

VISOKA TEHNIČKA ŠKOLA  
STRUKOVNIH STUDIJA  
SUBOTICA



## AUTOMATIZOVANO TESTIRANJE SISTEMA

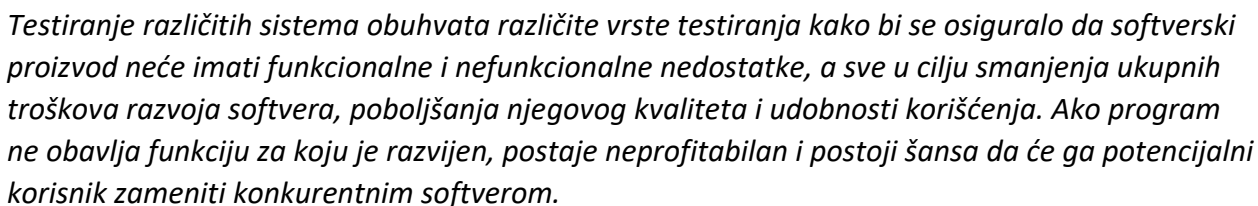
Seminarski rad iz predmeta: Softversko Inženjerstvo

**Student:** Aleksa Janković

**Broj indexa:** 16118131

**Studijski program:** Tehnička Informatika

Subotica, 2020.

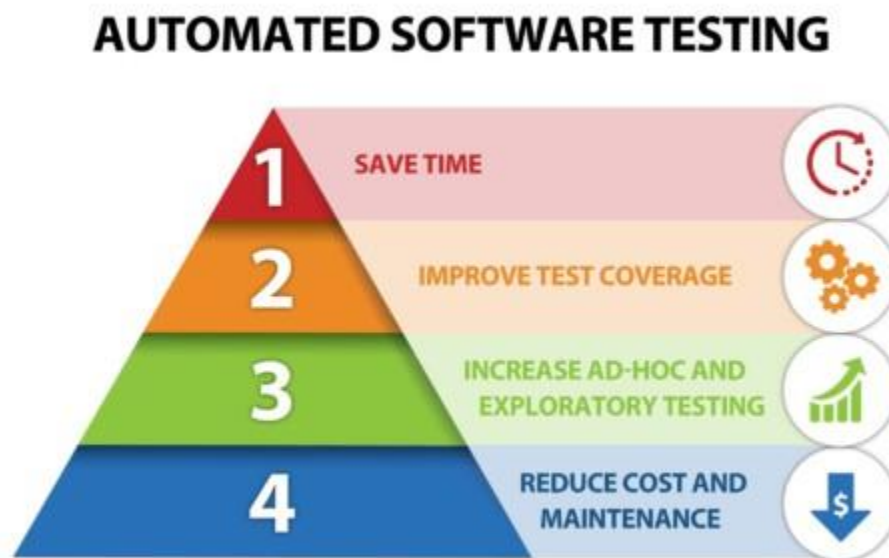


Automatsko testiranje softvera je proces u kojem se izvršavaju pripremljeni automatski test scenariji kako bi se pronašli defekti. Ručno testiranje obavlja čovek, koji sam izvršava sve korake. Kod automatskog testiranja, testove izvršava određeni računarski program, odnosno određeni alat za automatizaciju. Taj program izvršava korake, unosi podatke, poredi dobijene rezultate sa očekivanim i generiše izveštaj. Testovi koji treba da se repetativno izvršavaju (regresiono testiranje) mogu da se automatizuju kako bi se smanjila ukupna cena manualnog testiranja. Ovo je razumno kod test scenarija koji se odnose na funkcionalne zahteve koji se ne menjaju toliko često, a oduzimaju dosta napora za manualno testiranje. Automatizacija se može izvršiti bez obzira na to da li je aplikacija web, mobilna ili desktop. Takođe, moguće je i testiranje više međuzavisnih aplikacija, nezavisno od platforme. Automatizacija testova može da se izvrši imitirajući prave akcije klijenta tokom testiranja razvijene aplikacije ili kroz testiranje ne grafičkih nivoa aplikacije (API funkcionalno testiranje). Automatizacija testova sprečava ljudske greške u ručnom izvršavanju i smanjuje troškove procesa regresionog testiranja. Otežavajući faktor za automatizaciju je činjenica da se pojedine funkcionalnosti menjaju često i samim tim nisu pogodne za automatizaciju, a održavanje takvih test scenarija bi oduzimalo previše vremena i stručnih resursa.

