**Running++  
Android aplikacija** **za trčanje i praćenje statistike**

**Plan testiranja**

**Verzija 1.0**

**Pregled izmena**

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autori** |
| 7.5.2019. | 1.0 | Inicijalna verzija | Aleksandar Stoiljković  Nikola Gocić  Lazar Stoiljković |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

1. Ciljevi [4](#__RefHeading___Toc166369058)

2. Opseg [4](#__RefHeading___Toc166369059)

3. Reference [4](#__RefHeading___Toc166369060)

4. Zahtevi za testiranje 5

5. Strategije testiranja (Test Strategy) [7](#__RefHeading___Toc166369062)

5.1 Tipovi testiranja (Testing Types) 8

5.1.1 Testiranje podataka i integriteta baze podataka (Data and Database Integrity Testing) 8

5.1.2 Testiranje sistema (System Testing) 8

5.1.3 Testiranje poslovnog ciklusa (Business Cycle Testing) 9

5.1.4 Testiranje korisničkog interfejsa (User Interface Testing) 10

5.1.5 Testiranje performansi (Performance Testing) 10

5.1.6 Testiranje opterećenja (Load Testing) [1](#__RefHeading___Toc166369069)2

5.1.7 Testiranje kritičnih slučajeva(Stress Testing) [1](#__RefHeading___Toc166369070)3

5.1.8 Testiranje volumena (Volume Testing) [1](#__RefHeading___Toc166369071)3

5.1.9 Testiranje sigurnosti i kontrole pristupa (Security and Access Control Testing) [1](#__RefHeading___Toc166369072)4

5.1.10 Testiranje otkaza i oporavka (Failover / Recovery Testing) [1](#__RefHeading___Toc166369073)4

5.1.11 Testiranje konfiguracije (Configuration Testing) [1](#__RefHeading___Toc166369074)6

5.1.12 Testiranje instalacije (Installation Testing) [1](#__RefHeading___Toc166369075)6

5.2 Alati [1](#__RefHeading___Toc166369076)7

6. Resursi [1](#__RefHeading___Toc166369077)7

6.1 Radnici [17](#__RefHeading___Toc166369078)

6.2 Sistem [1](#__RefHeading___Toc166369079)9

7. Glavne postavke testiranja [19](#__RefHeading___Toc166369080)

8. Rezultati testiranja [19](#__RefHeading___Toc166369081)

8.1 Model testiranja [20](#__RefHeading___Toc166369082)

9. Projektni zadaci 20

**Plan Testiranja**

# Ciljevi

Ovaj dokument sadrži opis plana testiranja sistema Running++. Namena ovog dokumenta je da opiše način provere funkcionalni i nefunkcionalne zahteve postavljene 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 testiranje.
* Navesti dokumente koji će opisati rezultate testiranja.

# Opseg

Plan testiranja se odnosi na integraciju sistema i proveru izvršne verzije aplikacije Running++.

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 i
* testiranje kritičnih performansi aplikacije.

# Reference

Spisak korišćene literature:

1. Running++ - Predlog projekta, PredlogProjekta-Running++, V1.0, 2019, AMG
2. Running++ - Vizija sistema, VizijaSistema-Running++, V1.2, 2019, AMG
3. Running++ - Plan realizacije, PlanRealizacije-Running++, V1.0, 2019, AMG
4. Running++- Planirani raspored aktivnosti, RasporedAktivnosti, V1.0,2019,AMG
5. Running++ - Specifikacija zahteva, SpecifikacijaZahteva-Running++, V1.0, 2019 AMG
6. Running++ - Arhitekturni projekat, ArhitekturniProjekat-Running++, V1.0, 2019 AMG

# 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 Running++.

Proveriti mogućnost istovremenog pristupa istim podacima radi čitanja.

Proveriti zaključavanje podataka tokom pristupa bazi radi ažuriranja.

