



Projektni razvoj aplikacija

Predavanje 02



Timski rad

A zašto?

Zašto je dobro raditi u timu?

- Na individualnim projektima razvijate svoje tehnološke kompetencije, upravljanje vremenom, motivacijom
- U timu smo sposobniji, ostvarujemo više
 - Suma sposobnosti pojedinaca je manja od sposobnosti tima
- Važno je steći kompetencije rada u timu
- U svakoj grupi postoji osoba koja najviše potiče ostale
 - Nije dobro da ta osoba odradi većinu posla, jer je tako lakše

Savjeti

- Podijelite zadatke
 - U čemu je dobar svaki pojedinac?
- Redovito se nalazite i provjeravajte napravljeno
 - Pozitivna, poticajna, iskrena komunikacija
- Pomognite jedni drugima

Proces izrade softvera




The background features several abstract elements: a large orange circle in the upper left, a large orange semi-circle on the right containing the text, a pink square outline on the left, a pink dashed line in the lower left, and a pink line forming a corner in the upper right. A small pink circle is partially visible in the top right corner.

Proces prikupljanja zahtjeva i analize

Zahtjev

**Opis potrebe Naručitelja
i/ili Krajnjeg korisnika na
temelju kojeg izrađujemo
aplikaciju**



Proces prikupljanja
zahtjeva i analize



Istraživanje
zahtjeva

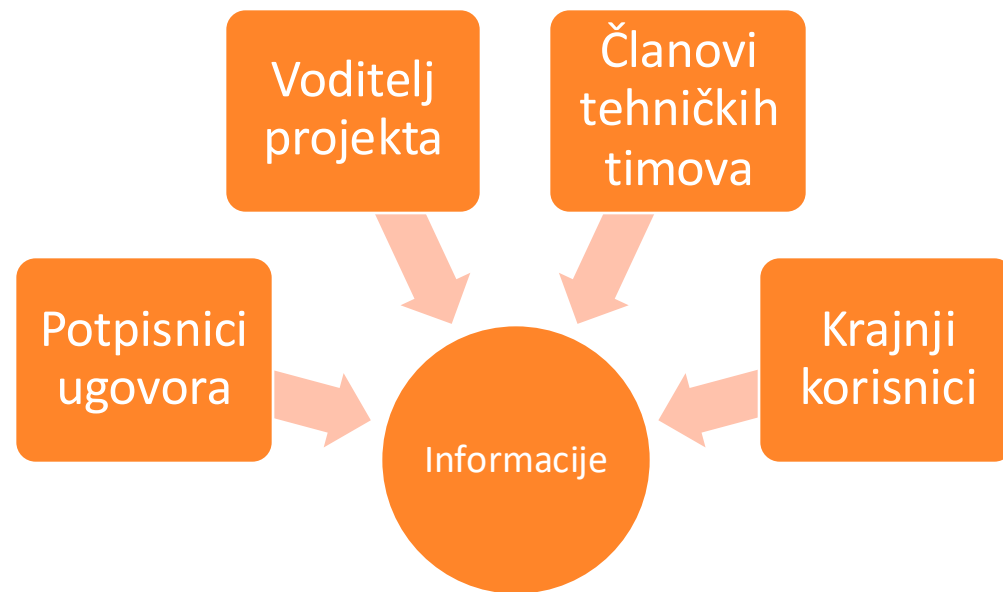
Istraživanje zahtjeva

- Na zajedničkom sastanku s
Naručiteljem /
Klijentom, Krajnjim
korisnicima
- Interaktivno






Od koga saznajemo informacije?



Istraživanje zahtjeva

- Naručitelj iznosi svoje zahtjeve (kako bi aplikacija trebala raditi)
- Naručitelji (uglavnom) nisu tehničke osobe
 - Ograničeno znanje o izgradnji softvera – zato trebaju nas
- Jasno definirati kako aplikacija mora funkcionirati je izazovan zadatak za Naručitelja



POTREBE su stvari ili
usluge koje morate imati
da biste živjeli.

ŽELJE su stvari ili usluge
koje biste voljeli imati, ali
možete živjeti bez njih.



WANTS AND NEEDS
CAN YOUR CHILD TELL THE DIFFERENCE?

Želje vs. poslovne potrebe

- Korisnik kaže što želi
 - Nekad i vrlo detaljno
- Naš krajnji cilj:
 - Osmisliti aplikaciju koja zadovoljava potrebe Naručitelja i Krajnjeg korisnika
- Naš zadatak:
 - **Razumjeti** poslovne potrebe
 - **Saznati** koje zadatke će aplikacijom rješavati

Što?

Zašto?

Kako?


Želje vs. poslovne potrebe

- Poslovna potreba
 - Osnovna funkcionalnost koju aplikacija mora imati kako bi svrha te aplikacije bila ispunjena
 - Sve mora biti izvedeno
- Želja
 - Može biti korisna, može dati aplikaciji dodatnu vrijednost
 - Nije nužno izvesti, ali je dobrodošlo
- Kako znati?
 - Rasprava
 - Može dovesti do promjene naručiteljeve vizije te aplikacije

Važno

- Cijelo vrijeme se moramo brinuti:
 - da je aplikacija izvediva
 - da su Naručiteljeva očekivanja realna
- „*Ne biti gljiva*”
 - Ne samo pasivno postavljati pitanja
 - Aktivno sudjelovati u raspravi





