

Studijski program: Informatika

Predmet: Softverski Inženjering

Softverski sistem za organizaciju turnira vitezova i zmajeva

-Verifikacija i Validacija–

|  |  |
| --- | --- |
| Predmetni nastavnik: | Student: |
| Hrvoje Puškarić | Nikola Tepšić 071/2021, |

Kragujevac 2024.

# Uvod

Razvoj softverskih sistema je složen proces koji zahteva pažljivo planiranje, implementaciju i proveru kako bi se osigurala njihova ispravnost i funkcionalnost. U ovom seminaru ćemo analizirati proces verifikacije i validacije softverskog sistema za organizaciju turnira vitezova i zmajeva, koji razvijamo za našeg klijenta, princa Petra Pliskog.

Verifikacija i validacija su ključni delovi svakog softverskog projekta jer omogućavaju identifikaciju i ispravljanje grešaka pre nego što softver bude pušten u upotrebu. Verifikacija se fokusira na proveru da li softver ispunjava sve tehničke zahteve definisane u specifikacijama, dok je validacija usmerena na potvrđivanje da softver zadovoljava potrebe korisnika i da funkcioniše u realnim uslovima.

**Projekat:** Softverski sistem za organizaciju turnira vitezova i zmajeva

**Klijent:** Princ Petar Pliski

Princ Petar Pliski je mladi i ambiciozni princ koji sanja da postane vitez i osvoji srce princeze na turniru vitezova i zmajeva. Da bi ostvario svoj san, potreban mu je softver koji će mu pomoći da organizuje turnir, upravlja vitezovima i zmajevima, i postigne slavu na bojnom polju.

**Pozitivne osobine princa Petra:**

**Analitički:** Pažljivo je razmatrao rizike i koristi vezane za svoju ideju.

Strateški orijentisan: Razumeo je dugoročne ciljeve i želeo je da softver podrži te ciljeve.

**Negativne osobine princa Petra:**

**Vizionarski:** Ponekad je imao ideje koje su previše fantastične da bi bile ostvarive.

**Komunikativno izazovan:** Ponekad je bilo teško razumeti njegove zahteve i ideje zbog njegovog ekscentričnog stila komunikacije.

U ovom radu ćemo detaljno opisati procese verifikacije i validacije, metodologije koje ćemo koristiti, kao i rezultate koje očekujemo da postignemo kroz ove procese. Na kraju, cilj je da se osigura da softverski sistem bude pouzdan, efikasan i prilagođen potrebama princa Petra Pliskog.

# Opis projekta

**Projekat:** Softverski sistem za organizaciju turnira vitezova i zmajeva

**Klijent:** Princ Petar Pliski

## Opis projekta

Projekat podrazumeva razvoj softverskog sistema koji će omogućiti organizaciju turnira vitezova i zmajeva. Sistem treba da podržava različite aspekte organizacije turnira, uključujući registraciju učesnika (vitezova i zmajeva), planiranje borbi, praćenje rezultata, i dodeljivanje nagrada.

## Ciljevi projekta

Cilj projekta je da se princu Petru Pliskom pruži alat koji će mu pomoći da uspešno organizuje turnir i postane poznat po svojim viteškim poduhvatima. Softver treba da obezbedi:

* **Efikasnu organizaciju turnira:** Omogućavanje lakoće planiranja i upravljanja turnirom.
* **Upravljanje učesnicima:** Registracija, praćenje i upravljanje vitezovima i zmajevima koji učestvuju u turniru.
* **Praćenje rezultata:** Evidentiranje i praćenje rezultata borbi u realnom vremenu.
* **Dodela nagrada:** Automatsko dodeljivanje nagrada pobednicima.

## Funkcionalnosti sistema

**Registracija učesnika:**

* Mogućnost registracije vitezova i zmajeva.
* Evidencija osnovnih podataka o učesnicima.

**Planiranje borbi:**

* Automatizovano kreiranje parova za borbu.
* Raspored borbi i evidencija vremena i mesta održavanja.

**Praćenje rezultata:**

* Evidencija rezultata borbi u realnom vremenu.
* Ažuriranje statusa učesnika na osnovu rezultata.

**Dodela nagrada:**

* Automatsko određivanje pobednika na osnovu rezultata.
* Dodela i evidencija nagrada pobednicima.