Proveriti tačnost podataka koji se pribavljaju iz baze

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

Proveriti slučaj korišćenja Pracenje aktivnosti(trcanje)

Proveriti slučaj korišćenja Prikaz trenutnih rezultata

Proveriti slučaj korišćenja Unos ličnih podataka

Proveriti slučaj korišćenja Ažuriranje ličnih podataka

Proveriti slučaj korišćenja Odjavljivanje iz aplikacije

Proveriti slučaj korišćenja Kreiranje naloga

Proveriti slučaj korišćenja Logovanje

Proveriti slučaj korišćenja Prijavljivanje preko Google/Facebook naloga

Proveriti slučaj korišćenja Pretrega ostalih korisnika

Proveriti slučaj korišćenja Dodavanje korisnika u listu prijatelja

Proveriti slučaj korišćenja Brisanje korisnika iz liste prijatelja

Proveriti slučaj korišćenja Prikaz i sortiranje rezultata

Proveriti slučaj korišćenja Prikaz dnevnih rezultata

Proveriti slučaj korišćenja Prikaz nedeljnih rezultata

Proveriti slučaj korišćenja Prikaz mesecnih rezultata

Proveriti slučaj korišćenja Soritranje-Distanca

Proveriti slučaj korišćenja Sortiranje-Vreme trčanja

Proveriti slučaj korišćenja Sortiranje-Potrošene kalorije

Proveriti slučaj korišćenja Sortiranje-Brzina trčanja

Proveriti slučaj korišćenja Logovanje u administratorski nalog

Proveriti slučaj korišćenja Brisanje i ažuriranje korisničkog naloga

Klijentska komponenta sistema treba da funkcionise na mobilnim uredjajima sa opertivnim sistemom Android(od verzije 4).

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

Proveriti posledice operacije brisanja,ažuriranja i dodavanja korisnika.

Proveriti posledice operacije prijavljivanja novog korisnika.

Proveriti posledice operacije pribavljanja korisnika sa prosledjenim korisničkim imenom(username).

Proveriti posledice operacije pribavljanja svih korisnika.

Proveriti posledice operacije izmene podataka o korisniku.

Proveriti posledice operacije brisanja korisnika iz baze.

Proveriti posledice operacije pribavljanja aktivnosti za određenog korisnika.

Proveriti posledice operacije pribavljanja svih aktivnosti.

Proveriti posledice operacije dodavanja nove aktivnosti.

Proveriti posledice operacije ažuriranja aktivnosti.

Proveriti posledice operacije brisanja korisnika i brisanja aktivnosti.

Proveriti posledice operacije slanja zahteva za prijateljstvo korisniku.

Proveriti posledice operacije prihvatanja zahteva za prijateljstvo.

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

Proveriti jednostavnost navigacije kroz skup prozora aplikacije.

Proveriti da li izgled prozora aplikacije odgovara definisanim standardima za izradu korisničkog interfejsa.

Proveriti da li korisnički interfejs omogućava jednostavno i intuitivno korišćenje bez potrebe za organizovanjem dodatne obuke

Sistem treba da bude lak i pregledan za korišćenje

**Testiranje performansi (Performance Testing)**

Proveriti vreme potrebno za prijavljivanje u aplikaciju.

Proveriti vreme potrebno za registrovanje.

Proveriti vreme potrebno za prikaz aktivnosti.

Proveriti vreme potrebno za dodavanje novog korisnika.

Proveriti vreme potrebno za dodavanje nove aktivnosti.

Proveriti da li vreme potrebno za pristupanje bazi podataka u cilju izvršenje upita ne prelazi 5 sekundi.

Proveriti da li vreme potrebno za pristupanje serveru u cilju pribavljanja podataka ne prelazi 5 sekundi.

**Testiranje opterećenja (Load Testing)**

Proveriti odziv sistema kada mu istovremeno pristupa više korisnika.

Proveriti odziv sistema kada više korisnika koristi podatke koji se nalaze u bazi.

**Testiranje kritičnih slučajeva (Stress Testing)**

Proveriti vreme odziv sistema prilikom prvog korišćenja.

Proveriti vreme odziva sistema kada podaci sa servera nisu keširani.

**Testiranje volumena (Volume Testing)**

Proveriti odziv sistema kada je 90% kapaciteta servera popunjeno podacima.

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

Proveriti prava pristupa različitih tipova korisnika (Guest korisnik, Korisnik sa nalogom i Administrator).

Proveriti mogućnost pristupa ličnim podacima korisnika u slučaju privatnog profila.

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

Proveriti da li je Running++ aplikacija i server dostupani 24 časa dnevno, 7 dana u nedelji.

