

Zadaća br. 4

Izveštaj o *black box* testiranju

Uputstvo za izradu zadaće

Izrada zadaće vrši se u formi izvještaja koja je data u nastavku. Potrebno je popuniti sva polja data u izvještaju, odgovoriti na pitanja i dodati tražene slike. Nije dozvoljeno brisati postojeća, niti dodavati nova polja.

Zadaća se radi u timovima od po tri studenta. Svi studenti iz istog tima popunjavaju isti izvještaj u jednom dokumentu, s tim da popunjavaju različite dijelove dokumenta ovisno o postavkama zadataka. Dovoljno je da jedan član tima pošalje izvještaj preko Zamgera.

Informacije o timu

Popuniti informacije o studentima koji vrše izradu zadaće.

Dodijeljeno programsko rješenje: KvizOdgovor

Ime i prezime: Berina Suljić
Broj indexa: 18385

Ime i prezime: Elma Šeremet
Broj indexa: 18318

Ime i prezime: Amina Šiljak
Broj indexa: 18496

Zadatak 1. (Funkcionalno testiranje)

Pairwise testiranje – član tima: Elma Šeremet

Za korisnički interfejs aplikacije dodijeljen nastavnoj grupi tima potrebno je kreirati tabelu testnih slučajeva koristeći pairwise testiranje. Prvo je potrebno kreirati tabelu particija, zatim je sortirati, a zatim provesti algoritam pairwise testiranja sa optimalnim brojem redova.

Tabela particija:

Particija	Ulazne varijable									
	Vrsta akcije	Ime	Godište	Korisnička grupa	Dodatni odgovor	Email	Novosti	Broj telefona	Recenzija	Odgovor
Ispravno	Dodavanje odgovora, dodavanje muzike za motivaciju	Barem 5 ili više karaktera	Vrijednost između 1800 i 2020 uključujući obe	Imate manje od 25 godina, Između 25 i 45, Između 45 i 75, više od 75 godina, Najvjerojatnije nesipravan datum jer više od 120 godina	Ovo nije moj prvi odgovor ili neoznačeno	Format email adrese (local-part @ domain)	Primaj novosti preko emaila ili neoznačeno	Barem 5 ili više karaktera	Ne želim ostaviti recenziju, Kviz je veoma zanimljiv, Kviz mi se ne sviđa	Barem 5 ili više karaktera
Neispravno	Sve ostale vrijednosti	Sve ostale vrijednosti (4 i manje)	Sve ostale vrijednosti	Sve ostale vrijednosti	Sve ostale vrijednosti	Sve ostale vrijednosti	Sve ostale vrijednosti	Sve ostale vrijednosti (4 i manje)	Sve ostale vrijednosti	Sve ostale vrijednosti (4 i manje)

Sortirana tabela particija:

Particija	Ulazne varijable									
	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor	Email	Novosti	Broj telefona	Vrsta akcije
1	Manje od 25 godina	Ne želim ostaviti recenziju	Barem 5 ili više karaktera	Barem 5 ili više karaktera	Između 1800 i 2020	Ovo nije moj prvi odgovor	Format local-part @ domain	Primaj novosti preko emaila	Barem 5 ili više karaktera	Dodavanje odgovora
2	Između 25 i 45 godina	Kviz je veoma zanimljiv	Ostalo	Ostalo	Ostalo	Neoznačeno	Ostalo	Neoznačeno	Ostalo	Dodavanje muzike
3	Između 45 i 75 godina	Kviz mi se ne sviđa	/	/	/	/	/	/	/	/
4	Više od 75 godina	/	/	/	/	/	/	/	/	/
5	Više od 120 godina	/	/	/	/	/	/	/	/	/

Sada je potrebno provesti algoritam pairwise testiranja. Potrebno je prikazati sve iteracije algoritma, kao i finalnu tabelu testnih slučajeva. Dozvoljeno je koristiti bilo koji način označavanja particija (brojevi, slova, boje, originalni nazivi i sl.) uz vođenje računa o tome da broj particija bude ispravan, kao i način njihovog kombinovanja.

Verifikacija i Validacija Softvera

U nastavku je data samo prva tabela testnih slučajeva, koju je potrebno kopirati i nadograđivati za svaku posebnu iteraciju, tako da se u konačnom dokumentu nalaze sve tabele iz svih iteracija pairwise postupka.

Iteracija 1

Tabela testnih slučajeva – Iteracija 1		
Kombinacija	Korisnička grupa	
1	Manje od 25 godina	A
2	Između 25 i 45 godina	B
3	Između 45 i 75 godina	C
4	Više od 75 godina	D
5	Više od 120 godina	E

Iteracija 2

Tabela testnih slučajeva – Iteracija 2				
Kombinacija	Korisnička grupa		Recenzija	
1	Manje od 25 godina	A	Ne želim ostaviti recenziju	A
2	Manje od 25 godina	A	Kviz je veoma zanimljiv	B
3	Manje od 25 godina	A	Kviz mi se ne sviđa	C
4	Između 25 i 45 godina	B	Ne želim ostaviti recenziju	A
5	Između 25 i 45 godina	B	Kviz je veoma zanimljiv	B
6	Između 25 i 45 godina	B	Kviz mi se ne sviđa	C
7	Između 45 i 75 godina	C	Ne želim ostaviti recenziju	A
8	Između 45 i 75 godina	C	Kviz je veoma zanimljiv	B
9	Između 45 i 75 godina	C	Kviz mi se ne sviđa	C
10	Više od 75 godina	D	Ne želim ostaviti recenziju	A
11	Više od 75 godina	D	Kviz je veoma zanimljiv	B
12	Više od 75 godina	D	Kviz mi se ne sviđa	C
13	Više od 120 godina	E	Ne želim ostaviti recenziju	A
14	Više od 120 godina	E	Kviz je veoma zanimljiv	B
15	Više od 120 godina	E	Kviz mi se ne sviđa	C

Odavde nastaviti postupak do dostizanja finalne tabele testnih slučajeva.

Iteracija 3

Tabela testnih slučajeva – Iteracija 3			
Kombinacija	Korisnička grupa	Recenzija	Odgovor
1	A	A	A
2	A	B	B
3	A	C	A
4	B	A	B
5	B	B	A
6	B	C	B
7	C	A	A
8	C	B	B
9	C	C	A
10	D	A	B
11	D	B	A
12	D	C	B
13	E	A	A
14	E	B	B
15	E	C	A

Iteracija 4

Tabela testnih slučajeva – Iteracija 4				
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime
1	A	A	A	A
2	A	B	B	B
3	A	C	A	B
4	B	A	B	A
5	B	B	A	A
6	B	C	B	B
7	C	A	A	A
8	C	B	B	B
9	C	C	A	A
10	D	A	B	B
11	D	B	A	A
12	D	C	B	B
13	E	A	A	A
14	E	B	B	B
15	E	C	A	A

Iteracija 5

Tabela testnih slučajeva – Iteracija 5					
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište
1	A	A	A	A	B
2	A	B	B	B	A
3	A	C	A	B	A
4	B	A	B	A	B
5	B	B	A	A	A
6	B	C	B	B	B
7	C	A	A	A	A
8	C	B	B	B	B
9	C	C	A	A	A
10	D	A	B	B	B
11	D	B	A	A	A
12	D	C	B	B	B
13	E	A	A	A	A
14	E	B	B	B	B
15	E	C	A	A	A

Iteracija 6

Tabela testnih slučajeva – Iteracija 6						
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor
1	A	A	A	A	B	A
2	A	B	B	B	A	B
3	A	C	A	B	A	A
4	B	A	B	A	B	B
5	B	B	A	A	A	B
6	B	C	B	B	B	A
7	C	A	A	A	A	A
8	C	B	B	B	B	B
9	C	C	A	A	A	A
10	D	A	B	B	B	B
11	D	B	A	A	A	A
12	D	C	B	B	B	B
13	E	A	A	A	A	A
14	E	B	B	B	B	B
15	E	C	A	A	A	A

Iteracija 7

Tabela testnih slučajeva – Iteracija 7							
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor	Email
1	A	A	A	A	B	A	A
2	A	B	B	B	A	B	B
3	A	C	A	B	A	A	A
4	B	A	B	A	B	B	B
5	B	B	A	A	A	B	A
6	B	C	B	B	B	A	B
7	C	A	A	A	A	A	B
8	C	B	B	B	B	B	A
9	C	C	A	A	A	A	A
10	D	A	B	B	B	B	B
11	D	B	A	A	A	A	A
12	D	C	B	B	B	B	B
13	E	A	A	A	A	A	A
14	E	B	B	B	B	B	B
15	E	C	A	A	A	A	A

Iteracija 8

Tabela testnih slučajeva – Iteracija 8								
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor	Email	Novosti
1	A	A	A	A	B	A	A	A
2	A	B	B	B	A	B	B	B
3	A	C	A	B	A	A	A	A
4	B	A	B	A	B	B	B	B
5	B	B	A	A	A	B	A	A
6	B	C	B	B	B	A	B	B
7	C	A	A	A	A	A	B	A
8	C	B	B	B	B	B	A	B
9	C	C	A	A	A	A	A	B
10	D	A	B	B	B	B	B	A
11	D	B	A	A	A	A	A	A
12	D	C	B	B	B	B	B	B
13	E	A	A	A	A	A	A	A
14	E	B	B	B	B	B	B	B
15	E	C	A	A	A	A	A	A

Iteracija 9

Tabela testnih slučajeva – Iteracija 9									
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor	Email	Novosti	Broj telefona
1	A	A	A	A	B	A	A	A	A
2	A	B	B	B	A	B	B	B	B
3	A	C	A	B	A	A	A	A	A
4	B	A	B	A	B	B	B	B	B
5	B	B	A	A	A	B	A	A	A
6	B	C	B	B	B	A	B	B	B
7	C	A	A	A	A	A	B	A	A
8	C	B	B	B	B	B	A	B	B
9	C	C	A	A	A	A	A	B	A
10	D	A	B	B	B	B	B	A	B
11	D	B	A	A	A	A	A	A	B
12	D	C	B	B	B	B	B	B	A
13	E	A	A	A	A	A	A	A	A
14	E	B	B	B	B	B	B	B	B
15	E	C	A	A	A	A	A	A	A

Iteracija 10

Tabela testnih slučajeva – Iteracija 10										
Kombinacija	Korisnička grupa	Recenzija	Odgovor	Ime	Godište	Dodatni odgovor	Email	Novosti	Broj telefona	Vrsta akcije
1	A	A	A	A	B	A	A	A	A	A
2	A	B	B	B	A	B	B	B	B	B
3	A	C	A	B	A	A	A	A	A	A
4	B	A	B	A	B	B	B	B	B	B
5	B	B	A	A	A	B	A	A	A	A
6	B	C	B	B	B	A	B	B	B	B
7	C	A	A	A	A	A	B	A	A	A
8	C	B	B	B	B	B	A	B	B	B
9	C	C	A	A	A	A	A	B	A	A
10	D	A	B	B	B	B	B	A	B	B
11	D	B	A	A	A	A	A	A	B	A
12	D	C	B	B	B	B	B	B	A	B
13	E	A	A	A	A	A	A	A	A	B
14	E	B	B	B	B	B	B	B	B	A
15	E	C	A	A	A	A	A	A	A	A

Na ovaj način definisan je minimalni broj testnih slučajeva za black box testiranje interfejsa koji iznosi 15 (što je neuporedivo manje od ukupno 3840 testnih slučajeva).