Automatsko testiranje može da zameni manuelno testiranje, ali ne bi trebalo nikad da ukloni potrebu za njim.

#### **Prednosti automatizacije:**

- Smanjuje se vreme potrebno za testiranje.
- Samim tim, može da se pokrije i više scenarija, čak i onih manje verovatnih.
- Automatski testovi mogu da se izvršavaju stalno, noću i koliko god puta nam je potrebno.
- Manja je mogućnost za grešku.
- Test automatizacija je skoro besplatna s obzirom da ne zahteva veći broj testera, obično samo jednog test inženjera koji nakon razvijanja test slučajeva može se posvetiti drugim kreativnim poslovima.



#### **Mane automatizacije:**

- Automatizovani razvoj testova je obično 3-5 puta skuplji proces od kompletnog ciklusa manualnog testiranja.
- Automatizacija je previše glomazna problematika. Ko automatizuje? Ko održava? Ova pitanja komplikuju samu ideju automatizacije.
- Automatizacija će zahtevati dodatno obučeno osoblje.

### Koje testove treba automatizovati?

Što više to bolje. Ručno testiranje treba da ostane kao ispomoć, ali je najbolje kada je sve automatizovano, pre svega testovi koji su kompleksni i imaju prioritet. Pre nego što se test automatizuje, mora ručno da se prođe kroz sve njegove korake.

**Alati za automatizaciju** - Izbor alata koji će se koristiti zavisi od tehnologije na koju se oslanja program koji testiramo. Ima ih sijaset i stalno novi izlaze na tržište. Neki od njih su: Selenium, QTP, Appium itd.



### Tehnike Testiranja:

#### Black Box

BlackBox testiranje je metod softverskog testiranja u kojem unutrašnja struktura sistema koji se testira nije poznata testeru (tester je vidi kao crnu kutiju i ne zna kako je softver implementiran). Koristeći ovu tehniku možemo da nađemo greške u netačnim ili nedostajućim funkcijama, greške interfejsa, greške pristupa bazama podataka, greške performansi, itd. Prednost ovog metoda testiranja je da se lako izvodi i ne zahteva znanje o programiranju ni o unutrašnjoj strukturi sistema. Jedna od mana jeste da testovi mogu biti teški za dizajniranje i redundantni. Takođe ova tehnika ne može da testira sve moguće funkcionalnosti detaljno, a jedan deo razvijenog koda može ostati netestiran.