Proveriti da li srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati

**Testiranje konfiguracije (Configuration Testing)**

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

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

Klijentska komponenta sistema treba da funkcionise na mobilnim uredjajima sa opertivnim sistemom Android(od verzije 4).

**Testiranje instalacije (Installation Testing)**

Proveriti instalaciju sistema na mobilnom uredjaju.

# 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 kako 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.

* 1. **Tipovi testiranja(Testing Types)**

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

|  |  |
| --- | --- |
| Cilj Testa: | Utvrditi da metodi i procesi za pristup bazi podataka funkcionišu korektno i bez grešaka po podacima. |
| Tehnike: | * Probati svaki metod pristupa bazi podataka, za korektne i nekorektne podatke (ili za pribavljanje podataka) * Ispitati bazu podataka da bi se utvrdilo da se podaci unose kao što se očekuje, da svi događaji u bazi su obrađuju korektno i da se podaci pribavljaju korektno i bez grešaka. |
| Kriterijum završetka: | Metodi i procesi za pristup bazama podataka rade kao što je predviđeno i bez grešaka u podacima. |

|  |  |
| --- | --- |
| Napomene: | * Koristiti bazu sa malim ili minimalnim brojem podataka, kako bi se povećala preglednost i lakše uočili neprihvatljivi slučajevi. |

### 5.1.2 Testiranje sistema (System Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi korektno kretanje kroz aplikaciju, unos podataka, njihovu obradu i pribavljanje. |
| Tehnike: | * Izvršiti svaki slučaj korišćenja, ispratiti sve tokove definisane slučajevima korišćenja, korišćenjem tačnih i netačnih podataka. * Utvrditi da li se očekivani rezultati javljaju kada se radi sa tačnim podacima. * Utvrditi da li se predviđene poruke o grešci ili upozorenja javljaju kada se radi sa netačnim podacima. |
| Kriterijum završetka: | * Svi planirani testovi su izvršeni. * Svi identifikovane greške su adekvatno obrađene. |
| Napomene: | * Potrebno je obezbediti pristup serveru * Potrebno je obezbediti fajlove sa test podacima za testiranje razlicitih slucajeva koriscenja |

5.1.3 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: | * Testiranje simulira poslovne cikluse koji obavljaju sledeće: * 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. * Sve funkcije koje su vremenski zavisne treba proveriti korišćenjem tačnih i netačnih podataka * Proveriti da li se sve funkcije koje se pojavljuju periodično da se izvršavaju/pokreću u odgovarajućem vremenskim trenucima. |
|  |  |

* Utvrditi da li se očekivani rezultati javljaju kada se radi sa tačnim podacima.
* Utvrditi da li se predviđene poruke o grešci ili upozorenja javljaju kada se radi sa netačnim podacima.
* Potvrditi da su poslovna pravila pravilno primenjena.

|  |  |
| --- | --- |
| Kriterijum završetka: | * Svi planirani testovi su izvršeni. * Svi identifikovane greške su adekvatno obrađene. |
| Napomene: | * Određivanje datuma i perioda može zahtevati dodatne aktivnosti. * Poslovni model bi trebao ad identifikuje odgovarajuće zahteve i procedure koje treba testirati. |

5.1.4 Testiranje korisnickog interfejsa (User Interface Testign)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi sledeće:   * Testiranje različitih formi aplikacije i funkcionalnosti koje te forme uključuju. * Elementi forme i njihov prikaz ima željeni oblik (Veličina i pozicija) . |

|  |  |
| --- | --- |
| Tehnike: | * Kreirati/modifikovati testove za svaku formu da bi se testirale odgovarajuce funcionalnosti i prikaz elemenata. |

|  |  |
| --- | --- |
| Kriterijum završetka: | Testiranjem svake forme treba da se potvrdi odgovarajuci izgled i ponasanje. |

