LanTern

Learn for Term

Plan testiranja

Verzija 1.0

Istorija revizija

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 11.05.2021. | 1.0 | Inicijalna verzija. | Snežana |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

1. Ciljevi 4

2. Opseg 4

3. Reference 4

4. Zahtevi za testiranje 4

5. Strategije testiranja (Test Strategy) 6

5.1 Tipovi testiranja (Testing Types) 6

5.1.1 Testiranje podataka i integriteta baze podataka (Data and Database Integrity Testing) 6

5.1.2 Testiranje sistema (System Testing) 7

5.1.3 Testiranje poslovnog ciklusa (Business Cycle Testing) 7

5.1.4 Testiranje korisničkog interfejsa (User Interface Testing) 8

5.1.5 Testiranje performansi (Performance Testing) 8

5.1.6 Testiranje opterećenja (Load Testing) 9

5.1.7 Testiranje kritičnih slučajeva(Stress Testing) 9

5.1.8 Testiranje volumena (Volume Testing) 10

5.1.9 Testiranje sigurnosti i kontrole pristupa (Security and Access Control Testing) 10

5.1.10 Testiranje otkaza i oporavka (Failover / Recovery Testing) 11

5.1.11 Testiranje konfiguracije (Configuration Testing) 12

5.2 Alati 12

6. Resursi 12

6.1 Radnici 13

6.2 Sistem 14

7. Kontrolne tačke testiranja 14

8. Izlazni produkti testiranja 15

8.1 Model testiranja 15

9. Projektni zadaci 15

Plan Testiranja

# Ciljevi

Ovaj dokument sadrži opis plana testiranja **LanTern** *Web* platforme. Namena ovog dokumenta je da opiše način provere funkcionalnih i nefunkcionalnih zahteva, postavljenih pri realizaciji predložene aplikacije. Ciljevi ovog dokumenta su:

* Izdvojiti komponente sistema koje treba testirati na osnovu postojećih informacija o projektu.
* Izdvojiti zahteve testiranja na osnovu datih predloga (na višem nivou).
* Predložiti i opisati strategije testiranja koje će biti korišćene.
* Odrediti resurse koji će biti potrebni za testiranje i proceniti vreme i trud koji treba uložiti pri testiranju.
* Navesti dokumente koji će opisati rezultate testiranja.

# Opseg

Plan testiranja se odnosi na integraciju sistema i proveru izvršne verzije aplikacije **LanTern**.

Plan testiranja ne opisuje testiranje pojedinačnih komponenti sistema, jer se pretpostavlja se da je svaki od programera izvršio kompletno testiranje koda pre njegovog dodavanja u sistem, korišćenjem metode crne kutije.

Plan testiranja će se odnositi na:

* testiranje interfejsa prema različitim tipovima korisnika,
* testiranje komunikacije sa bazom podataka,
* testiranje komunikacije komponenti aplikacije,
* testiranje funkcionalnosti izvršne verzije aplikacije.

# Reference

Spisak korišćene literature:

1. LanTern – Vizija sistema, 3S-LanTern-02, V1.5, 2021, 3STeam
2. LanTern – Specifikacija zahteva, 3S-LanTern-04 V1.1, 2021, 3Steam
3. LanTern – Arhitekturni projekat, 3S-LanTern-05 V1.1, 2021, 3STeam

# Zahtevi za testiranje

U ovom odeljku biće navedeni oni elementi sistema koje je potrebno testirati (slučajevi korišćenja, funkcionalni i nefunkcionalni zahtevi). Ovde će biti opisano samo šta je potrebno testirati, dok će detalji o samim testovima biti navedeni kasnije u dokumentu **Test specifikacija**.

**Testiranje podataka i integriteta baze podataka (Data and Database Integrity Testing)**

Proveriti mogućnost pristupa bazi **LanTern**.

Proveriti mogućnost istovremenog pristupa istim podacima radi čitanja.

Proveriti tačnosti podataka koji se pribavljaju iz baze.

**Testiranje sistema – funkcionalno testiranje (System Testing – functional testing)**

Proveriti slučaj korišćenja Registrovanje na platformu.

Proveriti slučaj korišćenja Prijavljivanje na platformu.

Proveriti slučaj korišćenja Pregled oglasa po gradu.

Proveriti slučaj korišćenja Pregled oglasa sortiranih po ceni časa.