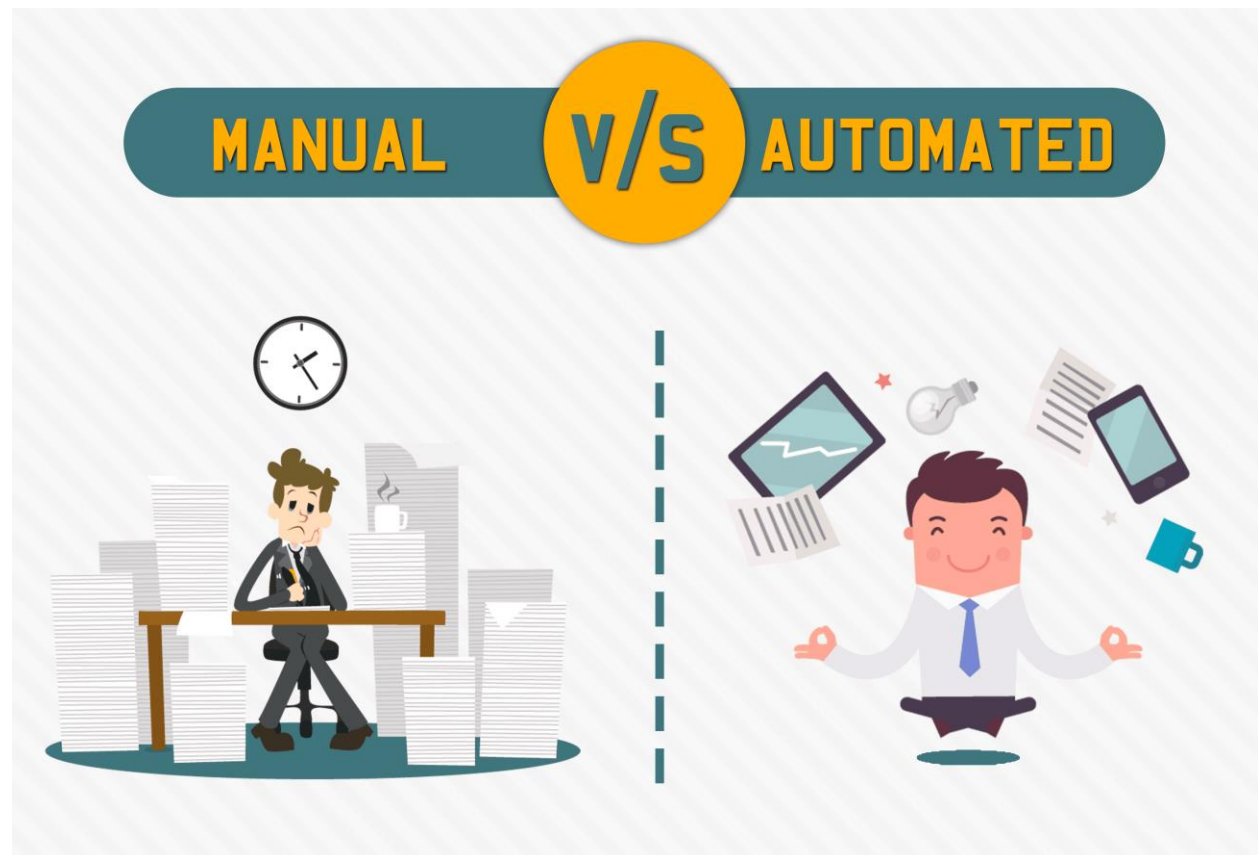
## White Box

WhiteBox testiranje je metod softverskog testiranja u kojem su unutrašnja struktura i svi implementirani detalji sistema koje obuhvata testiranje, u potpunosti poznati testeru. Prednost ovog načina testiranja je što testovi koriste ulaze kako bi pokrili sva grananja u kodu. Na taj način redundantnost je smanjena na minimum a pokrivenost test slučajeva je velika. Mana je da su testovi jako kompleksni i potrebno je da ih piše i testira veoma stručan tester koji dobro poznaje razvoj softvera. Takođe, ovaj metod podrazumeva detaljnu analizu, što podrazumeva puno utrošenog vremena.



## Grey Box

GreyBox testiranje je metod softverskog testiranja koji kombinuje BlackBox i WhiteBox metode. U ovoj metodologiji unutrašnja struktura sistema je delimično poznata testeru a tester poznaje strukturu podataka i algoritme koji su korišćeni za razvijanje softvera. Primer ovog metoda je analiza koda testirane aplikacije, formiranje funkcionalnih test scenarija na osnovu te analize i kasnijim testiranjem primenom BlackBox metoda testiranja. Prednost ovog načina testiranja jeste da se sa manjim brojem testova mogu pronaći potencijalne greške u odnosu na primenu BlackBox metoda testiranja. Ograničenje GreyBox metode je što pojedini delovi interne strukture moraju biti poznati.