### 5.1.5 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: | * Koristiti test procedure napravljene za testiranje sistema. * Primeniti pristup aplikaciji od strane vise korisnika * Ispitivati različite funkcionalnosti aplikacije |
| Kriterijum završetka: | * Uspešno izvršavanje svih funkcionalnosti za ocekivano vreme bez ikakvih kvarova. |
| Napomena: | * Testiranje performansi sistema uključuje opterećenje sistema u pozadini. Postoji više metoda koji se mogu koristiti da bi se dobili zahtevani uslovi: * Pozivati transakcije direktno na serveru, obično u formi SQL poziva. * Kreirati virtuelnog korisnika koji će simulirati više klijenata. Može se iskoristiti softver koji će sa udaljenog mobilnog uredjaja biti iskorišćen da poveća opterećenje. Time se, takođe, može postići povećanje opterećenja na mreži. * Testiranje performansi treba da se obavi na za to spremljenom mobilnom uredjaju u predviđeno vreme. Time se omogućuje potpuna kontrola i precizno merenje. * Baza podataka koja se koristi za testiranje performansi treba da ima veličinu baze koja će se realno koristiti. |

### 5.1.6 Testiranje opterećenja (Load Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Potvrditi vreme odziva sistema pod uslovima razlicitog opterecenja |

|  |  |
| --- | --- |
| Tehnike: | * Koristiti testove napravljene za testiranje poslovnog ciklusa. * Testiranje od strane veceg broja korisnika kako bise povecalo opterecenje * Promena macina pristupa podacima na serveru u cilju povecanja opterecenja |
| Kriterijum završetka: | * Uspesno izvrsenje svih funkcionalnosti aplikacije |
| Napomene: | * Testiranje performansi treba da se obavi na za to spremljenom mobilnom uredjaju u predviđeno vreme. Time se omogućuje potpuna kontrola i precizno merenje. * 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:   * malo memorije na serveru * maksimalan broj klijenata (stvarnih ili simuliranih) koji rade na sistemu * više korisnika koji zahtevaju istu transakciju nad istim podacima.   Napmena: Cilj testiranja kritičnih slučajeva može biti i određivanje uslova pod kojima sistem nije u mogućnosti da nastavi da radi korektno. |
|  |  |
| Tehnike: | * Koristiti testove koji su napravljeni za testiranje performansi. * Za ostale testove, treba koristiti više klijenata, koji će raditi na istom testu ili na komplementarnim testovima da bi se proizveo slučaj sa najvećim 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 su izvan predviđenih granica. |
|  |  |

### 5.1.8 Testiranje volumena (Volume testing)

|  |  |
| --- | --- |
| Cilj testa: | Potvrditi da sistem funkcioniše uspešno u sledećim slučajevima:   * Maksimalan broj klijenata koristi funkcionalnosti aplikacije koje vrse slanje/primanje podataka na/sa servera duzi vremenski preriod. * Maksimalna veličina baze podataka je dostignuta i više transakcija koje izvršavaju upite i pribavljaju podatke istovremeno. * Maksimalno opterecenje koje aplikacija moze da ostvari na mobilni uredjaj korisnika |
| Tehnike: | * Koristiti testove koji su napravljeni za testiranje performansi. * Koristiti više klijenata, koji ce na aplikaciji izvrsavati iste test primere sa ciljem opterecenja servera i baze podataka. * Kreira se baza podataka maksimalne veličine (stvarna, uvećana ili napunjena reprezentativnim podacima) i više klijenata koji pokreću funkcije aplikacije radi istovremenog pribavljanja ili slanja podataka u toku dužeg perioda vremena. |
| Kriterijum završetka: | Svi planirani testovi se izvršavaju i predviđene granice sistema su dostignute bez pada sistema. |

### 5.1.9 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 koje su predviđene za tip korisnika kome pripada.  Sigurnost sistema: Potvrditi da samo oni korisnici (Administraotori) kojima je dozvoljeno da pristupe sistemu mogu da pristupe sistemu. |

|  |  |
| --- | --- |
| Tehnike: | * Sigurnost funkcija/podataka: Identifikovati funkcije/podatke kojima ima pravo pristupa odgovarajući tip korisnika. * Napraviti testove za svaki tip korisnika i potvrditi dozvolu da izvršava transakcije koje su specifične za taj tip korisnika. * Ispitati date testove za sva 3 tipa korisnika (Guest korisnik, Korisnik sa nalogom i Administrator ) proveriti da li su svi podaci osigurani i da su pravi podaci konkretno dozvoljeni ili zabranjeni. |
| Kriterijum završetka: | Za svaki tip korisnika i odgovarajuće funkcije/podatke dozvoljene za korišćenje, sve transakcije funkcionišu na očekivani način, definisane u testovima sistema. |