## Klijent

**Princ Petar Pliski** je mladi princ sa velikim ambicijama. On sanja o tome da postane vitez i osvoji srce princeze kroz svoje poduhvate na turniru. Princ Petar je osoba sa sledećim karakteristikama:

**Pozitivne osobine:**

* **Analitički:** Princ pažljivo razmatra rizike i koristi vezane za svoje ideje.
* **Strateški orijentisan:** Ima dugoročne ciljeve i želi da softver podrži te ciljeve.

**Negativne osobine:**

* **Vizionarski:** Ponekad ima ideje koje su previše fantastične da bi bile ostvarive.
* **Komunikativno izazovan:** Njegov ekscentrični stil komunikacije ponekad otežava razumevanje njegovih zahteva i ideja.

## Tehnički zahtevi

Da bi sistem bio funkcionalan i koristan za princa Petra, potrebno je da ispunjava sledeće tehničke zahteve:

* **Funkcionalni zahtevi:** Registracija učesnika, planiranje borbi, praćenje rezultata, dodela nagrada.
* **Nefunkcionalni zahtevi:** Sigurnost podataka, jednostavnost upotrebe, skalabilnost sistema, performanse.

## Izazovi u komunikaciji

Jedan od ključnih izazova u ovom projektu je komunikacija sa klijentom. Princ Petar, zbog svog ekscentričnog stila, ponekad ima poteškoća da jasno definiše svoje zahteve. Zbog toga je važno koristiti jasne metode za prikupljanje zahteva, kao što su upitnici, intervjui, i redovni sastanci sa klijentom kako bi se osigurala pravilna interpretacija njegovih ideja.

# Verifikacija

Verifikacija je proces koji osigurava da softverski sistem ispunjava sve tehničke zahteve koji su definisani u specifikacijama. Ovaj proces se sastoji od nekoliko koraka koji uključuju pregled dokumentacije, provere koda, i različite vrste testiranja. Cilj verifikacije je identifikacija i ispravljanje grešaka pre nego što softver bude pušten u upotrebu.

## Definisanje zahteva

Prvi korak u verifikaciji je jasno definisanje zahteva koje softverski sistem treba da ispuni. Zahtevi su podeljeni u dve glavne kategorije:

* **Funkcionalni zahtevi:** Ovi zahtevi opisuju specifične funkcionalnosti koje sistem mora imati.
  + Registracija učesnika (vitezova i zmajeva).
  + Planiranje i raspored borbi.
  + Praćenje rezultata borbi u realnom vremenu.
  + Dodela nagrada pobednicima.
* **Nefunkcionalni zahtevi:** Ovi zahtevi se odnose na performanse, sigurnost, skalabilnost i jednostavnost upotrebe sistema.
  + Sigurnost podataka učesnika.
  + Visoke performanse i odziv sistema.
  + Skalabilnost kako bi se mogao nositi sa velikim brojem učesnika i borbi.
  + Intuitivan i user-friendly interfejs.

## Provera ispunjenosti zahteva

Verifikacija zahteva se sprovodi kroz sledeće korake:

### Pregled dokumentacije

Analiza dokumentacije je ključna za osiguranje da su svi zahtevi jasno definisani i razumljivi. Ovo uključuje:

* **Specifikacija zahteva:** Dokument koji detaljno opisuje sve funkcionalne i nefunkcionalne zahteve.
* **Tehnička dokumentacija:** Opisi arhitekture sistema, dizajnerski obrasci i detalji implementacije.
* **Plan testiranja:** Dokument koji opisuje pristup testiranju, uključujući strategije, metodologije i alatke koje će se koristiti.

### Provera koda

Provera koda je ključni deo verifikacije i uključuje nekoliko metoda:

* **Kod review:** Detaljan pregled koda od strane tima developera kako bi se identifikovale greške, neefikasnosti i neslaganja sa specifikacijama.
* **Statistička analiza koda:** Korišćenje alata za analizu koda kako bi se automatski otkrili potencijalni problemi, poput sigurnosnih ranjivosti ili neefikasnog koda.