Proces prikupljanja zahtjeva i analize



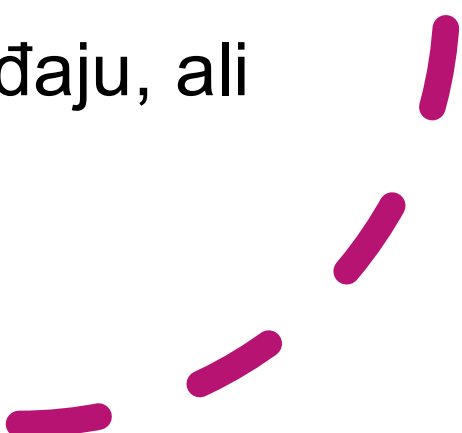
Bilježenje zahtjeva

- Za vrijeme sastanka s Naručiteljem bilježimo natuknice, crtamo skice, ...
- Nakon sastanka to pretvaramo u specifikaciju funkcionalnosti
- Format ovisi o dogovoru u timu

Proces prikupljanja zahtjeva i analize



Definiranje prioriteta

1. Važno je odrediti koje sve funkcionalnosti i značajke aplikacija **mora** imati kako bi bila upotrebljiva?
 2. Koji zahtjeve **trebamo** implementirati, ali **ne nužno odmah**, jer postoji neki drugi način na koji to korisnik može izvesti?
 3. Koje sva zahtjeve možemo promatrati kao **poboljšanja sustava**, ali **nisu nužni** za funkcioniranje?
 4. Koji se zahtjevi korisniku sviđaju, ali misli da ih **nije potrebno implementirati**?
- 

Nije sve
jednako važno

Važnost zahtjeva

Regular


- Istaknuto na sastancima
- Ako je implementirano, korisnik je zadovoljan

Nužno

- Često se podrazumijeva
- Ako toga nema – korisnik je nezadovoljan

Ekstra

- Ako to postoji, korisnik izuzetno zadovoljan
- Oprez



Proces prikupljanja zahtjeva i analize



Analiza zahtjeva



Konzistentnost

Potpunost

Jasnoća

- **Cilj:** Želimo dobru aplikaciju
- Analiziramo **cijelo vrijeme**
- Hoćemo li imati problema s nekim zahtjevima?
- Jesu li u kontradikciji?
- Jesu li dovoljno jasni?
- Je li sve definirano?



Sadržaj zahtjeva

- Definirano:
 - **Svi zadaci** i rezultati zadataka
 - **Sve mogućnosti unosa** podatka
 - Izvor, točnost podataka, opseg vrijednosti?
 - **Sve mogućnosti “izlaza”** (engl. output)
 - Odredište, točnost, opseg vrijednosti
 - **Formati** izvještaja
 - **Sve eksterne komponente** definirane (softver, hardver)
 - Sva sučelja za komunikaciju
 - Definirano **očekivano vrijeme odziva** za sve korisničke akcije
 - Definirano vrijeme za operacije
 - **Maksimalno zauzeće memorije**, diskovnog prostora

Stalna je samo promjena

- Ne čini se zahtjevnim
- „Korisnik zna što želi”
- Korisnik će se predomisлити tijekom vremena
- „Znam da misliš da si razumio što želim, ali, ono što sam rekao, nije ono što zapravo želim”



Stalna je samo promjena

- Dio procesa koji ima ogroman utjecaj na ostatak projekta
- Potreba da se započne s razvojem prije nego razumijemo potpunu sliku
- „Korisnici ne mogu shvatiti što žele dok ne vide prve iteracije gotove aplikacije”
- „Zahtjevi se mijenjaju tako brzo, da nema potrebe da sve razjasnimo”
- Aplikacija koja radi je najvažnija
- Sve od navedenog stoji

Zadaci

1. Razumjeti što Korisnik želi
2. Analiza
3. Provjera izvedivosti
4. Pregovaranje o načinu izvođenja
5. Nedvosmislena specifikacija
6. Potvrda
7. Upravljanje zahtjevima tijekom projekta

Tehnike prikupljanja zahtjeva

Čitanje
dokumentacije

Intervjui

Ankete

Sastanci

Prikupljanje zahtjeva

- Pitanja:
 - Što se aplikacijom želi postići
 - Kako se aplikacija uklapa u ostatak sustava
 - Kako će se aplikacija koristiti u svakodnevnom poslu
- Problemi:
 - Opseg
 - Problemi u razumijevanju
 - Nestalnost zahtjeva



Problemi

Korisnik i/ili
izvođač ne znaju
što je potrebno

Problemi u
komunikaciji

Izostavljanje
„očitih”
informacija

Konflikti između
korisnika

Dvosmisleni
zahtjevi

Zahtjevi koje
nije moguće
testirati

Problemi

- Saznajemo informacije od mnogih uloga
 - Mogu biti oprečne
- „Ovo nam niste definirali, pa nismo napravili”
„Ali, to se podrazumijeva”
- Ograničene mogućnosti promatranja
- Ne osjećaju se sigurno / ne žele reći



Razrada :: ponašanje i informacije

- Saznavanje **detalja** načina korištenja
- Koji entiteti postoje?
- Za svaki:
 - Koje attribute ima
 - Koji su akcije moguće
- Kakvi su odnosi među entitetima



Kako krenuti?

Prva pitanja

- Tko je **pokrenuo** razvoj ove aplikacije?
- Tko će ju **koristiti**? Tko su krajnji korisnici?
- Kakvu **ekonomsku korist** planirate ostvariti?
- Koje **probleme** planirate ovom aplikacijom riješiti?
- Imate li / postoje li neka **ograničenja** koja definiraju način izvedbe?
- Možemo li vaš **odgovor** smatrati **službenim**?
- Jesu li pitanja koja vam postavljamo **relevantna** za problem koji rješavamo?
- Postavljamo li **previše** pitanja?
- Možemo li **od još nekoga** dobiti dodatne informacije?