Ispravnost formirane tabele testnih slučajeva potrebno je provjeriti koristeći online alat CTWedge, ili neki drugi alat po izboru.

Alat odabran za formiranje testnih slučajeva: Pairwise Pict

Prikaz unosa ulaznih varijabli i njihovih particija:

```
#####  
# Paste test factors here.  
# Check the documents for more details.  
# https://github.com/Microsoft/pict/blob/master/doc/pict.md  
#####  
  
korisnickaGrupa : manjeOd25, izmedju25i45, izmedju45i75, viseOd75, viseod120  
recenzija : neZelim, zanimljiv, neSvidja  
odgovor: viseOd5, ostalo  
ime: viseOd5, ostalo  
godiste: izmedju1800i2020, ostalo  
dodaniOdgovor: nijePrvi, neoznaceno  
email: domen, ostalo  
novosti: primaj, neoznaceno  
brojTelefona: viseOd5, ostalo  
vrstaAkcije: odgovor, muzika
```

Prikaz izlaznih testnih slučajeva:

test-results - Notepad								
File Edit Format View Help								
korisnickaGrupa	recenzija	odgovor	ime	godiste	dodaniOdgovor	email	novosti	brojTelefona
viseod120	neZelim	ostalo	viseOd5	ostalo	neoznaceno	ostalo	odgovor	
manjeOd25	neSvidja	viseOd5	ostalo	izmedju1800i2020	nijePrvi	domen	primaj	viseOd5 muz:
izmedju25i45	neZelim	ostalo	ostalo	neoznaceno	domen	primaj	ostalo	muzika
izmedju25i45	neSvidja	viseOd5	viseOd5	izmedju1800i2020	nijePrvi	ostalo	neoznaceno	vis
viseod120	neSvidja	ostalo	ostalo	izmedju1800i2020	nijePrvi	domen	neoznaceno	ost
izmedju45i75	neSvidja	viseOd5	ostalo	ostalo	neoznaceno	ostalo	primaj	viseOd5 odgovor
viseOd75	zanimljiv	viseOd5	viseOd5	izmedju1800i2020	neoznaceno	ostalo	primaj	ostalo muz:
viseOd75	neZelim	ostalo	viseOd5	nijePrvi	domen	neoznaceno	viseOd5	odgovor
manjeOd25	neZelim	viseOd5	viseOd5	ostalo	neoznaceno	ostalo	odgovor	
viseOd75	neSvidja	ostalo	ostalo	ostalo	nijePrvi	domen	primaj	ostalo odgovor
izmedju45i75	zanimljiv	ostalo	viseOd5	izmedju1800i2020	nijePrvi	domen	neoznaceno	vis
izmedju25i45	zanimljiv	ostalo	ostalo	ostalo	nijePrvi	domen	primaj	ostalo odgovor
viseod120	zanimljiv	viseOd5	ostalo	izmedju1800i2020	neoznaceno	domen	primaj	viseOd5 odg
izmedju45i75	neZelim	ostalo	ostalo	izmedju1800i2020	neoznaceno	domen	primaj	ostalo muzika
manjeOd25	zanimljiv	ostalo	viseOd5	ostalo	neoznaceno	domen	neoznaceno	ostalo odgovor

Verifikacija i Validacija Softvera

Da li je dobiven isti broj testnih slučajeva kao pri ručnom provođenju *pairwise* testiranja? Da li je struktura testnih slučajeva ista i zašto? Ukoliko nije, da li to znači da jedan od provedenih postupaka nije ispravan? Obrazložiti svoj odgovor.

Dobila sam jednak rezultat kao i pri ručnom izvođenju *pairwise* testiranja. Struktura testnih slučajeva ne mora biti ista, naprimjer prateći logiku kojom sam išla, možda bi bilo logičnije da sam krenula otpočetak mijenjati A i B, odnosno izbor kombinacija u četvrtoj i petoj iteraciji da je obrnut bio bi možda logičniji, ali je sasvim svejedno, te budući da sam krenula na taj način na papiru, ostavila sam tako i ovdje. Oba postupka su ispravna.

Cause-effect graphing testiranje – član tima: Berina Suljić

Za korisnički interfejs aplikacije dodijeljen nastavnoj grupi tima potrebno je kreirati tabelu testnih slučajeva koristeći cause-effect grafove. Prvo je potrebno identificirati uzroke i posljedice, a zatim kreirati logičke uslove koji kodiraju kombinacije testnih slučajeva. Naposljetku je potrebno provesti algoritam cause-effect graphing testiranja sa optimalnim brojem kolona.

Uzroci:

- C1 – Godište manje od 1900
- C2 – Godište manje od 1945
- C3 – Godište manje od 1975
- C4 – Godište manje od 1995

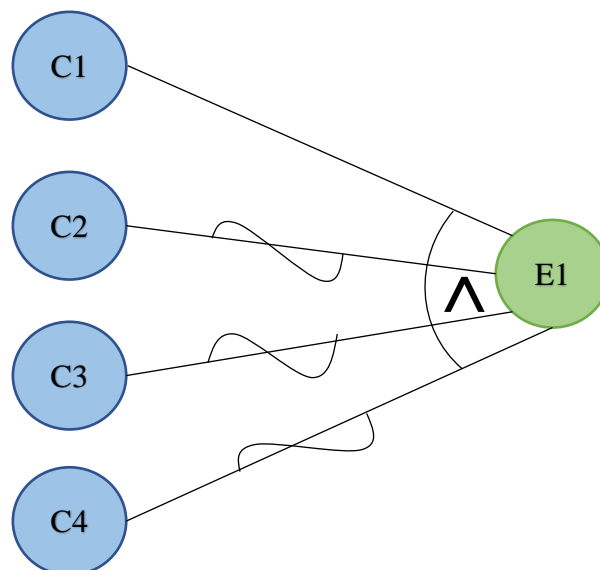
Posljedice:

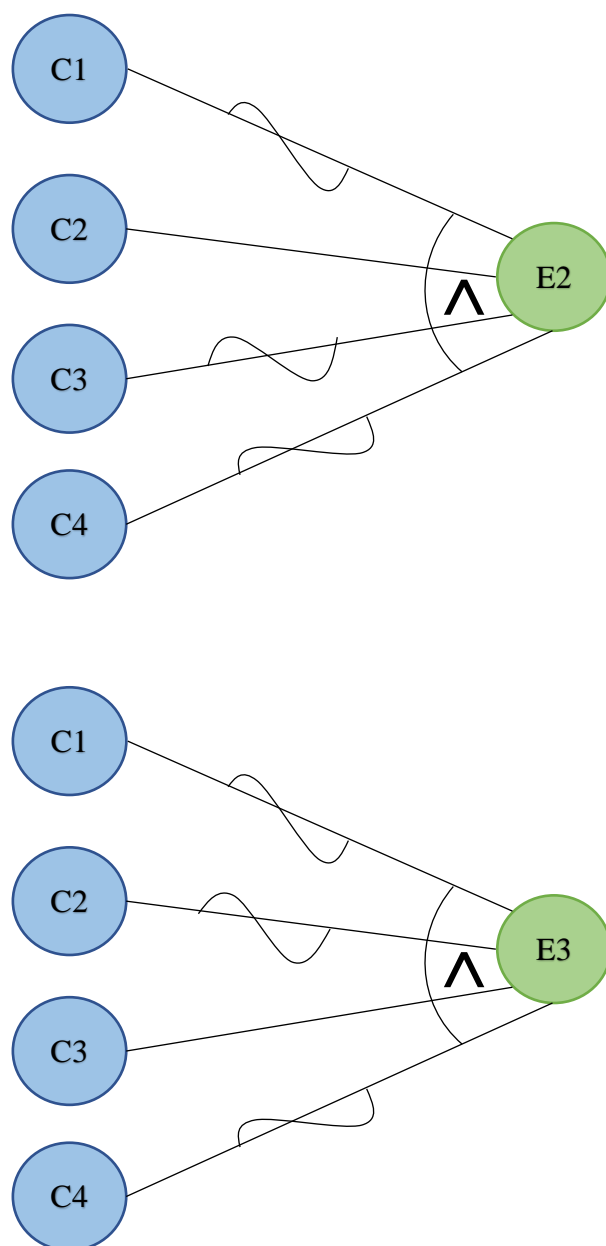
- E1 – prikaz poruke "Najvjerojatnije niste ispravno unijeli godište - imate više od 120 godina!"
- E2 – "Imate više od 75 godina!"
- E3 – prikaz poruke "Imate između 45 i 75 godina!"
- E4 – prikaz poruke "Imate između 25 i 45 godina!"
- E5 – prikaz poruke "Imate manje od 25 godina!"