### Testiranje

Testiranje je glavna aktivnost u procesu verifikacije. Postoji nekoliko vrsta testova koje je potrebno sprovesti:

* **Jedinični testovi:** Testiranje pojedinačnih komponenti sistema kako bi se osiguralo da svaka funkcioniše ispravno. Ovo uključuje testiranje funkcija, klasa i modula sistema.
* **Integracioni testovi:** Testiranje kombinacija komponenti kako bi se osiguralo da rade zajedno bez problema. Ovo je važno za proveru interakcija između različitih delova sistema.
* **Sistem testovi:** Testiranje celokupnog sistema kako bi se osiguralo da sve funkcioniše kao celina. Ovo uključuje simulaciju stvarnih uslova korišćenja.
* **Prihvatni testovi:** Testovi koje sprovodi klijent kako bi se osiguralo da sistem zadovoljava njegove potrebe i zahteve.

## Alati za verifikaciju

Za proces verifikacije, koristićemo različite alate i tehnologije, uključujući:

* **JUnit:** Za jedinične testove u Java okruženju.
* **Selenium:** Za automatizaciju testiranja web interfejsa.
* **SonarQube:** Za statičku analizu koda i identifikaciju potencijalnih problema.
* **JIRA:** Za praćenje grešaka i upravljanje procesom testiranja.

## Rezultati verifikacije

Rezultati verifikacije uključuju detaljne izveštaje o svim otkrivenim greškama, njihovim uzrocima i koracima za ispravljanje. Na kraju verifikacije, softver bi trebalo da bude stabilan, bez kritičnih grešaka, i spreman za validaciju sa korisnicima.

# Validacija

Validacija je proces koji osigurava da softverski sistem zadovoljava potrebe i očekivanja krajnjih korisnika. Dok verifikacija potvrđuje da sistem funkcioniše prema tehničkim specifikacijama, validacija potvrđuje da sistem radi ispravno u realnim uslovima korišćenja i da je koristan za korisnike. Ovaj proces uključuje različite vrste testiranja i povratne informacije od korisnika.

## Testiranje sistema

Testiranje sistema je ključna komponenta validacije i obuhvata sledeće vrste testova:

### Jedinični testovi

Jedinični testovi testiraju pojedinačne komponente sistema kako bi se osiguralo da svaka funkcioniše ispravno. Ovi testovi pomažu u identifikaciji i ispravljanju grešaka u ranoj fazi razvoja.

* **Cilj:** Osigurati da svaka funkcionalna jedinica sistema radi kako je predviđeno.
* **Alati:** JUnit (za Java aplikacije), pytest (za Python aplikacije).

### Integracioni testovi

Integracioni testovi proveravaju kako različite komponente sistema rade zajedno. Ovo je važno za identifikaciju problema koji se javljaju kada komponente međusobno komuniciraju.

* **Cilj:** Osigurati da interakcije između komponenti funkcionišu ispravno.
* **Alati:** Selenium (za testiranje web interfejsa), Postman (za API testiranje).

### Sistem testovi

Sistem testovi proveravaju celokupan sistem u okruženju koje simulira stvarne uslove korišćenja. Ovo uključuje funkcionalna, nefunkcionalna, performansna i sigurnosna testiranja.

* **Cilj:** Osigurati da ceo sistem funkcioniše ispravno u svim aspektima.
* **Alati:** Selenium, LoadRunner (za performansno testiranje), OWASP ZAP (za sigurnosno testiranje).

### Prihvatni testovi

Prihvatni testovi su završni testovi koje obavljaju krajnji korisnici kako bi se osiguralo da sistem zadovoljava njihove potrebe i zahteve. Ovi testovi uključuju funkcionalno testiranje i evaluaciju korisničkog iskustva.

* **Cilj:** Potvrditi da sistem ispunjava očekivanja korisnika.
* **Alati:** Manualno testiranje od strane korisnika, korišćenje anketnih alata za povratne informacije.

## Validacija sa korisnicima

Validacija sa korisnicima je ključna za osiguranje da softver zadovoljava stvarne potrebe korisnika. Ovaj proces uključuje prikupljanje povratnih informacija i sprovođenje evaluacija u realnim uslovima korišćenja.

### Feedback sesije

Redovne sesije sa korisnicima omogućavaju prikupljanje povratnih informacija o funkcionalnosti i upotrebljivosti sistema. Tokom ovih sesija, korisnici mogu izneti svoja mišljenja, predloge i potencijalne probleme.

* **Cilj:** Razumeti kako korisnici doživljavaju sistem i identifikovati oblasti za poboljšanje.
* **Metode:** Intervjui, fokus grupe.

### Pilotska implementacija

Pilotska implementacija podrazumeva uvođenje sistema u kontrolisano okruženje sa ograničenim brojem korisnika. Ovo omogućava testiranje sistema u realnim uslovima i identifikaciju potencijalnih problema pre šire implementacije.

