvrtkama
  + Sprema podatke o rezervacijama
* Poslužitelj:
  + Obrađuje zahtjeve korisnika

## 4.1 Opis obrazaca uporabe

* Rbr.: UC1
  + Naziv: Registracija
  + Opis: Registracija korisnika na web aplikaciju
  + Glavni aktor: Neregistrirani korisnik
  + Preduvjeti: Korisnik je punoljetan
  + Pokretač: Pritisak na gumb registracija
  + Opis osnovnog tijeka:
    - Korisnik upisuje svoje podatke
    - Baza podataka uspješno sprema podatke
  + Opis mogućih odstupanja:
    - Korisnik upisuje svoje podatke
    - Baza podataka ne sprema podatke zbog preduvjeta
* Rbr.: UC2
  + Naziv: Prijava
  + Opis: Prijava korisnika na web aplikaciju
  + Glavni aktor: Registrirani korisnik
  + Preduvjeti: Korisnik je prethodno registriran
  + Pokretač: Pritisak na gumb prijava
  + Opis osnovnog tijeka:
    - Korisnik upisuje svoje podatke
    - Sustav provjerava ispravnost podataka te proslijeđuje klijenta na stranicu za prijavljene klijente
  + Opis mogućih odstupanja:
    - Korisnik pogrešno upisuje svoje podatke

* + - Sustav provjerava ispravnost podataka i javlja grešku
* Rbr.: UC7
  + Naziv: Brisanje rezervacije
  + Opis: Korisnik briše postojeću jednokratnu ili trajnu rezervaciju
  + Glavni aktor: Registrirani korisnik
  + Preduvjeti: Korisnik je prethodno registriran i rezervacija postoji
  + Pokretač: Pritisak na gumb *Obriši*
  + Opis osnovnog tijeka:
    - Korisnik u izborniku bira opciju brisanja pritiskom na gumb
    - Od korisnika se traži potvrda akcije
    - Sustav provjerava postoji li tražena rezervacija
    - Oslobađa se rezervirani termin
    - Korisniku se rezervacija briše iz popisa postojećih rezervacija
  + Opis mogućih odstupanja:
    - Korisnik pokušava obrisati nepostojeću rezervaciju (npr. zahtjev se pošalje dvaput)
    - Korisniku je istekla sjednica
* Rbr.: UC8
  + Naziv: Traženje najbližeg parkinga
  + Opis: Korisnik od aplikacije traži parking najbliži zadanoj lokaciji
  + Glavni aktor: Registrirani korisnik
  + Preduvjeti: Korisnik je prethodno registriran
  + Pokretač: Pritisak na gumb *Potraži*
  + Opis osnovnog tijeka:
    - Korisnik bira opciju pretrage po lokaciji
    - Korisnik upisuje traženu lokaciju //pretpostavljam da tako radimo
    - Sustav obavlja pretragu i vraća rezultate
    - Korisniku se rezultati pretrage prikazuju sortirani po udaljenosti u rastućem poretku
  + Opis mogućih odstupanja:
    - Korisniku je istekla sjednica
    - Nije pronađeno niti jedno parkiralište //pretpostavljam da cemo imat neku gornju granicu na radijus pretrage, u tom slucaju dajemo opciju prosirivanja radijusa valjda?
* Rbr.: UC9
  + Naziv: Uređivanje podataka o parkingu
  + Opis: Tvrtka dodaje novi parking (//ovo možda u posebni UC?) ili uređuje podatke o postojećem parkingu
  + Glavni aktor: Registrirana tvrtka
  + Preduvjeti: Tvrtka je prethodno registrirana
  + Pokretač: Pritisak na gumb *Uredi*
  + Opis osnovnog tijeka:
    - //ostavljam prazno dok ne dogovorimo jel odvajamo dodavanje novih il ne
  + Opis mogućih odstupanja:
    - Korisniku koji upravlja podacima tvrtke je istekla sjednica
* Rbr.: UC10
  + Naziv: Pregled svih parkirališta
  + Opis: Registriranom korisniku se prikazuju sva postojeća parkirališta
  + Glavni aktor: Registrirani korisnik
  + Preduvjeti: Korisnik je prethodno registriran
  + Pokretač: Pritisak na gumb *Potraži*
  + Opis osnovnog tijeka:
    - Korisnik bira opciju pretrage svih parkirališta
    - Sustav obavlja pretragu i vraća rezultate
    - Korisniku se rezultati prikazuju //ovaj put nisu sortirani jer nemamo uvjete sortiranja?
  + Opis mogućih odstupanja:
    - Korisniku je istekla sjednica