Organiziranje prikupljanja zahtjeva

Organizacija
sastan(a)ka

Pravila
pripreme za
sastanak

Agenda

Voditelj
rasprave

Način
prikupljanja

The background features several abstract geometric elements: a large orange circle in the upper left, a large orange semi-circle on the right side, a pink square outline on the left, a pink dashed line in the lower left, and a pink line forming a right angle in the upper center. A small pink circle is partially visible in the top right corner.

Tipovi zahtjeva

Tipovi zahtjeva



Zahtjevi vezani uz
funkcionalnost

Zahtjevi vezani uz
karakteristike

Poslovni zahtjev

- Odgovara na pitanje koja je svrha projekta na kojem radimo
- Zašto izrađujemo ovu aplikaciju
- Što Korisnik planira ostvariti nakon što aplikaciju počne koristiti
- Kako glasi poslovni zahtjev za aplikaciju koju mi izrađujemo u projektu PRA



Korisnički zahtjevi

- Tko su Krajnji korisnici (Korisnici) aplikacije?
 - Oni koji će aplikaciju koristiti kad uđe u produkciju
- Koje zadatke / aktivnosti će oni obavljati korištenjem aplikacije?
- Prikupljanje korisničkih zahtjeva
 - Jako važno
 - Vremenski zahtjevno
- Različiti načini bilježenja
 - Scenarij korištenja
 - Korisničke priče

Zahtjevi vezani uz funkcionalnost

- Ključni, gradivni elementi softverskog projekta
1. Mogućnosti koje aplikacija nudi
 - Primjer: izmjena podataka
 2. Kako aplikacija reagira u ovisnosti o tome koju je akciju korisnik napravio
 - Primjer: ispravan / neispravan unos
 3. Kako se aplikacija ne ponaša
 - Primjer: uneseni podaci se ne gube

Zahtjevi vezani uz funkcionalnost

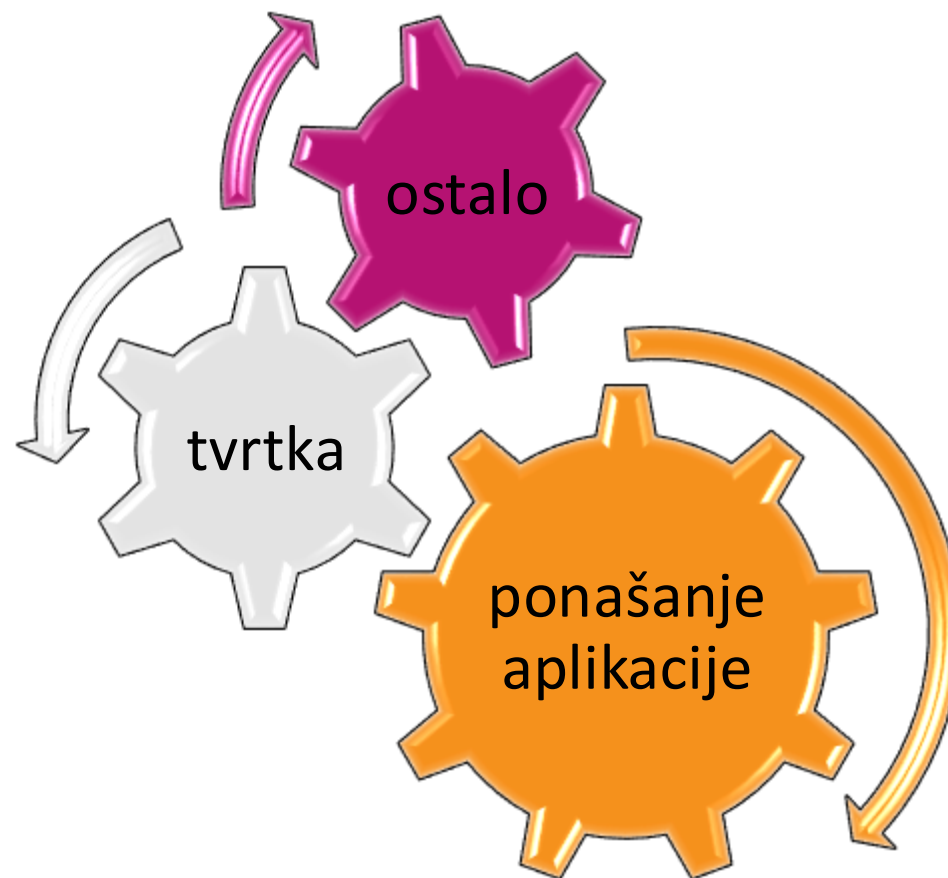
Koje mogućnosti aplikacija nudi?	<p>Aplikacija će studentima nuditi mogućnost izmjene osobnih podataka.</p> <p>Osobni podaci studenta unose se pri upisu i oni se u biti kasnije bitno ne mijenjaju. Ipak, studentima se nudi mogućnost da daju zahtjev za izmjenu osobnih podataka.</p> <p>Za izmjenu nekih podataka nije potrebno slati zahtjev, već se oni mijenjaju odmah nakon što student potvrdi izmjenu.</p>
Kako aplikacija reagira ovisno o tome koju je akciju korisnik napravio?	<p>Ako student sve promijenjene podatke ispuni ispravno, dio podataka šalje se studentskoj službi na provjeru, a ostatak se odmah mijenja.</p> <p>Ako student bilo koji od promijenjenih podataka ne unese u potpunosti ispravno, na ekranu mu se ispisuje gdje je pogriješio, a moguće radnje koje mu aplikacija nudi su ispravljanje podatka i ponovno slanje ili odustajanje od cjelokupne akcije.</p>
Kako se aplikacija ne ponaša?	<p>Bitno je napomenuti da se u ovom drugom slučaju, kada je došlo do greške u nekom podatku, svi ispravno uneseni podaci pamte i vidljivi su u formi za izmjenu. Na taj se način ne može dogoditi da student zbog nekog krivo unesenog podatka mora ponovno ispunjavati cijelu formu.</p>