* **Cilj:** Testirati sistem u stvarnim uslovima korišćenja i prikupiti povratne informacije.
* **Metode:** Ograničena implementacija, praćenje performansi i korisničkog iskustva.

### Ankete i upitnici

Ankete i upitnici su korisni alati za prikupljanje povratnih informacija od velikog broja korisnika. Oni omogućavaju sistematsku evaluaciju različitih aspekata sistema, uključujući funkcionalnost, upotrebljivost i zadovoljstvo korisnika.

* **Cilj:** Prikupiti kvantitativne i kvalitativne podatke o iskustvima korisnika.
* **Alati:** Google Forms, SurveyMonkey.

## Rezultati validacije

Rezultati validacije uključuju analizu povratnih informacija korisnika, rezultate različitih testova i preporuke za poboljšanja. Na osnovu ovih rezultata, sistem se može prilagoditi kako bi bolje zadovoljio potrebe korisnika i ispravio identifikovane probleme.

# Zaključak

Procesi verifikacije i validacije su ključni za uspešan razvoj i implementaciju softverskog sistema za organizaciju turnira vitezova i zmajeva. Kroz ove procese osiguravamo da sistem ne samo da ispunjava tehničke zahteve, već i da zadovoljava potrebe krajnjih korisnika.

## Ključni nalazi

**Verifikacija:**

* Verifikacija zahteva uključivala je pažljivo pregledanje dokumentacije, koda i sprovođenje različitih vrsta testiranja. Ovi koraci osigurali su da sistem funkcioniše prema tehničkim specifikacijama.
* Upotreba alata kao što su JUnit, Selenium, SonarQube i JIRA pomogla je u otkrivanju i ispravljanju grešaka, te u poboljšanju kvaliteta koda i performansi sistema.

**Validacija:**

* Validacija sa korisnicima uključivala je testiranje sistema u realnim uslovima korišćenja, prikupljanje povratnih informacija putem feedback sesija, pilotske implementacije i anketa.
* Ove aktivnosti omogućile su identifikaciju problema u upotrebljivosti i funkcionalnosti sistema, što je dovelo do dodatnih prilagođavanja kako bi se bolje zadovoljile potrebe korisnika.

## Benefiti za klijenta

Implementacija ovih procesa omogućila je princu Petru Pliskom da dobije pouzdan i efikasan softverski sistem za organizaciju turnira. Ključni benefiti uključuju:

* **Povećana efikasnost:** Sistem omogućava jednostavnije planiranje i upravljanje turnirom, što štedi vreme i resurse.
* **Poboljšana upotrebljivost:** Intuitivan interfejs i prilagodbe na osnovu povratnih informacija korisnika osiguravaju pozitivno korisničko iskustvo.
* **Sigurnost podataka:** Implementacija sigurnosnih mera štiti podatke o učesnicima i rezultatima turnira.
* **Skalabilnost:** Sistem može da podrži veliki broj učesnika i borbi, omogućavajući rast i širenje turnira.

## Preporuke za budući rad

Iako je trenutni sistem uspešno razvijen i implementiran, postoje mogućnosti za dalja poboljšanja:

* **Dodatne funkcionalnosti:** Uvođenje novih funkcionalnosti kao što su opcije za online praćenje turnira, integracija sa društvenim mrežama i napredne analitike.
* **Kontinuirano testiranje:** Redovno sprovođenje testiranja i prikupljanje povratnih informacija kako bi se osigurala kontinuirana ispravnost i poboljšanje sistema.
* **Edukacija korisnika:** Organizovanje treninga i radionica za korisnike kako bi se osiguralo da maksimalno koriste sve funkcionalnosti sistema.

## Zaključak

Verifikacija i validacija su neophodni procesi koji osiguravaju da softverski sistemi funkcionišu ispravno i zadovoljavaju potrebe korisnika. Kroz pažljivo planiranje, sprovođenje testiranja i prikupljanje povratnih informacija, osigurali smo da sistem za organizaciju turnira vitezova i zmajeva bude stabilan, pouzdan i korisnicima prilagođen alat. Princ Petar Pliski sada može s poverenjem organizovati svoje turnire, znajući da ima softversku podršku koja će mu pomoći da postigne svoje ciljeve i ostvari viteške poduhvate.