## Nivoi Testiranja:

### “Unit” Testovi

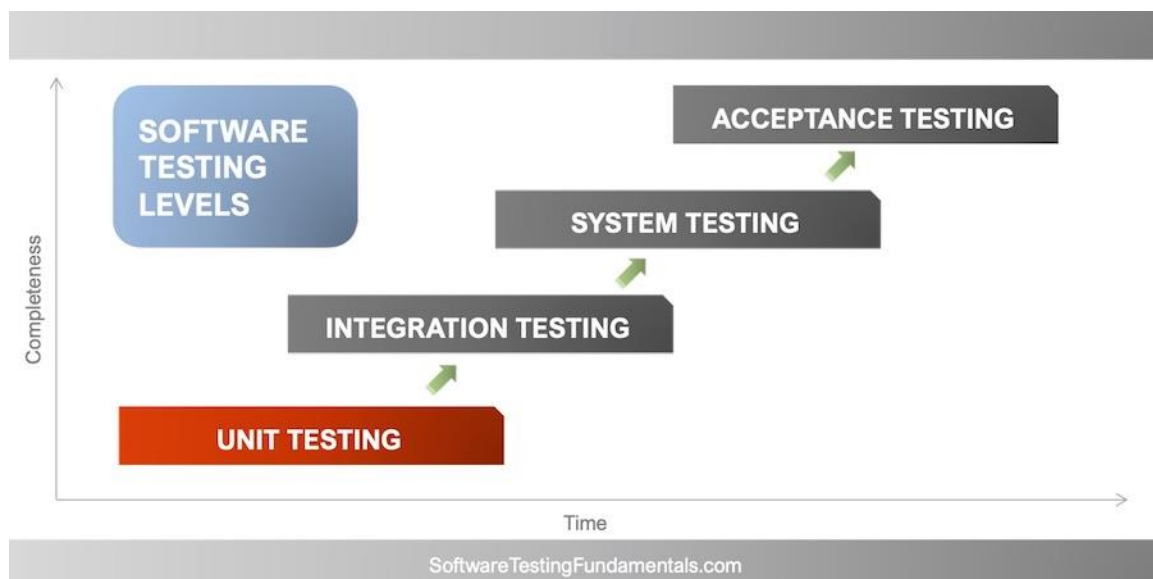
Testiranje osnovnih delova razvojnog koda obezbeđuje da dizajnirane modularne funkcionalnosti rade ispravno. To omogućava detekciju defekata u ranim fazama razvoja. Ovi osnovni delovi razvojnog koda ne mogu uvek da se testiraju bez upotrebe odgovarajućih drajvera ili alata. Za ovaj nivo testiranja se najčešće koristi WhiteBox metod, što znači da je unutrašnja struktura i implementacija dobro poznata test timu.

### Integraciono

Integraciono testiranje pruža proveru da li interakcija između implementiranih funkcija funkcioniše kako je i specificirano zahtevom. To bi bila provera da su dva ili više modula ispravno integrisana a njihova interakcija funkcioniše na planirani način. To se obično radi posle testiranja pojedinačnih modula i ovim testiranjem se mogu pronaći potencijalni problemi interakcije pojedinačnih komponenti aplikacije.

### Sistemska

Testiranje sistema obezbeđuje proveru funkcionalnosti celokupnog sistema kako bi potvrdilo da sve funkcionalnosti rade kao što je očekivano i da kvalitet sistema ispunjava sve ispostavljene zahteve. Ova aktivnost podrazumeva testiranje različitih aspekata ponašanja sistema: stabilnost - u trenucima neočekivanog povećanja opterećenja, pouzdanost - kada je sistem u recovery ili failure modu, bezbednost - od potencijalnog hakovanja sistema i performanse - vremenski odziv sistema kada je opterećen različitim nivoima opterećenja. Bilo koje od ovih karakteristika mogu da se testiraju istovremeno sa drugim karakteristikama ili individualno.