# **Ostali zahtjevi**

*Ostali ili nefunkcionalni zahtjevi nisu izravno vezani za primarnu funkciju sustava. Govore o određenim mjerama kvalitete i svojstvima i ograničenjima platforme i procesa ostvarenja sustava.*

*Pod mjerama i zahtjevima na kvalitetu sustava misli se na nekoliko aspekata: neosjetljivost na pogreške, (is)koristivost, pouzdanost, performanse, (p)održivost.*

*Pod ograničenjima se uglavnom misli na svojstva tehnologije i metodologiju implementacije: ograničenja tehnologije, vrijeme izrade, (ne)usvojeni standardi, informacije o procesu razvoja i oblikovanja.*

*Svaki takav zahtjev potrebno je navesti u jednoj ili dvije rečenice.*

# **Arhitektura i dizajn sustava**

*Ovo je je jedno od bitnijih poglavlja jer se opisuju temelji implementacije sustava. Moguće je razlikovati više aspekata arhitekture i oblikovanja sustava od kojih je većinu potrebno dokumentirati već prema zahtjevima projektnog zadatka. Neki od tih su: stil arhitekture i identifikacija podsustava, preslikavanje na radnu platformu, spremišta podataka, mrežni protokoli, globalni upravljački tok, sklopovsko-programski zahtjevi.*

*U svakoj stavci navedenoj u nastavku potrebno je dati tekstovni opis. te ilustrirati odgovarajućim UML dijagramima. Projektna grupa odlučuje gdje i koje sve dijagrame treba prikazati u pojedinim stavkama, ali cijela točka 6. Arhitektura i dizajn sustava, mora sadržavati barem po jedan:*

*Dijagram razreda*

*Dijagram objekata*

*Dijagram stanja*

*Dijagram aktivnosti*

*Komunikacijski dijagram*

*Dijagram komponenti*

*Ostali UML dijagrami (npr. Sekvencijski za objekte) mogu se koristiti po potrebi.*

*Za crtanje UML dijagrama preporuka je koristiti alat Astah Community.*

*Preporuča se da ovo poglavlje podijelite po sljedećim potpoglavljima:*

*6.1. Svrha, opći prioriteti i skica sustava*

*6.2. Dijagram razreda s opisom*

*6.3. Objektni dijagram.*

*6.4. Ostali UML dijagrami*

## *Svrha, opći prioriteti i skica sustava*

*Potpoglavlje treba sadržavati sljedeće:*

*Koji sustav ili dio sustava ovaj dio dokumentacije opisuje, izbor arhitekture temeljem principa oblikovanja pokazanih na predavanjima (objasniti zašto ste baš odabrali takvu arhitekturu).*

*Organizacija sustava s najviše razine apstrakcije (npr. klijent-poslužitelj, baza podataka, datotečni sustav, grafičko sučelje (ako nije do kraja poznato, dopuniti u 2. ciklusu)).*

## *Dijagram razreda s opisom*

*Potpoglavlje treba sadržavati dijagram razreda, jedan ili više njih, s opisima šta koji razred znači. Ono što je potrebno imati u dokumentaciji prilikom prve predaje je: svi razredi specifični za vaš projekt, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazive atributa razreda, veze i odnosi između razreda, okviri (forme) grafičkog sučelja specifične za pojedini programski jezik, ako su vam poznate.*