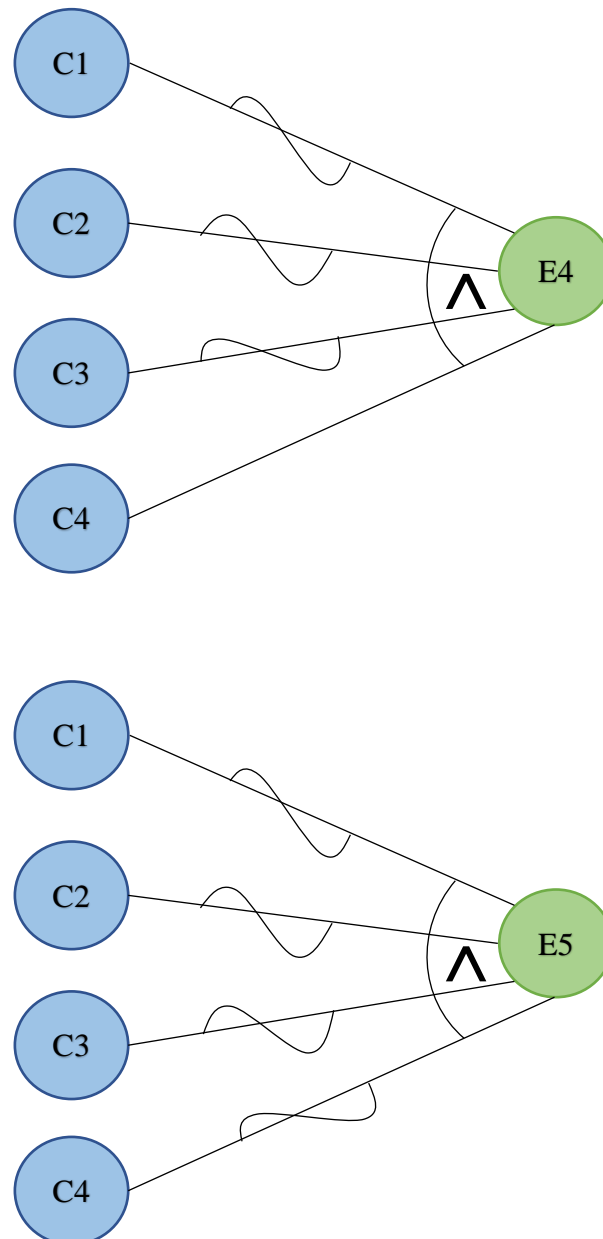
Svi logički uslovi:

$$\begin{aligned}E1 &= C1 \wedge \overline{C2} \wedge \overline{C3} \wedge \overline{C4} \\E2 &= \overline{C1} \wedge C2 \wedge \overline{C3} \wedge \overline{C4} \\E3 &= \overline{C1} \wedge \overline{C2} \wedge C3 \wedge \overline{C4} \\E4 &= \overline{C1} \wedge \overline{C2} \wedge \overline{C3} \wedge C4 \\E5 &= \overline{C1} \wedge \overline{C2} \wedge \overline{C3} \wedge \overline{C4}\end{aligned}$$

Podgrafovi logičkih uslova (član tima: Berina Suljić):







Sada je potrebno provesti algoritam cause-effect graphing testiranja. Potrebno je prikazati sve iteracije algoritma, kao i finalnu tabelu testnih slučajeva.

U nastavku je data samo prva tabela testnih slučajeva, koju je potrebno kopirati i nadograđivati za svaku posebnu iteraciju, tako da se u konačnom dokumentu nalaze sve tabele iz svih iteracija cause-effect graphing postupka.

Iteracija 1

Iskaz	T1
C1	1
C2	0
C3	0
C4	0
E1	1
E2	0
E3	0
E4	0
E5	0

Iteracija 2

Iskaz	T1	T2
C1	1	0
C2	0	1
C3	0	0
C4	0	0
E1	1	0
E2	0	1
E3	0	0
E4	0	0
E5	0	0

Iteracija 3

Iskaz	T1	T2	T3
C1	1	0	0
C2	0	1	0
C3	0	0	1
C4	0	0	0
E1	1	0	0
E2	0	1	0
E3	0	0	1
E4	0	0	0
E5	0	0	0

Iteracija 4

Iskaz	T1	T2	T3	T4
C1	1	0	0	0
C2	0	1	0	0
C3	0	0	1	0
C4	0	0	0	1
E1	1	0	0	0
E2	0	1	0	0
E3	0	0	1	0
E4	0	0	0	1
E5	0	0	0	0

Iteracija 5

Iskaz	T1	T2	T3	T4	T5
C1	1	0	0	0	0
C2	0	1	0	0	0
C3	0	0	1	0	0
C4	0	0	0	1	0
E1	1	0	0	0	0
E2	0	1	0	0	0
E3	0	0	1	0	0
E4	0	0	0	1	0
E5	0	0	0	0	1

Kreirani testni slučajevi predstavljaju minimalni broj testnih slučajeva za potpunu pokrivenost svih kombinacija vrijednosti ulaznih varijabli.

Da li je dobiven isti broj testnih slučajeva kao pri *pairwise* testiranju? Da li je struktura testnih slučajeva ista i zašto? Ukoliko nije, da li to znači da jedan od provedenih postupaka nije ispravan? Obrazložiti svoj odgovor.

Nije dobiven isti broj testnih slučajeva kao pri *pairwise* testiranju, jer ovdje je minimalan broj testnih slučajeva dovoljnih za potpunu pokrivenost svih kombinacija vrijednosti ulaznih varijabli. Struktura testova je ista, samo je ovdje napravljen minimalan broj testnih slučajeva.

Metod ekvivalentnih klasa – član tima: Amina Šiljak

Za korisnički interfejs aplikacije dodijeljen nastavnoj grupi tima potrebno je kreirati tabelu testnih slučajeva koristeći ekvivalentne klase. Prvo je potrebno samostalno odrediti scenarij prema kojem se objektu koji se koristi na korisničkom interfejsu dodjeljuju različiti ratingi ovisno o vrijednostima atributa, a zatim identificirati ekvivalentne klase i iskoristiti ih za formiranje tabele testnih slučajeva.

Scenarij dodjele ratinga objektu sa objašnjenjem na koji način svi atributi utječu na vrijednost ratinga:

Rating zavisi od godišta, da li je to korisnikov prvi odgovor te da li je odgovor tačan ili ne. Ukoliko je odgovor tačan, dobija se 5 bodova, a ukoliko je netačan -1. Na to se nadodaju dva boda ukoliko je korisnikovo godište u rang 2006-2020, i bod ukoliko je godište između 2000 i 2006.

U nastavku je data tabela ekvivalentnih klasa, koju je potrebno popuniti u skladu s prethodno kreiranim scenarijem. Potrebno je popuniti sve vrijednosti za sve ulazne varijable uz ispravno definisanje svih particija za sve ekvivalentne klase.

Ulazna varijabla	Validno			Nevalidno	
	Ekvivalentne klase	Vrijednosti	Granične vrijednosti	Ekvivalentne klase	Vrijednosti
Godište	(1) 2006-2020 (2) 2000-2006 (3) 1800-2000	(1) 2010 (2) 2003 (3) 1999	(1) 2006, 2020 (2) 2000, 2006 (3) 1800, 2020	Svaka alfanumerička vrijednost koja nije starosna dob ili nije između 1800 i 2020 uključivo	1700
Odgovor na kviz pitanje	(1) trinaesta, trinaesta sedmica, 13. sedmica (2) ostalo	13. sedmica		Svaka alfanumerička vrijednost sa manje od 5 karaktera	13.
Ime	(1) string sa pet ili više karaktera	Amina Šiljak		Svaka alfanumerička vrijednost koja ima manje od 5 karaktera	Ena

Verifikacija i Validacija Softvera

Email	(1) local-part@domain	asiljak1@etf.uns.a.ba		Svaka alfanumerička vrijednost koja ne odgovara formatu local-part@domain	asiljak1@
Novosti	(1) Primaj novosti preko maila, neoznačeno	Primaj novosti preko maila			
Broj telefona	(1) string sa pet ili više karaktera	061-111-111		Svaka alfanumerička vrijednost koja ima manje od 5 karaktera	061
Recenzija	(1) Ne želim ostaviti recenziju, Kviz je veoma zanimljiv, Kviz mi se ne sviđa	Kviz je veoma zanimljiv			
Dodatni odgovor	(1) Ovo nije moj prvi odgovor, neoznačeno	Ovo nije moj prvi odgovor			

U nastavku je data tabela testnih slučajeva koju je potrebno popuniti na osnovu prethodno popunjene tabele ekvivalentnih klasa. Potrebno je da tabela ima optimalan broj testnih slučajeva.

Testni slučaj	Ulazne varijable							
	Godiš	Odgovor na kviz pitanje	Ime	Email	Novosti	Broj telefona	Recenzija	Dodatni odgovor
1	1999	trinaesta	Johnny	jdepp@gmail.com	Neoznačeno	+33 1 86 26 22 00	Kviz je veoma zanimljiv	neoznačeno

Verifikacija i Validacija Softvera

2	2003	trinaesta	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
3	2011	deseta	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
4	2020	13.sedm ica	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
5	2006	trinaesta	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
6	1800	14.sedm ica	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
7	2022	trinaesta	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
8	2015	13.	John ny	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
9	2015	Trinaest a	John	jdepp@gmail.com	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
10	2015	Trinaest a	John ny	jdepp@	Neoznač eno	+33 1 86 26 22 00	Kviz je veoma zanimlj iv	neoznače no
11	2015	trinaesta	John ny	jdepp@gmail.com	Neoznač eno	+331	Kviz je veoma zanimlj iv	neoznače no

Da li je dobiven isti broj testnih slučajeva kao pri korištenju prethodne dvije metode? Da li je struktura testnih slučajeva ista i zašto? Ukoliko nije, da li to znači da jedan od provedenih postupaka nije ispravan? Obrazložiti svoj odgovor.

Što se tiče broja testnih slučajeva, on je različit u sve tri metode. Struktura testnih slučajeva je ista.