Zahtjevi vezani uz značajke aplikacije

- Zahtjevi koji nisu vezani uz funkcionalnost
 - Pouzdanost
 - Sigurnost
 - Upotrebljivost
 - Učinkovitost
 - Performanse
 - Jednostavnost održavanja
- *Primjer: „Proces odobravanja kredita mora se moći završiti unutar 5 minuta”*

Zahtjevi vezani uz značajke aplikacije

- Primjeri:
 - brzina odziva aplikacije,
 - broj korisnika koji istovremeno mogu koristiti aplikaciju,
 - količina diskovnog prostora koji aplikacija zauzima,
 - hardverska platforma na kojoj se aplikacija izvršava,
 - tehnologija koju aplikacija koristi i sl.

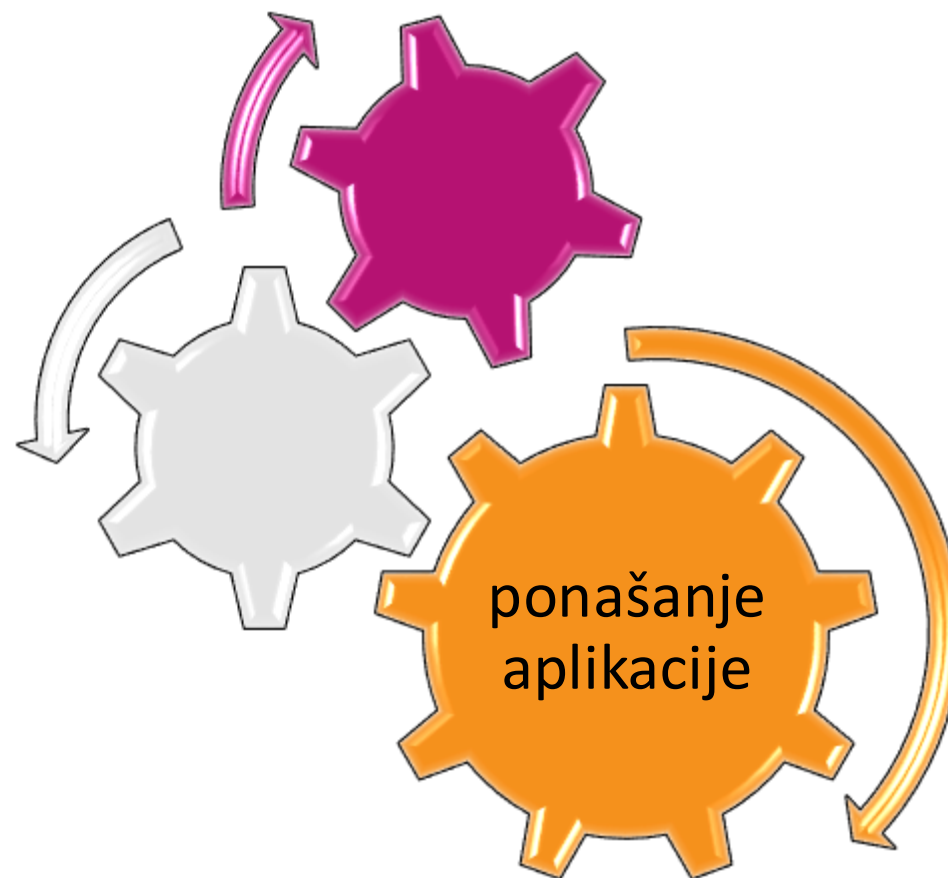


Zahtjevi vezani uz značajke aplikacije

- Odnose se na sustav u cjelini – **KRITIČNO!**
- *Primjer:*
 - Pretpostavimo da je naručitelj za primjer iz uvoda među ostalim dao sljedeći zahtjev vezan za funkcionalnost: „**Omogućiti studentima da mogu razmjenjivati multimedijalne sadržaje**“ te jedan zahtjev vezan za značajke aplikacije: „**Aplikacija mora imati odziv ispod sekunde kada joj istodobno pristupa 500 studenata**“.
 - Ako ne implementirate prvi zahtjev vezan za funkcionalnost, **aplikacija neće biti potpuna**, ali će biti upotrebljiva. No ako ne zadovoljite drugi zahtjev, **aplikacija neće biti upotrebljiva**

Zahtjevi vezani uz ponašanje aplikacije

- Jednostavnost korištenja
- Brzina izvođenja
- Memorijski zahtjevi
- Robusnost aplikacije
- Pouzdanost aplikacije



Zahtjevi vezani uz ponašanje aplikacije

Jednostavnost korištenja

Aplikacija mora biti izvedena tako da se korištenje aplikacije može savladati u okviru jednodnevnog treninga.

Brzina izvođenja

Aplikacija mora imati odziv ispod sekunde kada joj istodobno pristupa 500 studenata.

Memorijski zahtjevi

Zahtjevi aplikacije za memorijom neće prelaziti 4GB RAM-a.