Proveriti slučaj korišćenja Pregled oglasa po kategoriji.

Proveriti slučaj korišćenja Čuvanje oglasa.

Proveriti slučaj korišćenja Uklanjanje oglasa iz sačuvanih oglasa.

Proveriti slučaj korišćenja Kontaktiranje autora oglasa.

Proveriti slučaj korišćenja Kontaktiranje administratora.

Proveriti slučaj korišćenja Pregled profila korisnika.

Proveriti slučaj korišćenja Ažuriranje podataka o korisniku.

Proveriti slučaj korišćenja Postavljanje novog oglasa.

Proveriti slučaj korišćenja Ažuriranje postojećeg oglasa.

Proveriti slučaj korišćenja Brisanje postojećeg oglasa.

Proveriti slučaj korišćenja Dodavanje novih kategorija.

Proveriti slučaj korišćenja Brisanje postojećih kategorija.

Proveriti slučaj korišćenja Dodavanje novih gradova.

Proveriti slučaj korišćenja Brisanje postojećih gradova.

Proveriti slučaj korišćenja Brisanje oglasa.

Proveriti slučaj korišćenja Brisanje korisničkog naloga.

Severska komponenta sistema treba da funkcioniše na računaru pod operativnim sistemom Windows.

Klijentska komponenta sistema treba da funkcioniše na računaru Pentium I sa 128MB operativne memorije (*Specifikacija zahteva*, odeljak 7.6).

**Testiranje poslovnog ciklusa (Business Cycle Testing)**

Proveriti posledice operacije registracije novog korisnika.

Proveriti posledice operacije prijave korisnika.

Proveriti posledice operacije dodavanja novog oglasa.

**Testiranje korisničkog interfejsa (User Interface Testing)**

Proveriti jednostavnost navigacije kroz navigacioni meni.

Sistem treba da bude lak za korišćenje i prilagođen korisnicima koji poseduju dobar nivo znanja rada na računaru (*Vizija sistema*, odeljak 5.2).

**Testiranje performansi (Performance Testing)**

Proveriti vreme potrebno za registrovanje na platformu.

Proveriti vreme potrebno za prijavljivanje korisnika na platformu.

Proveriti vreme potrebno za dodavanje novog oglasa.

Proveriti da li vreme potrebno za pristupanje bazi podataka u cilju izvršenje upita ne prelazi 5 sekundi (*Specifikacija zahteva*, odeljak 7.4).

**Testiranje opterećenja (Load Testing)**

Proveriti odziv sistema kada mu istovremeno pristupa 150 posetilaca.

Proveriti odziv sistema kada 50 posetilaca istovremeno pristupa stranici Oglasi.

Proveriti da li sistem može da podrži do 1000 simultanih pristupa korisnika platformi (*Arhitekturni projekat, odeljak 11).*

**Testiranje kritičnih slučajeva (Stress Testing)**

Proveriti vreme odziva sistema prilikom prvog korišćenja.

Proveriti vreme odziva sistema kada 50 posetilaca istovremeno pristupa stranici Oglasi.

**Testiranje volumena (Volume Testing)**

Proveriti odziv sistema kada je 90% kapaciteta diska popunjeno.

**Testiranje sigurnosti i kontrole pristupa (Security and Access Control Testing)**

Proveriti mogućnost prijavljivanja na sistem sa računara u lokalnoj mreži.

Proveriti prava pristupa korisnika koji pripadaju različitim grupama (Posetilac platforme, Učenik-Profesor, Administrator).

**Testiranje otkaza i oporavka (Failover / Recovery Testing)**

Proveriti da li je **LanTern** platforma dostupna 24 časa dnevno, 7 dana u nedelji. Vreme kada platforma nije dostupna ne sme da pređe 10%. (*Vizija sistema*, odeljak 9; *Arhitekturni projekat*, odeljak 12)

**Testiranje konfiguracije (Configuration Testing)**

Severska komponenta sistema treba da funkcioniše na računaru pod operativnim sistemom Windows.

Klijentska komponenta sistema treba da funkcioniše na računaru Pentium I sa 128MB operativne memorije (*Specifikacija zahteva*, odeljak 7.6).

Proveriti da li klijentski deo **LanTern** platforme može da se koristi pomoću *Web browser-a*: Microsoft Edge, Google Chrome i Mozilla Firefox (*Specifikacija zahteva*, odeljak 7.6).