### 5.1.10 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:   * Isključenje napajanja na klijentu * Isključenje napajanja na serveru * Prekid komunikacije kroz mrežu * Gubljenje mogucnosti za odredjivanje trenutne lokacije mobilnog uredjaja klijenta * Prekid komunikacije ili gubitak podataka usled greške rada mobilnog uredjaja * Nekompletni ciklus (prekid u procesu filtriranja podataka, prekid u procesu sinhronizacije podataka) * Netačni pokazivači (ključevi) u bazi podataka * Netačni podatak u bazi podataka |

|  |  |
| --- | --- |
| Tehnike: | Test kreiran za testiranje funkcionalnosti aplikacije se koriste za formiranje niza transakcija sa serverom. Kada se postigne početna tačka koja omogućava testiranje, sledeće akcije treba da sprovesti (simulirati), individualno:   * Prekid napajanja na klijentakom uredjaju * Prekid napajanja na serveru * Prekid na mreži: simulirati ili izazvati gubitak komunikacije kroz mrežu (fizički isključiti vezu sa mrezom na mobilnom uredjaju klijenta ili isključiti mrežni server ili ruter)   Kada se uslovi iznad ostvare, dodatne transakcije treba izvršiti i pre dostizanja naredne tačke testa, procedure za oporavak treba pokrenuti.  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: | * Testiranje oporavka zahteva operacije koje nekada nisu izvodive ili h nije poželjno obavljati (simulacija prekida komunikacije ili napajanja). Alternativa tome su softverski alati za dijagnostiku. * Testiranje treba obavljati u časovima kada se resursi ne koriste ili na izolovanoj mašini. |

### Testiranje konfiguracije (Configuration Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Proveriti i potvrditi da klijentska aplikacija funkcioniše na konfiguracijama na kojima je to predviđeno. |
| Tehnike: | * Koristi testove integracije i funkcionalne testove. * Pokrenuti/zatvoriti različite aplikacije, bilo pre pokretanja testa, bilo kao deo testiranja. * Izvršiti izabrane transakcije da bi se simulirale korisničke aktivnosti u različitim aplikacijama. |
| Kriterijum završetka: | Za svaku kombinaciju, transakcije iz testa treba da se završe uspešno, bez grešaka. |
| Napomene: | * Koje podatke aplikacija koristi (koji je obim tih podataka) * Ceo sistem, aplikacija, mrežni serveri, baze podataka i slično treba biti opisano ka deo ovog testa. |

* + 1. Testiranje instalacije (Installation Testing)

|  |  |
| --- | --- |
| Cilj testiranja: | Proveriti i potvrditi da se softver korektno može instalirati na svakom od klijenata pod sledećim uslovima:   * Nova instalacija na novom mobilnom uredjaju, nikad instalirana aplikacija. * Ažurirati mobilni uredjaj prethodno instaliran sa istom verzijom aplikacije * Ažurirati računar prethodno instaliran sa prethodnom verzijom aplikacije |
|  |  |
| Tehnike: | * Ručno ili definisanjem automatskih skripti proveriti uslove na ciljnoj mašini (nikad instalirana aplikacija, ista verzija, nova verzija). * Pokrenuti i instalirati aplikaciju. * Korišćenjem predefinisanog podskupa testova funkcionalnih testova, proveriti rad aplikacije. |
| Kriterijum završetka: | Aplikacija treba da se izvršavaju uspešno i bez grešaka. |