Robusnost aplikacije

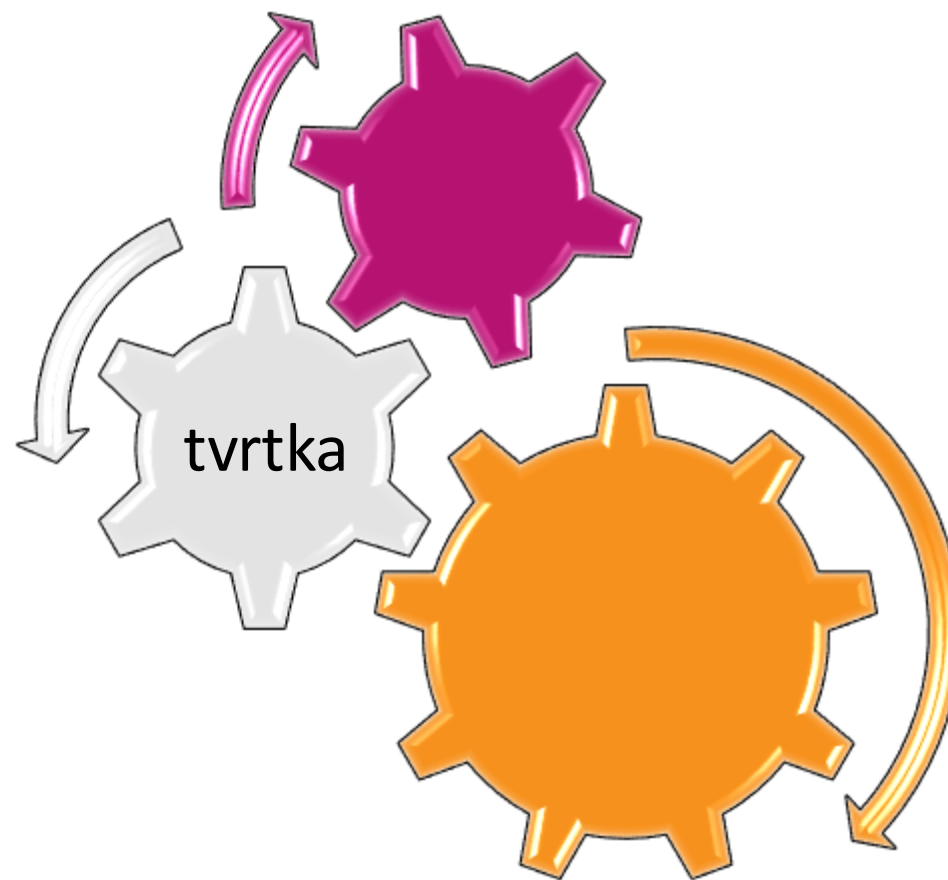
Vrijeme ponovne uspostave normalnog rada aplikacije nakon ispadanja iz aktivnosti zbog greške je maksimalno 5 minuta.

Pouzdanost aplikacije

Broj padova sustava neće prelaziti 3 u okviru tjedan dana.

Zahtjevi vezani uz tvrtku koja naručuje aplikaciju

- Tehnologija i proizvodi
- Objekti isporuke
- Industrijski standardi



Zahtjevi vezani uz tvrtku koja naručuje aplikaciju

Tehnologija i proizvodi

Aplikacija će se razvijati na .NET tehnologiji. Koristit će se C# programski jezik i SQL Server baza podataka.

Objekti isporuke

Na kraju projekta, izvođač će naručitelju isporučiti:

- funkcionalnu aplikaciju,
- potpuni kôd aplikacije (čiji je vlasnik naručitelj),
- dokumentaciju kôda i aplikacije u produkciji.

Industrijski standardi

Modul aplikacije za plaćanje školarine omogućuje, među ostalim načinima plaćanja, i plaćanje kreditnim karticama. Taj modul mora zadovoljavati PCI DSS standard (*).

(*) PCI DSS je skraćenica koja dolazi od: Payment Card Industry Data Security Standard definiranom od strane PCI Vijeća za sigurnosne standarde, a govori o sigurnosnim procedurama koje trebaju poštovati oni koji nude uslugu plaćanja kreditnim karticama na internetu. Više informacija na: www.pcisecuritystandards.org

Opis sučelja

- Odgovara na pitanje kako se aplikacija uklapa u ostatak svijeta
- Kojim sučeljima komunicira s vanjskim sustavima
- Primjer: Kako točno izgleda način komunikacije aplikacije za odobravanje kredita sa vanjskim sustavom – centralnim sustavom kreditnih obveznika
 - Koji protokol koristimo za komunikaciju?
 - Koje podatke prima i u kojem formatu?
 - Koje podatke vraća i u kojem formatu?

Tipovi zahtjeva



Poslovni zahtjev

Korisnički zahtjevi

Zahtjevi vezani uz funkcionalnost

Zahtjevi vezani uz karakteristike

Opis sučelja

Provjera ispunjenost zahtjeva

- Zahtjevi vezani za funkcionalnost
 - Jednostavno
- Zahtjevi vezani uz značajke
 - Kompleksnije
 - Loši primjeri opisa aplikacije ili rezultata projekta
 - jednostavna za korištenje
 - brza
 - neće trošiti previše memorije
 - vrijeme ponovne uspostave normalnog rada - minimalno
 - broj padova sustava - minimalan
 - isporučiti sve potrebne uratke isporuke

Provjera ispunjenost zahtjeva

Aplikacija mora biti jednostavna za korištenje.

Aplikacija mora biti brza.

Aplikacija neće trošiti previše memorije.