**Testiranje instalacije (Installation Testing)**

Proveriti instalaciju sistema na serveru.

# Strategije testiranja (Test Strategy)

Strategije testiranja predstavljaju preporuke na koji način treba vršiti testiranje softverske aplikacije. Prethodni odeljak o Zahtevima testiranja opisuje šta će biti testirano, dok će u ovom odeljku biti opisano na koji način će navedeni elementi biti testirani.

Osnova svake strategije testiranja jesu tehnike koje se koriste i kriterijumi na osnovu kojih se može utvrditi kada je testiranje završeno.

## Tipovi testiranja (Testing Types)

### Testiranje podataka i integriteta baze podataka (Data and Database Integrity Testing)

|  |  |
| --- | --- |
| Cilj Testa: | Utvrditi da funkcije i procesi za pristup bazi podataka funkcionišu korektno i bez grešaka po podacima. |
| Tehnike: | 1. Probati svaku funkciju za pristup bazi podataka, za korektne podatke (ili za pribavljanje podataka). 2. Ispitati bazu podataka da bi se utvrdilo da se podaci unose kao što se očekuje, da se svi događaji u bazi obrađuju korektno i da se podaci pribavljaju korektno i bez grešaka. |
| Kriterijum završetka: | Funkcije i procesi za pristup bazama podataka rade kao što je predviđeno i bez grešaka u podacima. |
| Napomene: | 1. Testiranje zahteva korišćenjem okruženja za rad sa konkretnim DBMS-om da bi se direktno uneli i ažurirali podaci u bazi podataka. 2. Procesi moraju da se ručno pokrenu. 3. Koristiti bazu sa malim ili minimalnim brojem podataka, kako bi se povećala preglednost i lakše uočili neprihvatljivi slučajevi. |

### Testiranje sistema (System Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi korektno kretanje kroz aplikaciju, unos podataka, njihovu obradu i pribavljanje. |
| Tehnike: | 1. Izvršiti svaki slučaj korišćenja, ispratiti sve tokove definisane slučajevima korišćenja, korišćenjem tačnih i netačnih podataka. 2. Utvrditi da li se očekivani rezultati javljaju kada se radi sa tačnim podacima. 3. Utvrditi da li se predviđene poruke o grešci ili upozorenja javljaju kada se radi sa netačnim podacima. |
| Kriterijum završetka: | 1. Svi planirani testovi su izvršeni. 2. Sve identifikovane greške su adekvatno obrađene. |
| Napomene: | 1. Potrebno je obezbediti pristup serveru na kome će biti instaliran sistem. |

### Testiranje poslovnog ciklusa (Business Cycle Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi odgovarajuću primenu i procese koji se obavljaju kao podrška zahtevanom modelu poslovanja i utvrđenom rasporedu. |
| Tehnike: | 1. Testiranje simulira poslovne cikluse koji obavljaju sledeće: 2. Testovi korišćeni za testiranje funkcija aplikacija treba modifikovati/proširiti da uvećaju broj izvršenja svake funkcije kako bi se simulirao pristup većeg broja različitih korisnika u odgovarajućem periodu vremena. 3. Utvrditi da li se očekivani rezultati javljaju kada se radi sa tačnim podacima. 4. Utvrditi da li se predviđene poruke o grešci ili upozorenja javljaju kada se radi sa netačnim podacima. |
| Kriterijum završetka: | 1. Svi planirani testovi su izvršeni. 2. Svi identifikovane greške su adekvatno obrađene. |
| Napomene: | 1. Poslovni model bi trebao da identifikuje odgovarajuće zahteve i procedure koje treba testirati. |

### Testiranje korisničkog interfejsa (User Interface Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi sledeće:   1. Navigacija kroz aplikaciju korektno odslikava funkcije i zahteve sistema, uključujući kretanje između stranica, polja i funkcija pristupa (prelazak tabom, kretanje miša). 2. Objekti stranica i njihove karakteristike, kao što su navigacioni menia i dobijanje fokusa u skladu sa standardima. |
| Tehnike: | 1. Kreirati/modifikovati testove za svaku stranicu da bi se potvrdila odgovarajuća navigacija. |
| Kriterijum završetka: | Svaka stranica treba da se potvrdi da je konzistentna sa referentnom verzijom ili prihvaćenim standardima. |
| Napomene: | Nema. |