*Primjer onoga što nije potrebno imati u dokumentaciji prilikom druge predaje su: tipovi parametara metoda razreda (npr. boolean, int), događaji (npr. klikovi miša).*

*Ako za neke stvari niste sigurni kako će biti implementirane, tada je bolje taj dio izostaviti prilikom prve predaje pa dopuniti na drugoj predaji. Prilikom druge predaje navest ćete sve one razrede sa svim detaljima koje direktno koristite u implementaciji sustava.*

## *Dijagram objekata*

*Prikazati stanje sustava u određenom trenutku. Dio je Revizije 1.*

## *Ostali UML dijagrami*

*Ovdje počinju sadržaji Revizije 2.*

*U ovom potpoglavlju potrebno je uvrstiti dodatna četiri dijagrama koji prikazuju arhitekturu sustava i to: komunikacijski dijagram, dijagram stanja, dijagam aktivnosti i dijagram komponenti. Potrebno je dati barem po jedan primjerak svakog od dijagrama. Potrebno je da svaki od dijagrama prikazuje (jedan) bitan dio funkcionalnosti sustava. Dijagram komponenti treba prikazivati sve komponente sustava. Prema potrebi, ovdje se mogu dodati i neki dodatni sekvencijski dijagrami za objekte.*

# **Implementacija i korisničko sučelje**

*U ovom poglavlju potrebno je:*

*dati dijagram razmještaja (deployment dijagram)*

*navesti koje su tehnologije i alati korišteni u razvoju sustava*

*dati isječak programskog koda koji implementira neku od temeljnih funkcionalnosti u sustavu*

*objasniti kako je ispitano implementirano rješenje i pokazati bar 4 ispitna scenarija*

*dati upute za instalaciju*

*dati upute za korištenje*

## *Dijagram razmještaja*

*Potrebno je umetnuti dijagram razmještaja i po potrebi ga opisati.*

## *Korištene tehnologije i alati*

*Navesti sve tehnologije i alate koji su primijenjeni pri izradi projekta te ih ukratko opisati; njihovo značenje i mjesto i način primjene.*

## *Isječak programskog koda vezan za temeljnu funkcionalnost sustava*

*U ovom poglavlju potrebno je prikazati isječak programa koji prema mišljenju studenta ostvaruje temeljnu funkcionalnost u sustavu (ili nekom modulu).*

## *Ispitivanje programskog rješenja*

*U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti s prikazom odabira ispitnih slučajeva.*

*Prilikom prezentacije svojih Ispitnih scenarija (minimalno četiri) studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete. Poželjno je da se napravi i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane te da se vidi na koji način sustav reagira kada nešto nije u potpunosti ostvareno.*

*Različiti ulazi za ispitne scenarije trebaju pokrivati temeljnu funkcionalnost nekog modula i nekoliko rubnih uvjeta.*

## *Upute za instalaciju*

*U ovom poglavlju potrebno je dati upute za instalaciju ostvarenog prototipa.*

## *Korisničke upute*

*Korisničke upute ovisit će o količini implementirane funkcionalnosti. Očekuje se da upute budu na oko pet A4 stranica koje će dati potpuni opis funkcionalnosti sustava sa stajališta krajnjeg korisnika.*

# **Zaključak i budući rad**

*U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.*

# **Popis literature**

*Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.*

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Oblikovanje programske potpore, FER ZEMRIS, <http://www.zemris.fer.hr/predmeti/opp>
3. I. Sommerville, „Software engineering“, 8th ed, Addison Wesley, 2007.
4. T.C.Lethbridge, R.Langaniere, „Object-Oriented Software Engineering“, 2nd ed. McGraw-Hill, 2005.
5. Software engineering ,Rutgers University, <http://www.ece.rutgers.edu/~marsic/Teaching/SE>
6. I. Marsic, „Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
7. Concepts: Requirements, <http://www.upedu.org/upedu/process/gcncpt/co_req.htm>
8. UML 2 Class Diagram Guidelines, <http://www.agilemodeling.com/style/classDiagram.htm>
9. Domain Class Diagram Modeling Standards and Guidelines, <http://www.bced.gov.bc.ca/imb/downloads/classdiagramstandards.pdf>
10. Astah Community, <http://astah.net/editions/community/>