## **Tipovi Testiranja:**

### **Bezbednosno Testiranje**

Bezbednosno testiranje podrazumeva izvršavanje nefunkcionalnih tipova testova kako bi se otkrili sigurnosni nedostaci. Ono se radi kako bi se sprečili različiti hakerski napadi. Ovaj tip testiranja je obavezan za bilo koju aplikaciju koja je dostupna na Internetu.

### **Testiranje Performansi**

Testiranje performansi obuhvata obavljanje simulacije različitih opterećenja nad sistemom kako bi se konstatovalo kako sistem radi u smislu njegovog odziva i stabilnosti. Postoji mnogo podvrsta testiranja performansi kao što su Test opterećenja (kada se opterećenje povećava u normalnim uslovima), Stress test (kada su opterećenja izvan granica normale), Spike testiranje (kad se preopterećenje povećava samo u datom trenutku), itd. Rezultat izvršavanja ovih testova se obično daje u formi izveštaja koji analizira kapacitet servera.

### **Regresivno Testiranje**

Regresivno testiranje podrazumeva obavljanje funkcionalnog i nefunkcionalnog testiranja nad već testiranim delovima softvera kada su razvijene nove funkcionalnosti ili izvršeno unapređenje razvojnog koda. Ovaj proces se obavlja kako bi se osiguralo da promena razvojnog koda nema neželjene efekte na stare funkcionalnosti. U svrhu izvršavanja ovih testova, ne pišu se novi, nego se koriste već postojeći testovi. Ako se često menja razvojni kod aplikacije, resursno bi bilo zahtevno da se iznova manuelno izvršavaju svi test slučajevi tako da se test slučajevi automatizuju kako bi se uštedelo na vremenu i angažovanim resursima. Automatizacija u svrhu regresionog testiranja se ne radi samo za testove najvećeg prioriteta, nego i za testove čije je manuelno testiranje izuzetno vremenski i resursno zahtevno.

### **Statičko Testiranje**

Statičko testiranje obuhvata testiranje softverskog zahteva. Planirane aplikativne funkcionalnosti se analiziraju na postojanje nejasnoća, dvosmislenosti, kao i da li se uopšte mogu testirati. Svaki zahtevani detalj koji može dovesti do pogrešnog tumačenja je razjašnjen. Ovaj proces je poznat kao statična analiza.

### **Smoke Testiranje**

Smoke testiranje obuhvata niz jednostavnih testova koji za cilj imaju potvrdu rada najvažnijih funkcija sistema. Ovaj tip testiranja se radi kako bi se ispitalo da li je sistem dovoljno stabilan da podrži dalje testiranje tako da se ne ulazi u "dublji" test ako ključne funkcije ne rade.