### Testiranje performansi (Performance Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi vreme odziva sistema za napravljene transakcije ili poslovne funkcije za sledeća dva slučaja:   * procenjeno prosečno opterećenje. * procenjeno maksimalno opterećenje. |
| Tehnike: | 1. Skripte treba da se izvršavaju na jednom računaru. |
| Kriterijum završetka: | 1. Jedan korisnik: Uspešno izvršavanje test skripti bez ikakvih padova i za očekivano vreme. 2. Više korisnika: Uspešno izvršavanje test skripti bez ikakvih padova i za očekivano vreme. |
| Napomena: | 1. Testiranje performansi sistema uključuje opterećenje sistema u pozadini. Postoji više metoda koji se mogu koristiti da bi se dobili zahtevani uslovi: 2. Slati zahteve direktno na serveru u formi SQL upita. 3. Koristiti više ljudi koji će sa različitih računara pokretati test skripte da bi se dostiglo odgovarajuće opterećenje. 4. Testiranje performansi treba da se obavi na pripremljenom računaru u predviđeno vreme. Time se omogućuje potpuna kontrola i precizno merenje. 5. Baza podataka koja se koristi za testiranje performansi treba da ima veličinu baze koja će se realno koristiti. |

### Testiranje opterećenja (Load Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi vreme odziva sistema za napravljene transakcije i određene slučajeve pod uslovima različitog opterećenja. |
| Tehnike: | 1. Koristiti testove napravljene za testiranje poslovnog ciklusa. 2. Modifikovati podatke tako da se uveća transakcija i modifikovati skripte da se poveća broj izvršenja transakcija. |
| Kriterijum završetka: | 1. Više korisnika: Uspešno izvršavanje test skripti bez ikakvih padova i za očekivano vreme. |
| Napomene: | 1. Testiranje performansi treba da se obavi na pripremljenom računaru u predviđeno vreme. Time se omogućuje potpuna kontrola i precizno merenje. 2. Baza podataka koja se koristi za testiranje performansi treba da ima veličinu baze koja će se realno koristiti. |

### Testiranje kritičnih slučajeva(Stress Testing)

|  |  |
| --- | --- |
| Cilj testa: | Proveriti da li sistem funkcioniše korektno i bez grešaka u sledećim kritičnim slučajevima:   1. Malo ili nimalo memorije na serveru. 2. Maksimalan broj klijenata (stvarnih ili simuliranih) koji rade na sistemu. 3. Više korisnika koji zahtevaju istu transakciju nad istim podacima. 4. Najgori slučaj broja transakcija koje sistem može da izdrži.   Napmena: Cilj testiranja kritičnih slučajeva može biti i određivanje uslova pod kojima sistem nije u mogućnosti da nastavi da radi korektno. |
| Tehnike: | 1. Koristiti testove koji su napravljeni za testiranje performansi. 2. Da bi se obavilo testiranje pod ograničenjima, testovi treba da se obaljaju sa jednog računara, a veličinu RAM i spoljne memorije treba smanjiti. 3. Za ostale testove, treba koristiti više klijenata, koji će raditi na istom testu ili na komplementarnim testovima da bi se proizveo slučaj sa najvećom količinom podataka u transakciji. |
| Kriterijum završetka: | Svi planirani testovi se izvršavaju i predviđene granice sistema su dostignute bez pada sistema ili su uslovi pod kojima se pojavljuje otkaz sistema izvan predviđenih granica. |
| Napomene: | 1. Testiranje mreže može zahtevati korišćenje alata koji omogućavaju povećanje protoka podataka na mreži. 2. Veličina spoljne memorije treba da bude privremeno smanjena kako bi smanjila prostor koji može da koristi baza podataka. 3. Testiranje zahteva sinhronizaciju istovremenog pristupa klijenata istim podacima. |