# **Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)**

*U ovom dijelu dodatka potrebno je ispisati (numerirane) liste slika, dijagrama, tablica, ispisa kôda i ostalih pomoćnih sadržaja iz teksta dokumentacije.*

# **Dodatak B: Dnevnik sastajanja**

*U ovom dijelu dodatka potrebno je prenijeti sadržaj dnevnika sastajanja, tj. datoteke 'dnevnik.txt' iz repozitorija grupe. Datoteku 'dnevnik.txt' potrebno je voditi kroz repozitorij, a ovdje će to biti preneseno kao dio konačne dokumentacije.*

# **Dodatak C: Prikaz aktivnosti grupe**

*U ovom dijelu dodatka potrebno je navesti*

* *popis članova grupe i njihovih zaduženja, tj. odrađenih poslova (u postocima ukupno odrađenog posla)*
* *prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je u* BitBucket *sučelju instalirati dodatak* AwesomeGraph *koji daje prikaz grafički prikaz aktivnosti grupe, tj. '*commit' *akcija tijekom trajanja projekta, te na kraju projekta generirane grafove prenijeti u ovo poglavlje dokumentacije.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Popis aktivnosti** | **Članovi grupe** (abecednim redom) | | | | | | |
| Ime Prezime voditelja | Ime Prezime 2 | Ime Prezime 3 | Ime Prezime 4 | Ime Prezime 5 | Ime Prezime 6 | Ime Prezime 7 |
| **Upravljanje projektom** |  |  |  |  |  |  |  |
| **Opis projektnog zadatka** |  |  |  |  |  |  |  |
| **Rječnik pojmova** |  |  |  |  |  |  |  |
| **Opis funkcionalnih zahtjeva** |  |  |  |  |  |  |  |
| **Opis ostalih zahtjeva** |  |  |  |  |  |  |  |
| **Arhitektura i dizajn sustava** |  | | | | | | |
| Svrha, opći prioriteti i skica sustava |  |  |  |  |  |  |  |
| Dijagram razreda s opisom |  | | | | | | |
| Dijagram objekata |  |  |  |  |  |  |  |
| Ostali UML dijagrami |  |  |  |  |  |  |  |
| **Implementacija i korisničko sučelje** |  | | | | | | |
| Dijagram razmještaja |  |  |  |  |  |  |  |
| Korištene tehnologije i alati |  |  |  |  |  |  |  |
| Isječak programskog kôda |  |  |  |  |  |  |  |
| Ispitivanje programskog rješenja |  |  |  |  |  |  |  |
| Upute za instalaciju |  |  |  |  |  |  |  |
| Korisničke upute |  |  |  |  |  |  |  |
| **Plan rada** |  |  |  |  |  |  |  |
| **Pregled rada i stanje ostvarenja** |  |  |  |  |  |  |  |
| **Zaključak i budući rad** |  |  |  |  |  |  |  |
| **Popis literature** |  |  |  |  |  |  |  |
| **Dodaci** |  | | | | | | |
| Indeks |  |  |  |  |  |  |  |
| Dnevnik sastajanja |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

*Napomena: Doprinose u aktivnostima treba navesti u postocima po članovima grupe. Zbroj postotaka u svakom retku treba biti 100%.*

Pregled pohrana kroz vrijeme trajanja projekta (primjer):



# **Dodatak D: Plan rada / Pregled rada i stanje ostvarenja**

*U ovom poglavlju potrebno je navesti:*

* *(u rev. 1) koji je plan rada za rev. 2,*
* *(u rev. 2) koji je status implementacije u odnosu na postavljene ciljeve, procjenu vremena dovršetka projekta (ako zadatak nije u potpunosti ispunjen), koje bi bile smjernice za daljnji rad kad bi se nastavilo s projektom te u čemu bi se sastojale buduće nadogradnje.*