Zadatak 2. (Automatizovano testiranje)

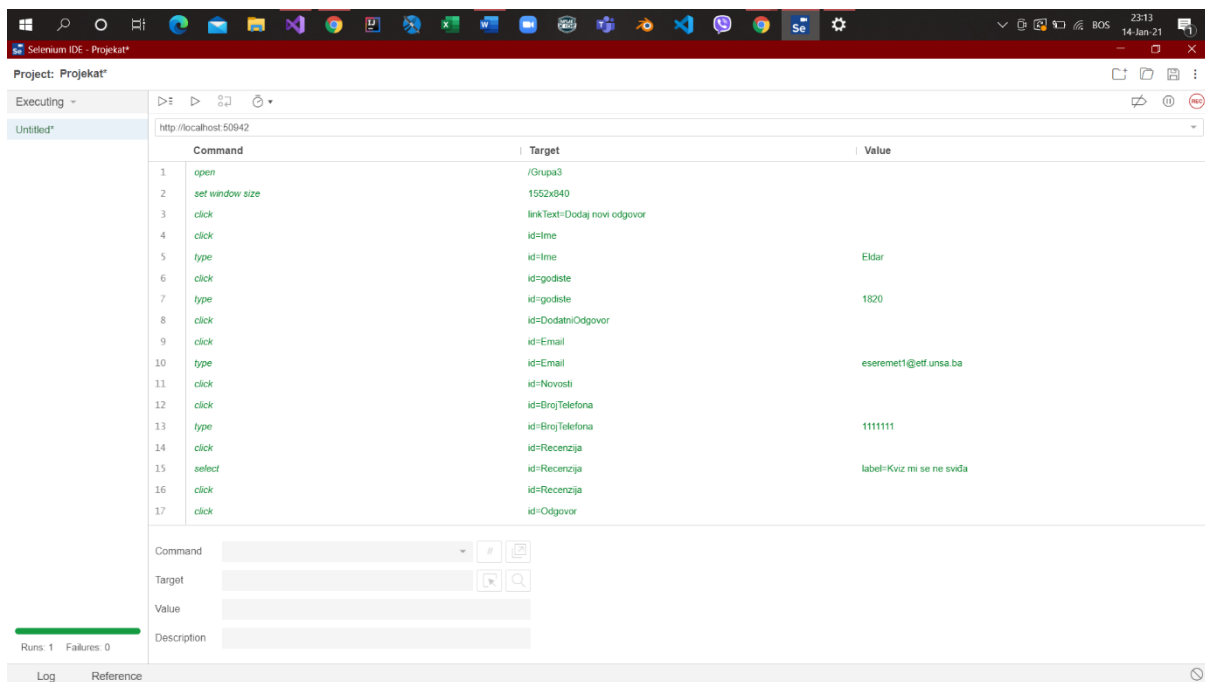
Na osnovu prethodno provedenih postupaka potrebno je izvršiti automatizovano testiranje korisničkog interfejsa aplikacije koristeći Selenium alat. Pritom nije dozvoljeno imati više od jednog testnog slučaja za sve validne i jednog testnog slučaja za sve nevalidne testne slučajeve.

Automatizacija pairwise testiranja – član tima: Elma Šeremet

Opis testnog slučaja za sve validne podatke (scenarij po kojem će se izvršiti testiranje):

S obzirom da postoji samo jedan slučaj sa svim validnim podacima, samo podaci iz tog slučaja su uneseni, a zatim je provjereno da li je ispod ijednog polja ispisana greška. Ukoliko nije ispisana nijedna greška prikaže se alert sa tekstom „Ispravno“.

Selenium skripta za test za sve validne podatke (prikaz nakon pokretanja, odakle se vidi da skripta uspješno prolazi):



Windows Taskbar

Selenium IDE - Projekat*

Project: Projekat*

Executing +

Untitled*

http://localhost:50942

Command	Target	Value
13 type	id=BrojTelefona	1111111
14 click	id=Recenzija	
15 select	id=Recenzija	label=Kviz mi se ne sviđa
16 click	id=Recenzija	
17 click	id=Odgovor	
18 type	id=Odgovor	Ovo je odgovor
19 execute script	return document.getElementById("ime-error") == null	notPresentime
20 execute script	return document.getElementById("godiste-error") == null	notPresentGodiste
21 execute script	return document.getElementById("Email-error") == null	notPresentEmail
22 execute script	return document.getElementById("BrojTelefona-error") == null	notPresentBrojTelefona
23 execute script	return document.getElementById("Odgovor-error") == null	notPresentOdgovor
24 if	\$(notPresentime) == true && \$(notPresentGodiste) == true && \$(notPresentEmail) == true && \$(notPresentBrojTelefona) == true && \$(notPresentOdgovor) == true	
25 execute script	alert("Ispravno")	
26 end		
27 close		

Command

Target

Value

Description

Runs: 1 Failures: 0

LogReference

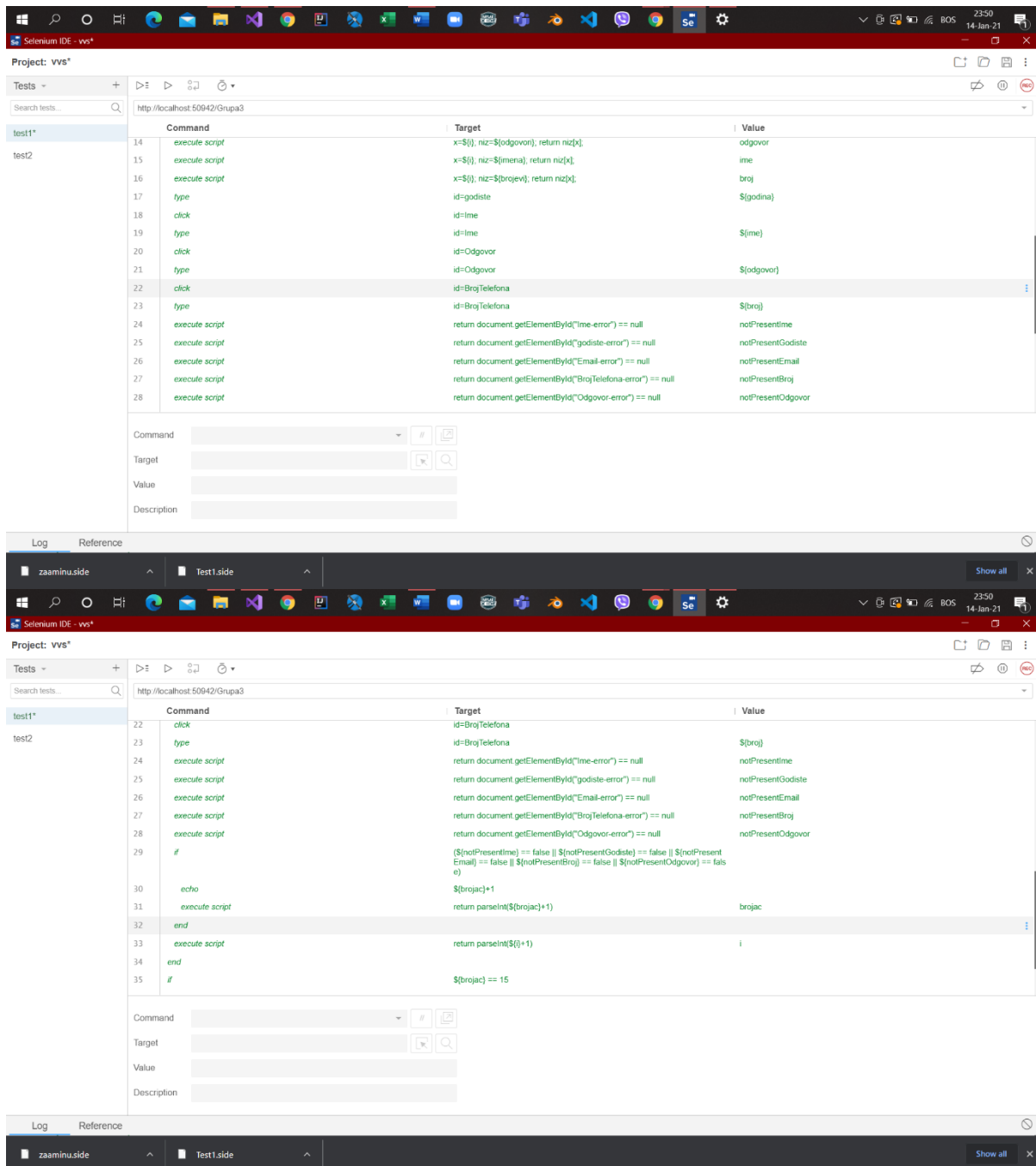
Selenium skripta za test za sve nevalidne podatke (prikaz nakon pokretanja, odakle se vidi da skripta uspješno prolazi):

The screenshot shows the Selenium IDE interface with a project named 'vvs'. The test suite 'test1*' contains two tests: 'test1' and 'test2'. The 'test1' test has 13 steps:

#	Command	Target	Value
1	open	http://localhost:50942/Grupa3	
2	set window size	1074x757	
3	execute script	return ["1700","2010","2011","2021","2005","1701","2005","2022","2004","1700","1699","1780","2000","1789","1696"]	godine
4	execute script	return ["dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset"]	odgovori
5	execute script	return ["dvadeset","tri","tri","dvadeset","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset"]	imena
6	execute script	return ["dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset","tri","dvadeset"]	brojevi
7	click	linkText=Grupa 3	
8	click	linkText=Dodaj novi odgovor	
9	store	0	brojac
10	store	0	i
11	while	parseInt(\$j)/=6	
12	click	id=godiste	
13	execute script	x=\${0}, niz=\${godine}, return niz[x];	godina

Below the table, there are input fields for Command, Target, Value, and Description.

Verifikacija i Validacija Softvera



The image displays two screenshots of the Selenium IDE interface, showing test scripts for a project named 'vvs*'. The top screenshot shows a manual script with commands like 'execute script' and 'click'. The bottom screenshot shows a scripted test using 'if', 'echo', and 'end' commands.

Test 1 (Manual Script):

Line	Command	Target	Value
14	execute script	x=\${i}, niz=\${odgovor}, return niz[x];	odgovor
15	execute script	x=\${i}, niz=\${imeraj}, return niz[x];	ime
16	execute script	x=\${i}, niz=\${brojevi}, return niz[x];	broj
17	type	id=godiste	\$(godina)
18	click	id=ime	
19	type	id=ime	\$(ime)
20	click	id=Odgovor	
21	type	id=Odgovor	\$(odgovor)
22	click	id=BrojTelefona	
23	type	id=BrojTelefona	\$(broj)
24	execute script	return document.getElementById("ime-error") == null	notPresentime
25	execute script	return document.getElementById("godiste-error") == null	notPresentGodiste
26	execute script	return document.getElementById("Email-error") == null	notPresentEmail
27	execute script	return document.getElementById("BrojTelefona-error") == null	notPresentBroj
28	execute script	return document.getElementById("Odgovor-error") == null	notPresentOdgovor

Test 2 (Scripted Test):

Line	Command	Target	Value
22	click	id=BrojTelefona	
23	type	id=BrojTelefona	\$(broj)
24	execute script	return document.getElementById("ime-error") == null	notPresentime
25	execute script	return document.getElementById("godiste-error") == null	notPresentGodiste
26	execute script	return document.getElementById("Email-error") == null	notPresentEmail
27	execute script	return document.getElementById("BrojTelefona-error") == null	notPresentBroj
28	execute script	return document.getElementById("Odgovor-error") == null	notPresentOdgovor
29	if	(\$(!notPresentime) == false (!\$notPresentGodiste) == false (!\$notPresentEmail) == false (!\$notPresentBroj) == false (!\$notPresentOdgovor) == false)	
30	echo	\$(brojac)+1	
31	execute script	return parseInt((\$(brojac)+1)	brojac
32	end		
33	execute script	return parseInt((\$(i)+1)	i
34	end		
35	if	\$(brojac) == 15	

Koji dio testova je snimljen, a koji manuelno skriptovan?