### Testiranje volumena (Volume Testing)

|  |  |
| --- | --- |
| Cilj testa: | Potvrditi da sistem funkcioniše uspešno u sledećim slučajevima:   1. Maksimalan broj klijenata koji su povezani (simulirani) i koji izvršavaju iste poslovne funkcije koje čine najgori slučaj za duži period vremena. 2. Maksimalna veličina baze podataka je dostignuta i više transakcija koje izvršavaju upite i pribavljaju podatke istovremeno. |
| Tehnike: | 1. Koristiti testove koji su napravljeni za testiranje performansi. 2. Koristiti više klijenata, koji će raditi na istom testu ili na komplementarnim testovima da bi se proizveo slučaj sa najvećom količinom podataka u transakciji na duži period vremena. 3. Kreira se baza podataka maksimalne veličine (stvarna, uvećana ili popunjena reprezentativnim podacima) i više klijenata koji pokreću transakcije radi istovremenog pribavljanja podataka u toku dužeg perioda vremena. |
| Kriterijum završetka: | Svi planirani testovi se izvršavaju i predviđene granice sistema su dostignute bez pada sistema. |
| Napomene: | 1. Odabrati period vremena koji će biti prihvatljiv za testiranje uslova velikog volumena. |

### Testiranje sigurnosti i kontrole pristupa (Security and Access Control Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Sigurnost funkcija/podataka: Potvrditi da korisnik može da koristi samo one funkcije/podatke koji su predviđeni za tip korisnika kome pripada.  Sigurnost sistema: Potvrditi da samo oni korisnici kojima je dozvoljeno da pristupe sistemu mogu da pristupe sistemu. |
| Tehnike: | 1. Sigurnost funkcija/podataka: Identifikovati funkcije/podatke koji imaju pravo pristupa za odgovarajućeg tipa korisnika. 2. Napraviti testove za svaki tip korisnika i potvrditi dozvolu da izvršava transakcije koje su specifične za taj tip korisnika. 3. Izmeniti tip korisnika i proveriti ga na iste testove. Proveriti da li su funkcije/podaci korektno dozvoljeni ili zabranjeni za korišćenje. |
| Kriterijum završetka: | Za svaki tip korisnika i odgovarajuće funkcije/podatke dozvoljene za korišćenje funkcionišu na očekivani način, definisane u testovima sistema. |
| Napomene: | Nema. |
|  |  |

### Testiranje otkaza i oporavka (Failover / Recovery Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi da proces opravka (ručni ili automatski) korektno vraća bazu podataka, aplikaciju i sistem u očekivano stanje. Sledeći tipovi uslova treba da budu uključeni u testiranje:   1. Isključenje napajanja na serveru. 2. Prekid komunikacije kroz mrežu. 3. Prekid komunikacije ili gubitak podataka usled greške rada hard diska. 4. Nekompletni ciklus (prekid u procesu filtriranja podataka, prekid u procesu sinhronizacije podataka). 5. Netačni pokazivači (ključevi) u bazi podataka. 6. Netačni podatak u bazi podataka. |
| Tehnike: | Test kreiran za testiranje funkcionalnosti i poslovnog ciklusa se koriste za formiranje niza transakcija. Kada se postigne početna tačka koja omogućava testiranje, sledeće akcije treba sprovesti (simulirati), individualno:   1. Prekid napajanja na serveru. 2. Prekid na mreži: simulirati ili izazvati gubitak komunikacije kroz mrežu (fizički isključiti komunikacioni kabl ili isključiti mrežni server ili ruter). 3. Prekid komunikacije, odnosno gubitak napajanja, sa hard diskom: simulirati ili fizički eliminisati komunikaciju sa jednim ili više kontrolera ili uređaja.   Kada se uslovi iznad ostvare treba pokrenuti procedure za oporavak.  Testiranje nekompletnih ciklusa koristi istu tehniku opisanu iznad, s tim što procesi na bazi podataka treba da budu prekinuti ili zaustavljeni.  Testiranje na sledeće uslove zahteva da se sistem nalazi u poznatom stanju. Nekoliko polja u bazi podataka, pokazivača i ključeva treba narušiti ručno i direktno u bazi podataka (korišćenjem alata baze podataka). Dodatne transakcije treba izvršiti korišćenjem testova iz testiranja funkcionalnosti i testiranja poslovnog ciklusa. |
| Kriterijum završetka: | U svim opisanim slučajevima aplikacija, baza podataka i sistem treba da se nakon završetka i procedura oporavka, vrate u očekivano stanje. To stanje uključuje da je narušavanje podataka ograničeno samo na narušena polja, pokazivače, ključeve i izveštaje u okviru procesa koji nisu završeni usled nastanka prekida. |
| Napomena: | 1. Testiranje oporavka zahteva operacije koje nekada nisu izvodljive ili ih nije poželjno obavljati (simulacija prekida komunikacije ili napajanja). Alternativa tome su softverski alati za dijagnostiku. 2. Resursi sistema (računara), baze podataka i mrežne grupe se koriste. 3. Testiranje treba obavljati u časovima kada se resursi ne koriste ili treba ga obavljati na izolovanoj mašini. |