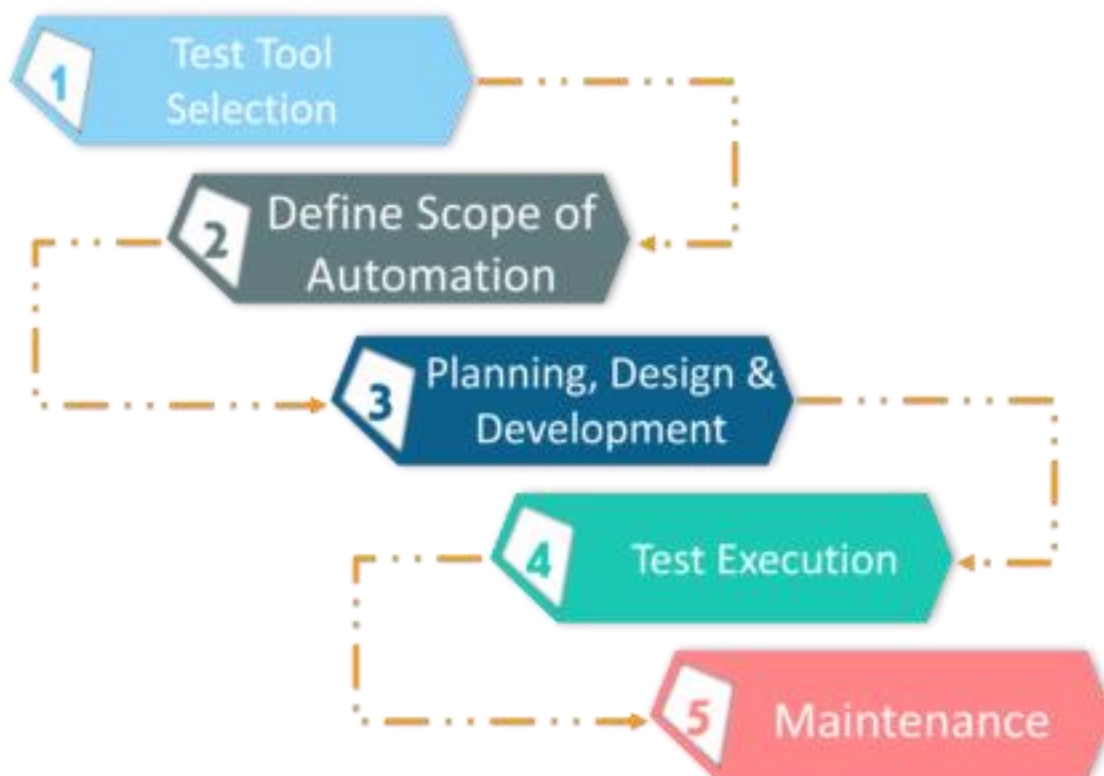
## Funkcionalno Testiranje

Funkcionalno testiranje obuhvata proveru da li je softver razvijen u skladu sa funkcionalnom specifikacijom. Formiraju se detaljni test scenariji i koriste se kao osnova za čitav proces funkcionalnog testiranja - najčešće BlackBox metodologijom. Rezultat testiranja se dostavlja u formi izveštaja koji tačno navodi koje funkcije ne rade u skladu sa aplikativnim zahtevom.

## Testiranje Upotrebljivosti

Bilo da se neki softverski proizvod koristi na više platformi i različitim uređajima (desktop, web, Android, iOS, itd.) ili ne, važno je proveriti da li svi korisnici mogu da ga koriste lako i jednostavno. Ovaj proces je poznat kao testiranje upotrebljivosti. Ako korisnik ne može lako da koristi aplikaciju postoji velika šansa da će koristiti neku drugu konkurentu aplikaciju. Ovaj tip testiranja je obavezan kad je reč o web i mobilnim aplikacijama.

### Koraci automatizacije:



Najbolji alati za Automatsko Testiranje 2020. godine:

# Best Automation Testing Tools for 2020



 Katalon Studio



## Literatura:

Softversko Inženjerstvo Teorija i Praksa

Automatizovano QA testiranje - Obuke i kursevi - [obukeikursevi.com](http://obukeikursevi.com)

DTC – Testiranje Softvera - [dtc.rs/sr/testiranje-softvera](http://dtc.rs/sr/testiranje-softvera)

Šta je to automatsko testiranje softvera? -

[internetzanatlija.com/2020/01/19/sta-je-to-automatsko-testiranje-softvera/](http://internetzanatlija.com/2020/01/19/sta-je-to-automatsko-testiranje-softvera/)

Wikipedia - Testiranje Softvera - [sr.wikipedia.org/sr-ec/Тестирање\\_софтвера](http://sr.wikipedia.org/sr-ec/Тестирање_софтвера)

## Sadržaj

|                                                             |    |
|-------------------------------------------------------------|----|
| Naslov.....                                                 | 1  |
| Šta je to automatizovano testiranje sistema? .....          | 2  |
| Prednosti i Mane Automatizacije.....                        | 3  |
| Alati za automatizaciju i Tehnike Testiranja .....          | 4  |
| Nivoi Testiranja .....                                      | 6  |
| Tipovi Testiranja.....                                      | 7  |
| Koraci Automatizacije .....                                 | 8  |
| Najbolji alati za Automatsko Testiranje 2020. godine: ..... | 9  |
| Literatura i Sadržaj .....                                  | 10 |