Čitav kod je manuelno skriptovan, osim početne dvije linije.

Automatizacija cause-effect graphing testiranja – član tima: Berina Suljić

Zbog činjenice da cause-effect slučajevi ne rezultuju neuspjehom, već prikazom različitih poruka, preporučuje se da se sve varijante ulaza i izlaza razmatraju kao kolekcije podataka.

Opis svih testnih slučajeva (scenarij po kojem će se izvršiti testiranje):

U ovom testu je istestirano 5 testnih slučajeva i u svakom su uneseni validni podaci:

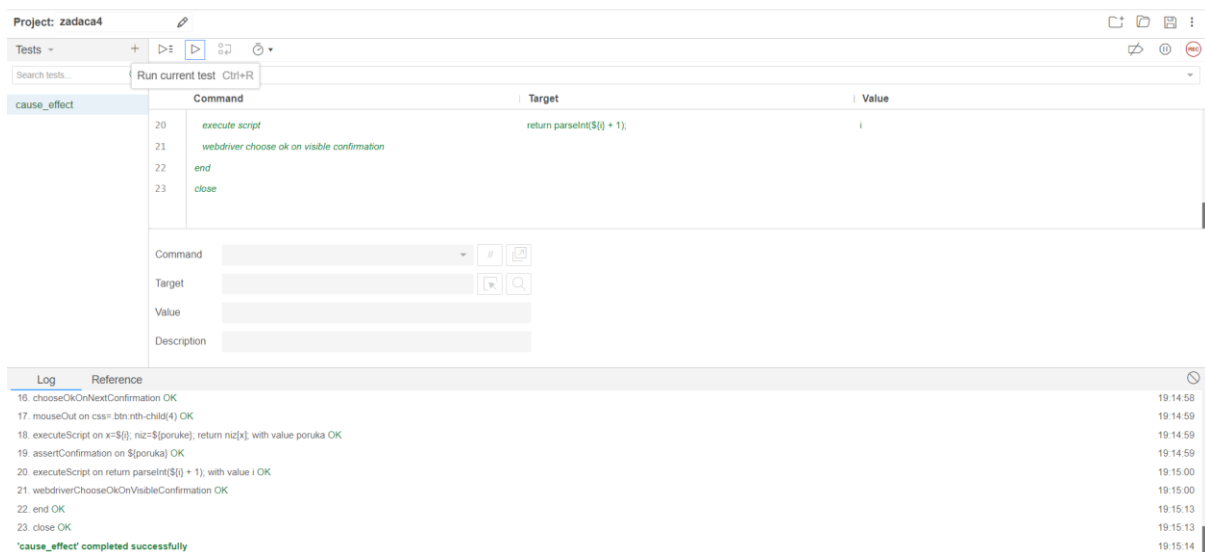
- ime Berina i godište 1800, koje treba prikazati poruku "Najvjerojatnije niste ispravno unijeli godište - imate više od 120 godina!"
- ime Amina i godište 1920, koje treba prikazati poruku "Imate više od 75 godina!"
- ime Elmica i godište 1950, koje treba prikazati poruku "Imate između 45 i 75 godina!"
- ime Lamija i godište 1990, koje treba prikazati poruku "Imate između 25 i 45 godina!"
- ime Nekoo i godište 2000, koje treba prikazati poruku "Imate manje od 25 godina!"

Podaci su unijeti kao kolekcija podataka, te su u for each petlji redom uneseni u korisnički interfejs.

Selenium skripta za test za sve podatke (prikaz nakon pokretanja, odakle se vidi da skripta uspješno prolazi):

http://localhost:50942			
	Command	Target	Value
1	open	/	
2	set window size	1552x840	
3	click	linkText=Grupa 3	
4	click	linkText=Dodaj novi odgovor	
5	execute script	return[["Berina", "1800"], ["Amina", "1920"], ["Elmica", "1950"], ["Lamija", "1990"], ["Nekoo", "2000"]]	podaci
6	execute script	return["Najvjerojatnije niste ispravno unijeli godište - imate više od 120 godina!", "Imate više od 75 godina!", "Imate između 45 i 75 godina!", "Imate između 25 i 45 godina!", "Imate manje od 25 godina!"]	poruke
7	store	0	i
8	for each	podaci	podatak
9	click	id=lme	
10	type	id=lme	\${podatak[0]}
11	click	css=body	
12	click	id=godiste	
13	type	id=godiste	\${podatak[1]}
14	click	css=.btn:nth-child(4)	
14	click	css=.btn:nth-child(4)	
15	mouse over	css=.btn:nth-child(4)	
16	choose ok on next confirmation		
17	mouse out	css=.btn:nth-child(4)	
18	execute script	x=\${i}; niz=\${poruke}; return niz[x];	poruka
19	assert confirmation	\${poruka}	
20	execute script	return parseInt(\${i} + 1);	i
21	webdriver choose ok on visible confirmation		
22	end		
23	close		

Verifikacija i Validacija Softvera



Step	Command	Target	Value
20	execute script	return parseInt(\$() + 1);	i
21	webdriver choose ok on visible confirmation		
22	end		
23	close		

Log	Reference
16. chooseOkOnNextConfirmation OK	19:14:58
17. mouseOut on css= btn:nth-child(4) OK	19:14:59
18. executeScript on x=\$(), niz=\$[poruke], return niz[i], with value poruka OK	19:14:59
19. assertConfirmation on \$[poruka] OK	19:14:59
20. executeScript on return parseInt(\$() + 1), with value i OK	19:15:00
21. webdriverChooseOkOnVisibleConfirmation OK	19:15:00
22. end OK	19:15:13
23. close OK	19:15:13
'cause_effect' completed successfully	

Koji dio testova je snimljen, a koji manuelno skriptovan?

Dio testova koji je snimljen jeste ulazak na stranicu i dolazak do određene forme. Potom se izvršio unos imena i unos godišta, te se kliknulo na dugme, nakon čega se pojavio alert gdje je potom kliknuto OK. Zatim se izašlo sa stranice.

Ono što je manuelno skriptovano, jeste korištenje kolekcije podataka, konkretno korištenje kolekcija podaci (u kojoj se nalaze svi podaci koji su uneseni, svaki podatak u sebi opet ima 2 podatka, gdje je prvi podatak ime, a drugi podatak godiste) i poruke (gdje su redom unesene poruke koje trebaju biti prikazane za određenu kombinaciju podataka). Potom, ručno je kreirana varijabla i. Kod koji je bio snimljen od trenutka unosa imena i godišta, pa sve do pred komandu close, ručno je stavljen unutar petlje koja prolazi kroz sve podatke. Vrijednosti target i value su promijenjene na osnovu izmjena koje su uvedene korištenjem kolekcije baze podataka. Također, skriptovan je i dio u kojem se dobija poruka kao varijabla, kao i povećavanje varijable i unutar svake iteracije petlje.

Automatizacija metode ekvivalentnih klasa – član tima: Amina Šiljak

Opis testnog slučaja za sve validne podatke (scenarij po kojem će se izvršiti testiranje):

Pošto 6 od 8 ulaznih podataka imaju samo jednu validnu ekvivalentnu klasu, u testovima za validne podatke uvijek imaju istu vrijednost i njima je unesena vrijednost prije ulaska u petlju. Za druge dvije varijable napravljena su dva niza: „godine“ i „odgovori“ iz kojih se pri svakom prolasku kroz petlju uzimaju vrijednosti i zatim se provjerava da li igdje na ekranu stoji poruka da je neka od vrijednosti nevalidna. Ako takva poruka ne postoji, povećava se varijabla „brojac“. Nakon izlaska iz petlje, provjerava se da li je spomenuta varijabla povećana pri svakom prolasku kroz petlju to jest da li su svi ulazni podaci validni, i ako jesu, ispisuje se poruka „Validni svi unosi!“

Selenium skripta za test za sve validne podatke (prikaz nakon pokretanja, odakle se vidi da skripta uspješno prolazi):

1	open	http://localhost:50942/Grupa3	
2	set window size	1074x757	
3	execute script	return ["1999", "2003", "2011", "2020", "2006", "1800"]	godine
4	execute script	return ["trinaesta", "trinaesta", "deseta", "13 sedmica", "trinaesta", "14 sedmica"]	odgovori
5	click	linkText=Grupa 3	
6	click	linkText=Dodaj novi odgovor	
7	store	0	brojac
8	store	0	i
9	click	id=Ime	
10	type	id=Ime	Johnny
11	click	id=Email	
12	type	id=Email	jdepp@gmail.com
13	click	id=BrojTelefona	
14	type	id=BrojTelefona	+33 1 86 26 22 00
15	click	id=Recenzija	
16	select	id=Recenzija	label=Kviz je veoma zanimljiv
17	while	parseInt(\$(i))!=6	
18	click	id=godiste	
19	execute script	x=\${i}; niz=\${godine}; return niz[x];	godina
20	execute script	x=\${i}; niz=\${odgovori}; return niz[x];	odgovor
21	type	id=godiste	\${godina}
22	click	id=Odgovor	
19	execute script	x=\${i}; niz=\${godine}; return niz[x];	godina
20	execute script	x=\${i}; niz=\${odgovori}; return niz[x];	odgovor
21	type	id=godiste	\${godina}
22	click	id=Odgovor	
23	type	id=Odgovor	\${odgovor}
24	execute script	return document.getElementById("time-error") == null	notPresentTime
25	execute script	return document.getElementById("godiste-error") == null	notPresentGodiste
26	execute script	return document.getElementById("Email-error") == null	notPresentEmail
27	execute script	return document.getElementById("BrojTelefona-error") == null	notPresentBroj
28	execute script	return document.getElementById("Odgovor-error") == null	notPresentOdgovor
29	if	\$(notPresentTime) == true && \$(notPresentGodiste) == true && \$(notPresentEmail) == true && \$(notPresentBroj) == true && \$(notPresentOdgovor) == true	
30	execute script	return parseInt(\$(brojac)+1)	brojac
31	end		
32	click	css=.row	
33	execute script	return parseInt(\$(i)+1)	i
34	end		
35	if	\$(brojac) == 6	
36	execute script	alert("Validni svi unosi!")	
37	end		
38	close		

Opis testnog slučaja za sve nevalidne podatke (scenarij po kojem će se izvršiti testiranje):

Verifikacija i Validacija Softvera

Testovi su koncipirani tako da svako polje za koje je to moguće dobije po jednu nevalidnu vrijednost, i da se provjeri da li se ispod svakog tog polja nalazi poruka greške, te ako ako se nalazi ispisuje se poruka da su svi unosi nevalidni. Na početku se prave četiri niza. Prvi niz sadrži po jednu nevalidnu vrijednost za svaku varijablu, a za svaku tu vrijednost u drugom nizu se nalazi id u koji se ona smiješta, u trećem id polja u kojem bi trebala biti greška za tu vrijednost, i u četvrtom poruka greške u polju sa id-em iz trećeg niza. Zatim se prolazi kroz petlju i u svakoj iteraciji uzme po jedna vrijednost iz sva četiri niza, te ako je greška zaista ispisana povećava se brojač. Ako brojač odgovara veličinama spomenutih nizova test je uspješan.