**5.2 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 |

**6. Resursi**

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

**6.1 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 | Aleksandar | Obezbeđuje nadgledanje upravljanjem procesa  Odgovornosti:   * Obezbediti tehničku direkciju * Prikupljanje odgovarajućih resursa * Upravljanje izveštavanjem |
| Projektant testova | Nikola | Identifikuje, određuje prioritete i implementira slučajeve testiranja  Odgovornosti:   * Napraviti plan testiranja * Napraviti model testiranja * Proceniti efektivnost napora uloženog u testiranje. |
| Tester sistema | Lazar,  Nikola,  Aleksandar | Izvršiti testiranje  Odgovornosti:   * Izvršiti testiranje * Zabeležiti rezultate * Popraviti greške * Opisati greške |
| Administrator testiranja sistema | Aleksandar | Pripremiti okruženje za testiranje i utvrditi da ono odgovara pretpostavkama.  Odgovornosti:   * Administrirati upravljanje testiranjem * Upravljati pristupom radnika resursima na kojima se obavlja testiranje. |
| Administrator baze podataka | Lazar | Pripremiti podatke i bazu podataka za testiranje i utvrditi da njihovo stanje odgovara pretpostavkama.  Odgovornosti:   * Administrirati podatke (bazu podataka) za testiranje |
| Projektant | Nikola | Identifikovati i definisati operacije, atribute i veze između različitih klasa testova  Odgovornosti:   * Identifikovati i definisati klase testova * Identifikovati i definisati pakete testova |
| Implementacija | Nikola,  Aleksandar,  Lazar | Implementirati testove i rasporediti ih u odgovarajuće klase i pakete testova.  Odgovornosti:   * Kreirati klase i pakete testova implementirane u test modelu. |

**6.2 Sistem**

U sledećoj tabeli su navedeni sistemski resursi koji će se koristiti za testiranje Running++ android aplikacije.

|  |  |
| --- | --- |
| **Sistemski resursi** | |
| **Resurs** | **Ime / Tip / Serijski broj** |
| Server | Desktop računar 1 |
| Klijentski Deo |  |
| —3 mobilna uređaja  (sa internet pristupom) | Mobilni uređaj 1(Lazar)  Mobilni uređaj 2(Nikola)  Mobilni uređaj 3(Aleksandar) |
| —1 lokalni računar | Laptop računar 1 |
| Test repozitorijum |  |
| —Server | Desktop računar 1 |
| Računari za razvoj testova - 6 | Desktop računar 2  Desktop računar 3  Desktop računar 4 |
| Simulator opterećenja | Serijski broj: ABC-123 |

**7. Kontrolne tačke testiranja**

Aktivnosti testiranja zavise od iteracija u razvoju softvera. Ciklus razvoja softvera Running++ aplikacije biće odrađen u jednoj iteraciji koja sadži 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 15 | Maj 22 |

**8. Izlazni produkti testiranja**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Dokument** | **Vlasnik** | **Ocena / Raspodela** | **Due Date** |
| Plan Testiranja | Lazar | Rukovodioci testiranja | Maj 07 |
| Test okruženje | Aleksandar | - | Maj 12 |
| Model testiranja | Nikola i Aleksandar | Interna ocena | Maj 15 |
| Skup test podataka | Lazar | Interna ocena | Maj 16 |
| Test procedure | Nikola,  Aleksandar | Interna ocena | Maj 17 |
| Test skripte | Nikola,  Aleksandar | - | Maj 17 |
| Test podrška | Nikola,  Aleksandar | - | Maj 17 |
| Izveštaj o uočenim greškama | Nikola,  Aleksandar | Rukovodioci testiranja | Maj 18 |
| Rezultati testa | Lazar | Rukovodilac testiranja | Maj 19 |
| Izveštaj o evaluaciji testa | Lazar | Rukovodilac testiranja | Maj 22 |

**8.1 Model testiranja**

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

**9. Projektni zadaci**

U sledećoj tabeli su navedeni projektni zadaci i testiiranja sistema:

**Plan testiranja**

Identifikovati zahteve za testiranjem

Proceniti rizike

Razviti strategije testiranja

Identifikovati resurse testiranja

Napraviti raspored

Napraviti Plan testiranja

**Projektovanje testova**

Analizirati potrebni rad

Razviti model testiranja

Identifikovati i opisati test slučajeve

Identifikovati i grupisati procedure testiranja

Oceniti i napraviti uvid u pokrivenost testiranja

**Implementacija testova**

Postaviti okruženje za testiranje

Napisati skripte za testiranje

Razviti podršku za testiranje

Identifikovati specifične funkcionalnosti za projektovanje i implementaciju modela testiranja

Utvrditi eksterne skupove podataka

**Testiranje**

Izvršiti test procedure

Ocenite izvršenje testova

Popraviti testove koji nisu prošli

Proveriti rezultate

Istražiti neočekivane razultate

Zabeležiti greške

**Ocene 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