Vrijeme ponovne uspostave normalnog rada aplikacije nakon ispadanja iz aktivnosti zbog greške je minimalno.

Broj padova sustava bit će minimalan.

Na kraju projekta, izvođač će naručitelju isporučiti sve potrebne uratke isporuke.

Aplikacija mora izvedena tako da se korištenje aplikacije može savladati u okviru jednodnevnog treninga.

Aplikacija mora imati odziv ispod sekunde kada joj istodobno pristupa 500 studenata.

Zahtjevi aplikacije za memorijom neće prelaziti 4GB RAM-a.

Vrijeme ponovne uspostave normalnog rada aplikacije nakon ispadanja iz aktivnosti zbog greške je maksimalno 5 minuta.

Broj padova sustava neće prelaziti 3 u okviru tjedan dana.

Na kraju projekta, izvođač će naručitelju isporučiti:

- funkcionalnu aplikaciju,
- potpuni kôd aplikacije (čiji je vlasnik naručitelj),
- dokumentaciju kôda i aplikacije u produkciji.

Na kraju projekta, izvođač će naručitelju isporučiti sve potrebne uratke isporuke u skladu s Pravilnikom tvrtke naručitelja koji je prilog ovom dokumentu.

Provjera ispunjenost zahtjeva

Kuda sa zahtjevima koje ne možemo provjeriti?

Korisnički zahtjevi i zahtjevi sustava

Korisnički zahtjev

Aplikacija treba omogućiti prijavu ispita svim studentima koju su prijavljeni u aplikaciju, te imaju upisanu godinu.

Zahtjev sustava

Nakon što je definiran rok za neki predmet i unesen u sustav, studenti se prijavljuju za taj ispitni rok preko alata „Prijava“ koji se nalazi u glavnom izborniku sa lijeve strane aplikacije.

Odabirom toga alata studentu se na ekranu prikazuje tablica s predmetima koje može prijaviti.

Predmeti su grupirani abecednim redom. U plavim okvirima nalaze se predmeti u koje je student upisan, dok se u žutim okvirima nalaze dijelovi tog predmeta, naravno ako ih predmet ima.

Neki predmet može biti podijeljen na dijelove koji su zasebne cjeline i tako se predaju, a nastavu može izvoditi više nastavnika svaki za svoj dio.

Na kraju kada student položi svaki od dijelova predmeta dobiva ukupnu ocjenu (samo jedna ocjena). Ovakvih predmeta ima iznimno malo, ali je svejedno potrebno predvidjeti i izvesti i prikaz takvih predmeta.

Podjela na korisničke zahtjeva i zahtjeve sustava

Korisnički zahtjevi

- Kratke izjave
- Vanjsko ponašanje sustava
- Jednostavan, lako razumljiv jezik
- Oprez: nedvosmisleno
- Namijenjen ljudima koji ne moraju nužno imati tehničko znanje

Zahtjevi sustava

- Upareni s korisničkim zahtjevima
- Ne trebaju sadržavati previše detalja oko izvedbe
- Što, ne kako
- Dijagrami, skice
- Ostatak poslovnog sustava

Namjena korisničkih zahtjeva i zahtjeva sustava

Korisnički zahtjevi

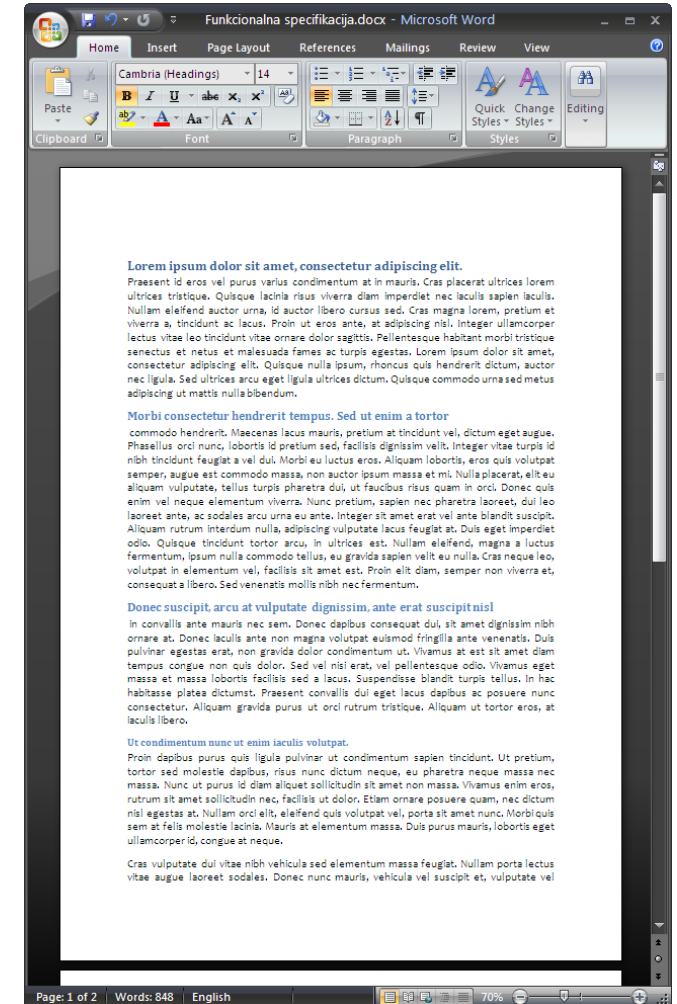
- osobama koje potpisuju ugovor
- voditeljima projekta
- krajnjim korisnicima
- osobama iz tehničkog tima
- arhitektima sustava