Selenium skripta za test za sve nevalidne podatke (prikaz nakon pokretanja, odakle se vidi da skripta uspješno prolazi):

1	open	http://localhost:50942/Grupa3	
2	set window size	1074x757	
3	execute script	return ["2022", "13.", "John", "jdepp@", "-331"]	nevalidnaVrijednost
4	execute script	return ["godiste", "Odgovor", "Ime", "Email", "BrojTelefona"]	ids
5	execute script	return ["godiste-error", "Odgovor-error", "Ime-error", "Email-error", "BrojTelefona-error"]	errori
6	execute script	return ["Niste ispravno unijeli godiste!", "Morate unijeti barem 5 karaktera!", "Morate unijeti barem 5 karaktera!", "E mail adresa nije ispravna!", "Morate unijeti barem 5 karaktera!"]	errorPoruke
7	click	linkText=Grupa 3	
8	click	linkText=Dodaj novi odgovor	
9	store	0	brojac
10	store	0	i
11	while	parseInt(\$(!))<5	
12	click	id=godiste	
13	execute script	x=\${!}; niz=\${nevalidnaVrijednost!}; return niz[x];	nevalidnaVrijednost
14	execute script	x=\${!}; niz=\${ids}; return niz[x];	id
15	click	id=\${id}	
16	type	id=\${id}	\$(nevalidnaVrijednost)
17	click	css=row	
18	execute script	x=\${!}; niz=\${errori}; return niz[x];	error
19	store text	id=\${error}	error
20	execute script	x=\${!}; niz=\${errorPoruke}; return niz[x];	errorPoruka
21	if	\$(errorPoruka) == \$(error)	
22	execute script	return parseInt(\$(brojac)+1)	brojac
23	end		
24	execute script	return parseInt(\$(i)+1)	i
25	end		
26	if	\$(brojac) == 5	
27	execute script	alert("Nevalidni svi unosi!")	
28	end		
29	close		

Koji dio testova je snimljen, a koji manuelno skriptovan?

U testu za nevalidne podatke sav kod osim ulaska na stranicu je manuelno skriptovan. U testu za validne snimljen čitav početni dio do ulaska u petlju osim deklarisanja nizova, a sve ostalo je urađeno manuelno,

Zadatak 3. (Testiranje performansi)

Potrebno je izvršiti load testiranje nasumično odabrane stranice koristeći alat Loadster. Svi članovi tima trebaju imati različite postavke za broj i lokaciju virtuelnih korisnika pri vršenju testiranja.

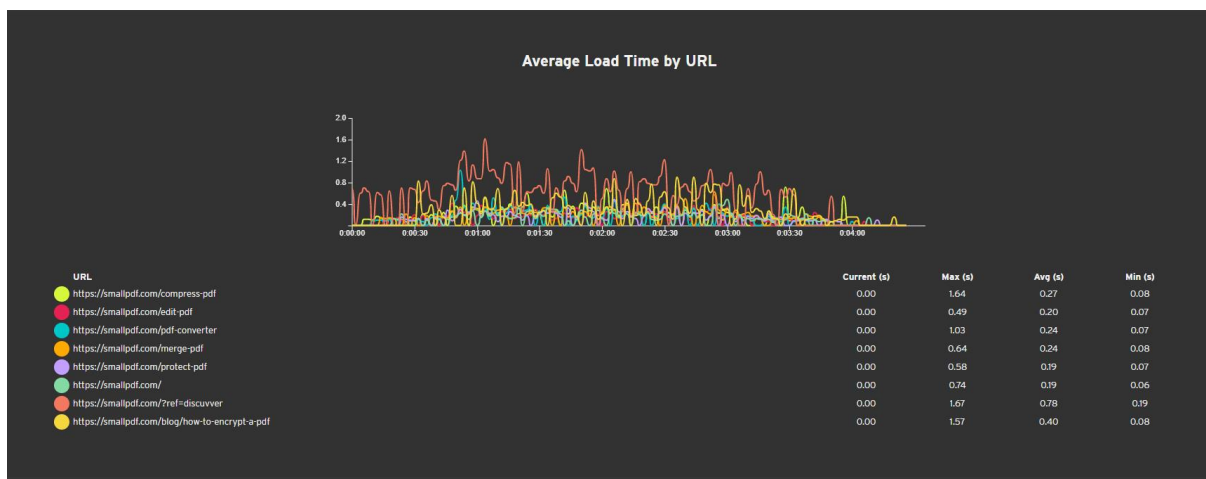
Nasumično odabrana web-stranica: <https://smallpdf.com/?ref=discuver>

Load testiranje – član tima: Amina Šiljak

Prikaz postavki za virtuelne korisnike prije početka testiranja:



Prikaz srednjeg vremena pristupa stranici:



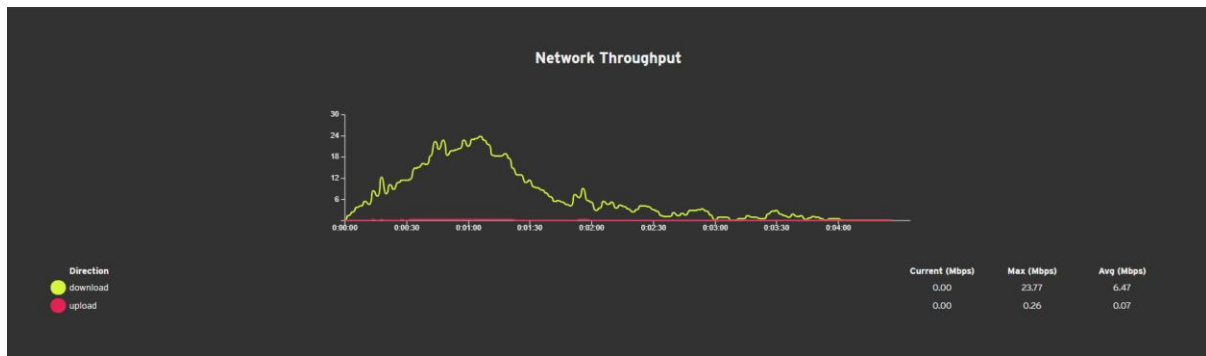
Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se vrijeme čekanja povećava sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Vrijeme čekanja se jeste povećalo sa povećanjem broja virtuelnih korisnika, te vidimo na grafu da se prije prve minute vrijeme pomalo povećava, dok se poslije prve minute pojavljuju periodi

Verifikacija i Validacija Softvera

sa i po duplo većim vremenom učitavanja od onog prije prve minute na većini url-ova. Poslije treće minute, kada stranica prestane biti preopterećena, ovo vrijeme se dosta smanji. Load time je najveći na početnoj stranici i on dostigne najveću vrijednost 1.67s, dok prije ulaska svih korisnika on ne dostigne ni 0.8s, što znači da vrijeme učitavanja i performanse ove web-aplikacije uveliko zavise od broja aktivnih korisnika.

Prikaz propusnosti mreže stranice:

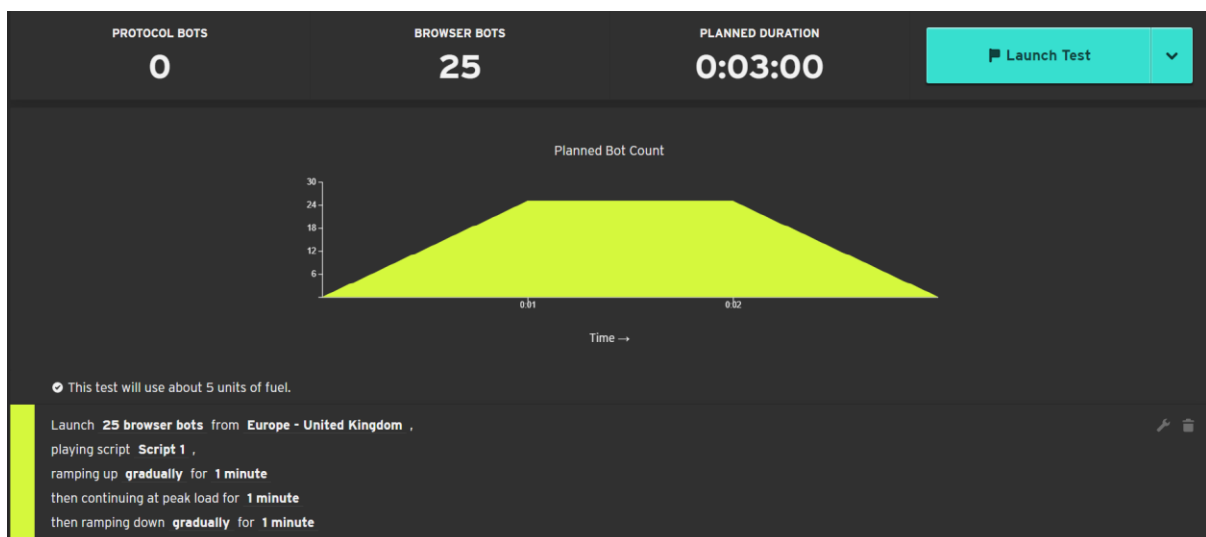


Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se propusnost mreže smanjuje sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Ovaj graf baš i nema tipični izgled. Propusnost se postepeno povećava kako se povećava broj korisnika, u prvoj minuti doživi vrhunac i onda počinje postepeno opadati, te nigdje ne drži konstantnu vrijednost.