### Testiranje konfiguracije (Configuration Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Proveriti i potvrditi da klijentska aplikacija funkcioniše na konfiguracijama na kojima je to predviđeno. |
| Tehnike: | 1. Koristi testove integracije i funkcionalne testove. 2. Pokrenuti/zatvoriti različite aplikacije, bilo pre pokretanja testa, bilo kao deo testiranja. 3. Izvršiti izabrane transakcije da bi se simulirale korisničke aktivnosti u različitim aplikacijama. 4. Ponoviti navedene procese, smanjujući RAM memoriju koju koristi klijent. |
| Kriterijum završetka: | Za svaku kombinaciju, transakcije iz testa treba da se završe uspešno, bez grešaka. |
| Napomene: | 1. Koje su aplikacije dostupne i mogu se koristiti na klijentu? 2. Koje podatke aplikacija koristi (koji je obim tih podataka)? 3. Ceo sistem, mrežni serveri, baze podataka i slično treba biti deo ovog testa. |

## Alati

U sledećoj tabeli su pobrojani alati koji će biti korišćeni za testiranje sistema:

|  |  |  |  |
| --- | --- | --- | --- |
| **Faza** | **Alat** | **Proizvođač/ matični proiz.** | **Verzija** |
| Test Management | Rational Unified Process | Rational | TBD |
| Test Design | Rational Rose | Rational | TBD |
| Defect Tracking |  |  | TBD |
| Functional Testing |  |  | TBD |
| Performance Testing |  |  | TBD |
| Test Coverage Monitor or Profiler |  |  | TBD |
| Other Test Tools |  |  | TBD |
| Project Management | Project  Word  Excel | Microsoft  Microsoft  Microsoft | TBD |
| DBMS tools | MySQL | MySQL | TBD |

# Resursi

U ovom odelju su opisani resursi koji se mogu koristiti za testiranje sistema, glavne odgovornosti i njihovo znanje ili veštine.

## Radnici

Ova tabela pokazuje predlog rasporeda radnika na aktivnostima testiranja.

|  |  |  |
| --- | --- | --- |
| **Ljudski resursi** | | |
| **Radnik** | **Preporučeni minimalni broj radnika**  (broj radnika koji će raditi puno radno vreme) | **Posebne preporuke / komentari** |
| Rukovodilac testiranja | Snežana | Obezbeđuje nadgledanje upravljanjem procesa.  Odgovornosti:   1. Prikupljanje odgovarajućih resursa. 2. Upravljanje izveštavanjem. |
| Projektant testova | Snežana | Identifikuje, određuje prioritete i implementira slučajeve testiranja.  Odgovornosti:   1. Napraviti plan testiranja. 2. Napraviti model testiranja. 3. Proceniti efektivnost napora uloženog u testiranje. |
| Tester sistema | Snežana | Izvršiti testiranje.  Odgovornosti:   1. Izvršiti testiranje. 2. Zabeležiti rezultate. 3. Popraviti greške. 4. Opisati greške. |
| Administrator testiranja sistema | Snežana | Pripremiti okruženje za testiranje i utvrditi da ono odgovara pretpostavkama..  Odgovornosti:   1. Administrirati upravljanje testiranjem. 2. Upravljati pristupom resursima na kojima se obavlja testiranje. |
| Administrator baze podataka | Snežana | Pripremiti podatke i bazu podataka za testiranje i utvrditi da njihovo stanje odgovara pretpostavkama.  Odgovornosti:   1. Administrirati podatke (bazu podataka) za testiranje. |
| Projektant | Snežana | Identifikovati i definisati operacije, funkcije i veze između različitih skripti testova.  Odgovornosti:   1. Identifikovati i definisati skripte testova. 2. Identifikovati i definisati pakete testova. |
| Implementacija | Snežana | Implementirati testove i rasporediti ih u odgovarajuće skripte i pakete testova.  Odgovornosti:   1. Kreirati skripte i pakete testova implementirane u test modelu. |

## Sistem

U sledećoj tabeli su navedeni sistemski resursi koji će se koristiti za testiranje **LanTern** platforme.