Zahtjevi sustava

- voditeljima projekta
- krajnjim korisnicima
- osobama iz tehničkog tima
- arhitektima sustava
- razvojnim inženjerima
- inženjerima koji provode testiranje

Dokument s popisom zahtjeva aplikacije

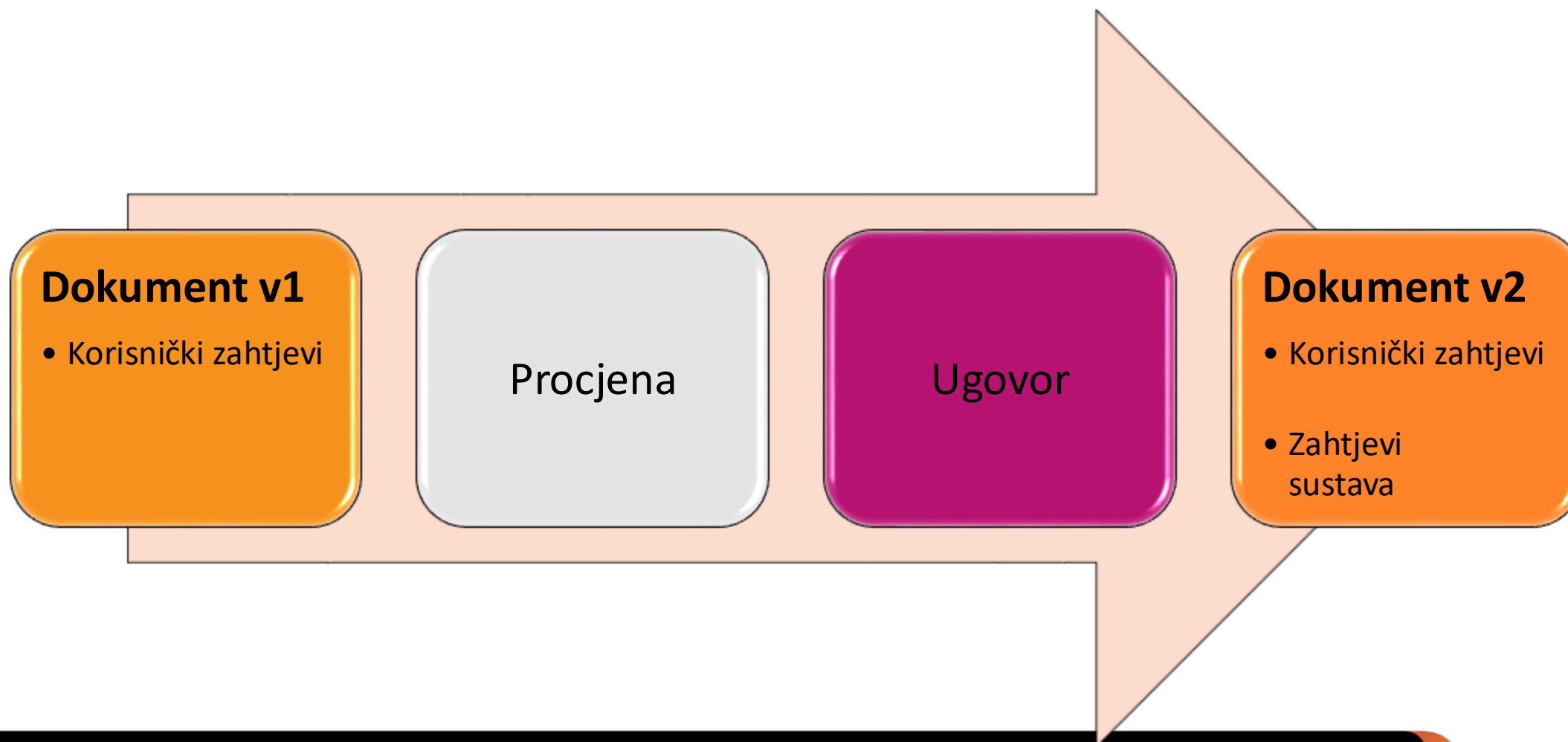
- Sadrži potpuni opis aplikacije
 - Zahtjevi vezani za funkcionalnost
 - Zahtjevi vezani za značajke
- Opisuje ponašanje aplikacije
- Ne opisuje način implementacije
- Nazivi:
 - Funkcionalna specifikacija aplikacije
 - Specifikacija funkcionalnosti aplikacije



Definicije

- Ugovor
 - Pravni dokument oko kojeg su se složili Kupac i Dobavljač
 - Uključuje tehničke zahtjeve, cijenu i rokove
- Kupac
 - Tvrtka / Osoba koja plaća proizvod ili aplikaciju
 - Ne definira nužno zahtjeve
- Dobavljač
 - Izvoditelj radova
- Korisnik
 - Osoba / osobe koje će koristiti aplikaciju

Tijek izrade dokumenta



Razlozi za formalnu specifikaciju

- Korisnik određuje funkcionalnost umjesto programera
 - Ako su zapisani, korisnik ih može provjeriti, ovjeriti
 - Ako nisu, onda programer smišlja zahtjeve tijekom programiranja
- Izbjegavanje neslaganja
 - Određuje se opseg projekta – prije programiranja
- Smanjenje izmjena nakon što razvoj počne
 - Bug vs. Izmjena zahtjeva

Mit :: Zahtjevi uklesani u kamen

- Iluzija
 - Miran put 😊
 - Osmišljavanje arhitekture > programiranje > testiranje > produkcija
 - Lijepo je razmišljati da nakon potpisivanja dokumenta neće biti promjena 😊
- Stvarnost
 - Korisnik ne može pouzdano definirati što mu treba prije nego to vidi
 - Kako razvoj odmiče, tako korisnik bolje razumije svoje potrebe
 - Slijepo praćenje zapisanog – ne slušanje korisnika
 - Koliko promjena je uobičajeno?