|  |  |
| --- | --- |
| **Sistemski resursi** | |
| **Resurs** | **Serijski broj** |
| Server | Serijski broj: X179773562b |
| Klijentski Test Računari: 3 udaljena računara (sa internet pristupom) | Serijski broj: A8339223  Serijski broj: B9334022  Serijski broj: B9332544 |
|  |  |
| Lokalnih računara: 3 (povezanih u lokanu mrežu) | Serijski broj: R3322411 (IT Lab)  Serijski broj: A8832234 (IT Lab)  Serijski broj: W4592233 (IT Lab) |
| Test repozitorijum | Serial No: X179773562b |
| Server |  |
| Računari za razvoj testova: 4 | Serijski broj: R3322411  Serijski broj: A8832234  Serijski broj: W4592233  Serijski broj: X3333411 |
| Simulator opterećenja | Serijski broj: ABC-123 |

# Kontrolne tačke testiranja

Aktivnosti testiranja zavise od iteracija u razvoju softvera. Ciklus razvoja softvera **LanTern** biće odrađen u jednoj iteraciji koja sadrži kompletan ciklus testiranja, planiranje, projektovanje, razvoj, izvršenje i evaluaciju.

Sledeća tabela prikazuje kontrolne tačke testiranja:

|  |  |  |  |
| --- | --- | --- | --- |
| **Zadatak** | **Trud (pd)** | **Početak** | **Kraj** |
| Iteracija **C1**: R1.0 Release  -Planiranje testova  -Projektovanje testova  -Razvoj testova  -Izvršenje testova  -Evaluacija testova | 5 | Maj 11 | Maj 23 |

# Izlazni produkti testiranja

Rezultati testiranja koji su definisani ovim planom testiranja navedeni su u sledećoj tabeli.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dokument** | **Vlasnik** | **Ocena / Raspodela** | **Rok** |
| Plan Testiranja | Snežana | Rukovodilac testiranja | Maj 11 |
| Test okruženje | Snežana | Rukovodilac testiranja | Maj 12 |
| Model testiranja | Snežana | Rukovodilac testiranja | Maj 14 |
| Skup test podataka | Snežana | Rukovodilac testiranja | Maj 14 |
| Test procedure | Snežana | Rukovodilac testiranja | Maj 15 |
| Test skripte | Snežana | Rukovodilac testiranja | Maj 17 |
| Test podrška | Snežana | Rukovodilac testiranja | Maj 19 |
| Izveštaj o uočenim greškama | Snežana | Rukovodilac testiranja | Maj 20 |
| Rezultati testa | Snežana | Rukovodilac testiranja | Maj 21 |
| Izveštaj o evaluaciji testa | Snežana | Rukovodilac testiranja | Maj 23 |

## Model testiranja

Model testiranja definiše sve test slučajeve zajedno sa referencama na test procedure i test skripte koje su vezane za svaki test slučaj.

# Projektni zadaci

U sledećoj tabeli su navedeni projektni zadaci testiranja sistema:

|  |
| --- |
| **Plan testiranja** |
| Identifikovati zahteve za testiranjem |
| Proceniti rizike |
| Razviti strategije testiranja |
| Identifikovati resurse testiranja |
| Napraviti raspored testiranja |
| Napraviti Plan testiranja |
| **Projektovanje testova** |
| Analizirati potrebni rad |
| Razviti model testiranja |
| Identifikovati i opisati test slučajeve |
| Identifikovati i grupisati procedure testiranja |
| Oceniti, napraviti uvid u pokrivenost testiranja |
| **Implementacija testova** |
| Postaviti okruženje za testiranje |
| Napisati skripte za testiranje |
| Razviti podršku za testiranje |
| Identifikovati sprecifične funkcionalnosti za projektovanje i Implementaciju modela testiranja |
| **Testiranje** |
| Izvršiti test procedure |
| Oceniti izvršenje testova |
| Proveriti rezultate |
| Istražiti neočekivane rezultate |
| Zabeležiti greške |
| **Ocena testiranja** |
| Oceniti pokrivenost funkcionalnosti test slučajevima |
| Proceniti pokrivenost koda test slučajevima |
| Analizirati greške |
| Utvrditi da li su kriterijumi završetka i kriterijumi uspešnosti postignuti |
| Napraviti Izveštaj o oceni testiranja |