Kako se nositi s promjenama?

- Ako dokument nije dobro napisan – STOP, korak nazad
- Svi moraju biti svjesni cijene izmjene(a)
 - Izjave: “napraviti novi raspored”, “izvijestiti o trošku”
 - Hladan tuš: Moramo to imati > Bilo bi zgodno imati
- Uvođenje procedure izmjene
- Način rada koji podržava promjene
 - Kraći razvojni ciklusi
- Odustajanje od projekta
 - Ako su zahtjevi loše definirani ili podložni velikim izmjenama

Dovršenost zahtjeva

- Ako neki zahtjev nije moguće odmah definirati, da li je to jasno zapisano?
- Jesu li zapisani svi zahtjevi?
 - Ako su svi zadovoljeni, da li je proizvod / rješenje prihvatljivo?
- Udovoljavanje **jačemu**
 - Korisnik, šef
 - Da li su svi zahtjevi izvedivi?

Kvaliteta zahtjeva

- Jesu li svi zahtjevi zapisani na način da ih korisnik razumije?
 - Da li i korisnik to misli? 😊
- Postoji li nekonzistentnost u dokumentu?
- Da li zahtjevi definiraju samo što treba biti izvedeno, a ne i način izvedbe?
- Da li je razina detalja konzistentna?
 - Neujednačenost u detaljima opisa
- Može li se dokument predati nekom drugom?
- Je li moguće provjeriti (testirati) zahtjeve?

Osnovne pretpostavke dokumenta

- Specifikacija softverskog proizvoda, programa ili skupa programa koji zajedno obavljaju neku funkciju
- Pišu ga: Kupac ili/i Dobavljač
- Treba uključivati:
 - **Funkcionalnost**
 - Što softver treba raditi
 - **Vanjska sučelja**
 - Kako softver treba komunicirati s ljudima, hardverom i ostalim softverom
 - **Karakteristike aplikacije**
 - Brzina izvođenja, vrijeme odziva, dostupnost, vrijeme oporavka od grešaka
 - **Prenosivost, održavanje, sigurnost**
 - **Ograničenja:** jezik, baza, operativni sustav

Značajke kvalitetne specifikacije

- Točnost
 - Svaki zahtjev iz specifikacije treba biti ispunjen
 - Mora se gledati širi kontekst (svi dokumenti projekta, specifikacija čitavog sustava)
- Nedvosmislenost
 - Ako svaka izjava ima samo jednu interpretaciju
 - Ako je jasno što pojedini izraz znači; ako nije navesti u popisu pojmova
 - Autorima i onima koji će koristiti specifikaciju (dizajn, implementacija, praćenje izvođenja, verifikacija, edukacija)
 - Paziti da svi razumiju

Značajke kvalitetne specifikacije

- Potpunost
 - Svi značajni zahtjevi (funkcionalnost, značajke)
 - Mogući odgovori softvera (za sve tipove ulaznih podataka u svim situacijama) – ispravno i neispravno
 - „Definirati će se kasnije”
 - Zašto odgovor nije poznat već sada?
 - Što mora biti zadovoljeno da bi se znalo što točno treba biti implementirano?
 - Tko je zadužen za to?
 - Kada će se to znati?

Značajke kvalitetne specifikacije

- Dosljednost
 - Potvrditi da ne postoje zahtjevi u kojima su opisane suprotne stvari
 - Primjeri:
 - Na jednom mjestu izvještaji u excel formatu, na drugom u xml formatu
 - Na jednom mjestu definirano da postoje dva ulazna podatka, na drugom da postoji mogućnost višestrukog unosa po dva podatka
 - Na jednom mjestu da iza koraka A uvijek slijedi B, a na drugom da se mogu odvijati paralelno
 - Dva zahtjeva koriste za opis isti objekt iz stvarnog svijeta, ali koriste različite riječi

Značajke kvalitetne specifikacije

- Oznake važnosti

- Nije sve jednako važno
- Nužno
 - Softver nije prihvatljiv ako to nije izvedeno
- Poželjno
 - Značajno podiže kvalitetu softvera, ali nije neprihvatljiv ako ga nema
- Neobavezno
 - Nešto na čemu Dobavljač može premašiti očekivanja
- Je li se Kupac izjasnio oko svakog zahtjeva
 - Proces obično otkrije skrivene zahtjeve
- Jesu li programeri komentirali svaki zahtjev po pitanju složenosti

- Oznake stabilnosti zahtjeva

- Broj očekivanih promjena na zahtjevu
 - Na temelju budućih događanja koje se tiču tvrtke, ljudi, ...

Značajke kvalitetne specifikacije

- Mogući pogledi
 - Objekti
 - Objekti iz stvarnog svijeta, njihova svojstva i metode (servisi)
 - Procesi
 - Hijerarhija raznih funkcija – podatkovnih tokova
 - Ponašanja
 - Vanjsko ponašanje sustava
- Ovisno o prirodi projekta
- Korištenje prirodnog jezika da se olakša kupcu

Preporuke za pisanje dokumenta

- Standardan format
 - Numerirani zahtjevi: naslov (korisnički zahtjev), opis (zahtjev sustava)
 - **Detaljno i dobro opisani**
- Konzistentan način izražavanja
 - Obavezno, opcionalno
- Bez informatičkog žargona
- Analiza na kraju