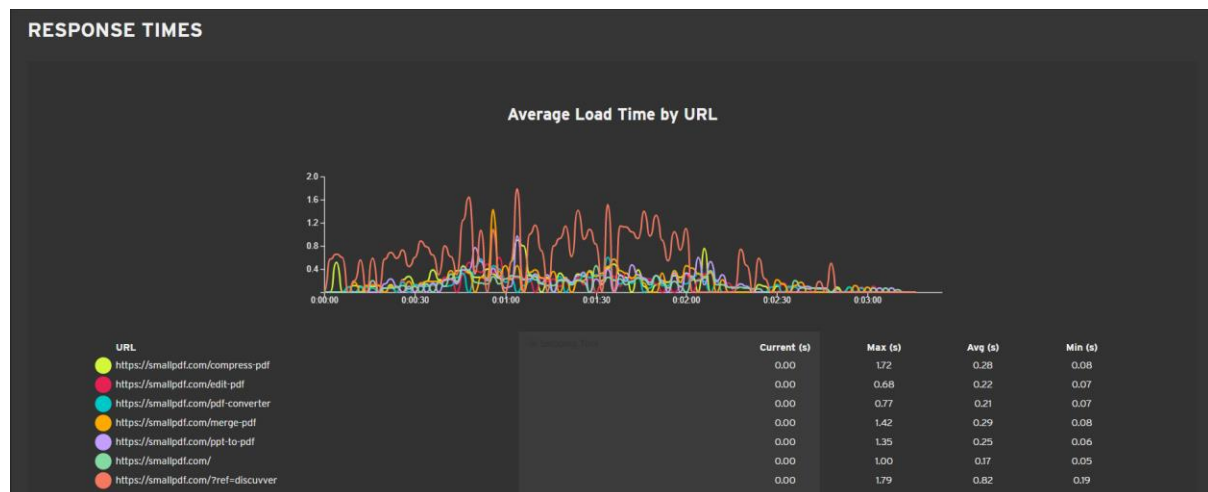
Load testiranje – član tima: Berina Suljić

Prikaz postavki za virtuelne korisnike prije početka testiranja:



Prikaz srednjeg vremena pristupa stranici:

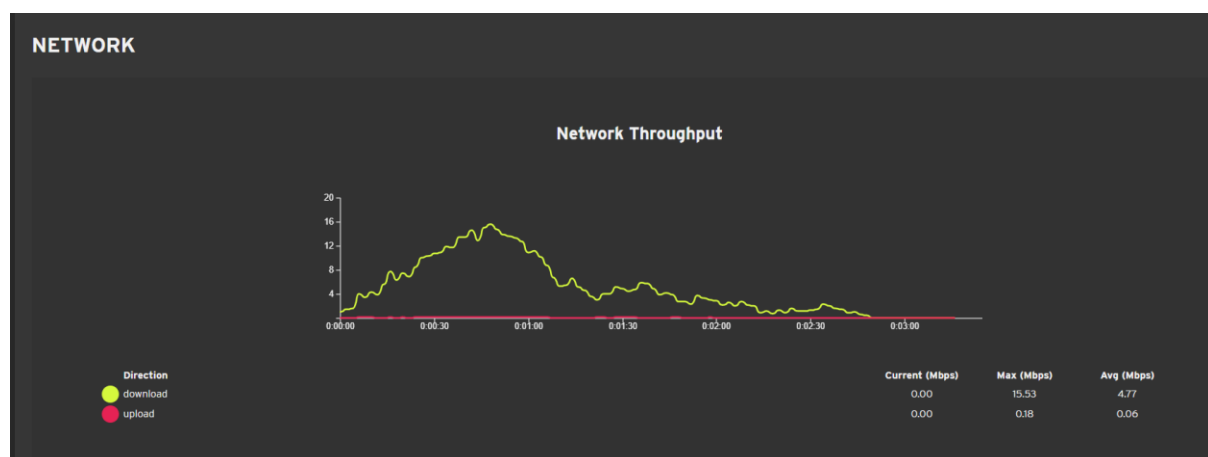
Verifikacija i Validacija Softvera



Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se vrijeme čekanja povećava sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Sa povećanjem broja virtuelnih korisnika, povećava se i vrijeme čekanja. Sa grafa prikazanog na slici iznad zaključujemo da se prije prve minute vrijeme pomalo povećava (kako se povećava i broj korisnika), a poslije prve minute se drastično povećava vrijeme u odnosu na ranije. Poslije treće minute, kada stranica prestane biti preopterećena, ovo vrijeme se dosta smanjuje. Load time je najveći na <https://smallpdf.com/?ref=discuvver> i on dostigne najveću vrijednost 1.79s, dok prije ulaska svih korisnika on dostigne 0.82s, što znači da vrijeme učitavanja i performanse ove web-aplikacije uveliko zavise od broja aktivnih korisnika.

Prikaz propusnosti mreže stranice:



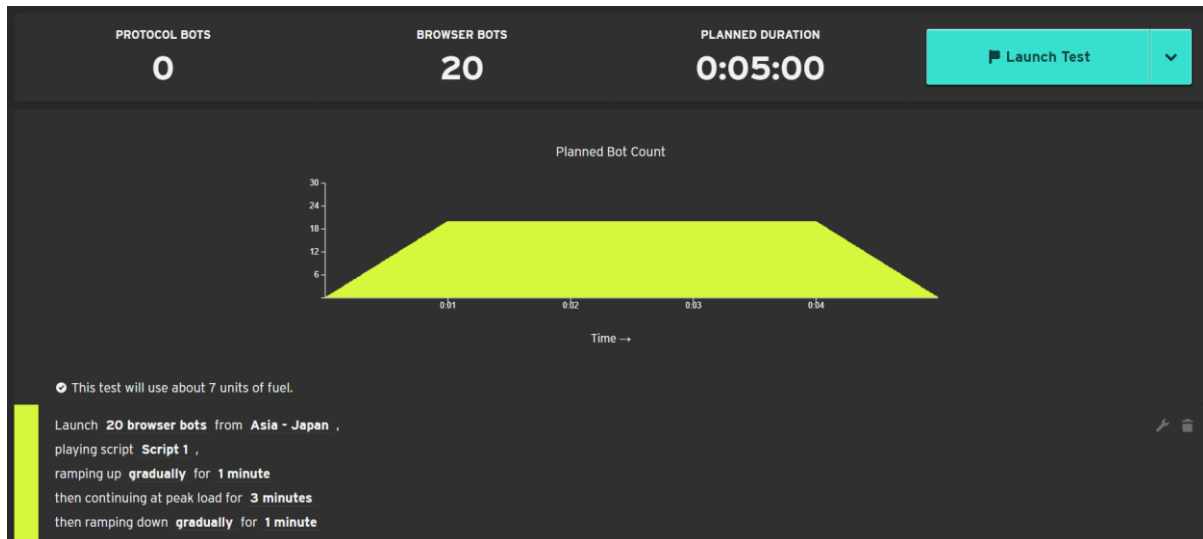
Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se propusnost mreže smanjuje sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Propusnost mreže se postepeno povećava kako se povećava broj korisnika, u prvoj minuti doživi vrhunac i onda počinje postepeno opadati, te nigdje ne drži konstantnu vrijednost. I iz ovoga vidimo da broj korisnika u mnogome doprinosi performansama web-aplikacije.

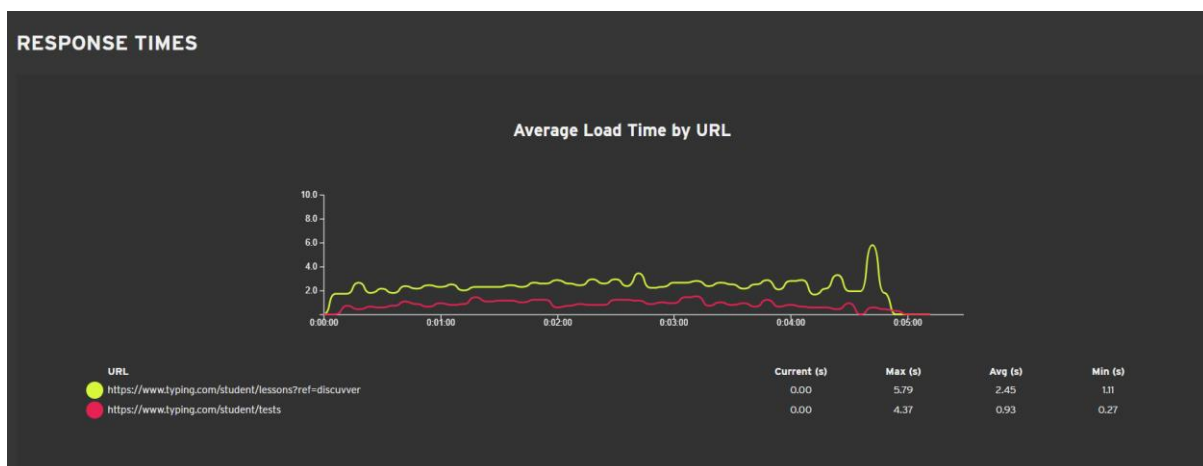
Load testiranje – član tima: Elma Šeremet

Nasumično odabrana web-stranica: <https://www.typing.com/student/lessons?ref=discuvver>

Prikaz postavki za virtuelne korisnike prije početka testiranja:



Prikaz srednjeg vremena pristupa stranici:



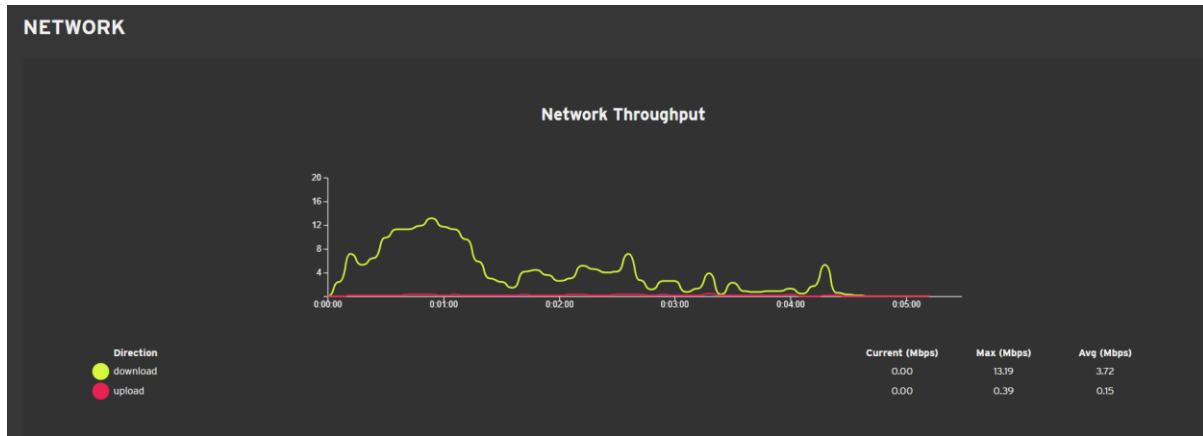
Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se vrijeme čekanja povećava sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Najvažniji i svakako najčešće korišteni graf load testiranja je graf srednjeg vremena učitavanja. Na grafu na slici iznad može se vidjeti kako se mijenjala količina vremena pristupa stranici kroz vrijeme i kroz povećanje broja virtuelnih korisnika. Vrijeme pristupa je u određenim trenucima dostizalo i 6 sekundi, a srednje vrijeme pristupa je 2.45 sekundi, što znači da je svaki virtuelni korisnik morao čekati u prosjeku 2.45 sekundi da bi mu se učitala webstranica. Iako je vrijeme pristupa veoma dugo, ovaj graf zapravo pokazuje da to vrijeme nije linearno raslo

Verifikacija i Validacija Softvera

sa rastom broja korisnika, odnosno da je aplikacija u stanju da uglavnom otprilike istom brzinom opslužuje zahtjeve bez obzira na broj korisnika.

Prikaz propusnosti mreže stranice:



Šta se može zaključiti sa grafa prikazanog na slici iznad? Da li se propusnost mreže smanjuje sa povećanjem broja virtuelnih korisnika? Šta to znači za performanse web-aplikacije?

Propusnost mreže se uglavnom povećava kako se povećava broj korisnika, te također u prvoj minuti dostigne vrhunac i onda počinje opadati. Također, nigdje ne drži pretjerano konstantnu vrijednost i vidimo da broj korisnika ipak doprinosi performansama web-aplikacije.

Sada je potrebno izvršiti testiranje sadržaja nasumično odabranih stranica koristeći alat Pingdom. Svi članovi tima trebaju testirati drugu nasumično odabranu stranicu.

Testiranje sadržaja – član tima: Amina Šiljak

Nasumično odabrana web-stranica: <https://asoftmurmur.com/?ref=discuvver>

Prikaz osnovnih metrika:

Your Results:

 DOWNLOAD HAR

 SHARE RESULT



Performance grade

A 96

Page size

1.0 MB

Load time

1.22 s

Requests

15

Da li su osnovne metrike zadovoljavajuće? Izvršiti njihovu analizu.






Ukupna ocjena stranice je odlična. Load time je prešao 1s što je više nego što bi trebalo. Veličina stranice je dovoljno mala kao i broj poslanih zahtjeva. Sve u svemu, stranica ima zadovoljavajuće osnovne metrike.

Detaljni prikaz informacija o sadržaju stranice:






Response codes

RESPONSE CODE	RESPONSES
200 OK	11
206 Partial Content	4

Content size by content type

CONTENT TYPE	PERCENT	SIZE
 Image	56.60%	232.8 KB
 HTML	31.49%	129.5 KB
 Font	6.60%	27.2 KB
 Script	4.83%	19.9 KB
 XHR	0.48%	2.0 KB
Total	100.00%	411.4 KB

Requests by content type

CONTENT TYPE	PERCENT	REQUESTS
 Image	44.44%	4
 XHR	22.22%	2
 Script	11.11%	1
 HTML	11.11%	1
 Font	11.11%	1
Total	100.00%	9

Koji sadržaj čini najveći dio stranice? Izvršiti analizu prednosti i mana web-stranice na osnovu njenog sadržaja.

Najveći dio stranice čine slike, što nije idealan slučaj, jer bi na prvom mjestu trebale biti js skripte, a one na ovoj stranici čine tek 4.83%. Međutim, ovo je razumljivo s obzirom na to da stranici ne treba velika programska logika, i što slike zauzimaju jako malu količinu memorije. Na drugom mjestu je html kod što je odlično, a css-a uopće i nema pa on neće povećavati vrijeme load-anja stranice.

Testiranje sadržaja – član tima: Berina Suljić

Nasumično odabrana web-stranica: <https://www.memrise.com/?ref=discuvver>

Prikaz osnovnih metrika:

Your Results:



DOWNLOAD HAR



SHARE RESULT



Performance grade

D 66

Page size

1.6 MB

Load time

1.25 s

Requests

88

Da li su osnovne metrike zadovoljavajuće? Izvršiti njihovu analizu.

Ukupna ocjena stranice na osnovu parametara koji se odnose na brzinu odziva, vrijednosti HTTP odgovora, validnost sadržaja stranice i mrežne karakteristike jeste 66 (D), što je loša ocjena. Srednja vrijednost količine podataka potrebnih za učitavanje stranice je uredi. Srednja vrijednost odziva na osnovu poslanih zahtjeva jeste 1,25s, što je više od 1s i više nego što bi trebalo biti. Broj poslanih zahtjeva na osnovu kojih su generisane vrijednosti ostalih metrika iznosi 88, što je srednje zadovoljavajuće. Osnovne metrike bi trebale biti bolje.

Detaljni prikaz informacija o sadržaju stranice:

Response codes

RESPONSE CODE	RESPONSES
200 OK	85
204 No Content	2
301 Moved Permanently	1

Content size by content type

CONTENT TYPE	PERCENT	SIZE
Image	40.97%	651.3 KB
Script	39.91%	634.5 KB
Font	14.54%	231.1 KB
CSS	1.94%	30.9 KB
HTML	1.88%	29.9 KB
XHR	0.74%	11.8 KB
Redirect	0.02%	260.0 B
Total	100.00%	1.6 MB

Requests by content type

CONTENT TYPE	PERCENT	REQUESTS
Image	44.83%	39
Script	31.03%	27
Font	6.90%	6
XHR	5.75%	5
CSS	5.75%	5
HTML	4.60%	4
Redirect	1.15%	1
Total	100.00%	87

Verifikacija i Validacija Softvera

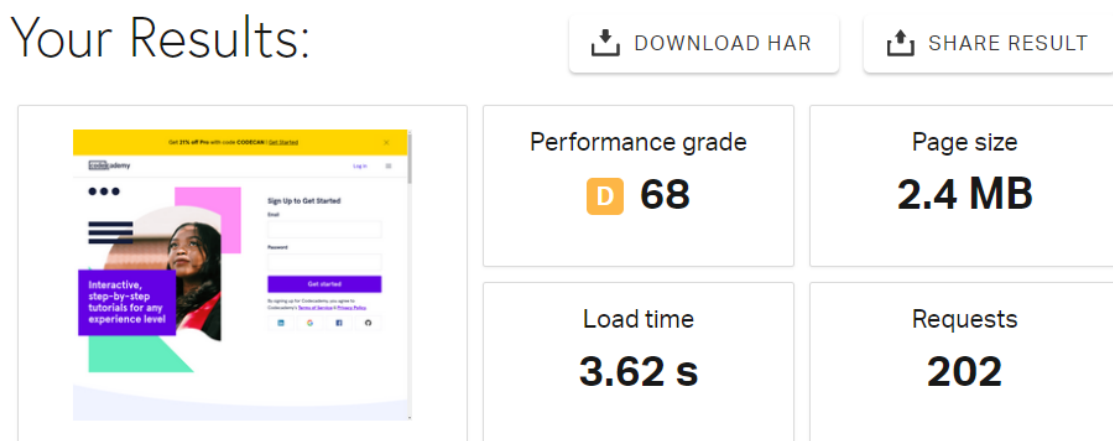
Koji sadržaj čini najveći dio stranice? Izvršiti analizu prednosti i mana web-stranice na osnovu njenog sadržaja.

Najveći dio stranice (40.97%) čine slike, dok poslije njih sa 39.91% čine skripte. Obzirom da najveći dio stranice čine slike, to nije baš dobro, jer one zauzimaju i najviše. HTML fajlovi su jako malo zastupljeni, što je mana ove stranice. Ali, programerska logika kroz skripte je jako zastupljena, pa je to jedna od prednosti.

Testiranje sadržaja – član tima: Elma Šeremet

Nasumično odabrana web-stranica: <https://www.codecademy.com/?ref=discuvver>

Prikaz osnovnih metrika:



Da li su osnovne metrike zadovoljavajuće? Izvršiti njihovu analizu.

Ukupna ocjena stranice na osnovu parametara koji se odnose na brzinu odziva, vrijednosti HTTP odgovora, validnost sadržaja stranice i mrežne karakteristike jeste 68 (D), što je loša ocjena. Srednja vrijednost količine podataka potrebnih za učitavanje stranice je uredno. Srednja vrijednost odziva na osnovu poslanih zahtjeva jeste 3.62s, što je jako dugo i nezadovoljavajuće. Broj poslanih zahtjeva na osnovu kojih su generisane vrijednosti ostalih metrika iznosi 202, što je znatno više nego na nekim drugim stranicama iz naprimjer tutorijala, ali opet ako uzmemo u obzir da ih zna biti hiljade na jednoj stranici i ovo je relativno malo. Osnovne metrike bi trebale biti bolje.

Detaljni prikaz informacija o sadržaju stranice:

Verifikacija i Validacija Softvera

Response codes

RESPONSE CODE	RESPONSES
200 OK	197
204 No Content	4
301 Moved Permanently	1

Content size by content type

CONTENT TYPE	PERCENT	SIZE
Script	73.76%	1.7 MB
Image	9.47%	212.4 KB
Font	6.48%	145.5 KB
HTML	5.29%	118.8 KB
CSS	3.97%	89.1 KB
XHR	1.00%	22.4 KB
Redirect	0.03%	573.0 B
Total	100.00%	2.2 MB

Requests by content type

CONTENT TYPE	PERCENT	REQUESTS
Script	48.40%	91
Image	23.94%	45
CSS	15.43%	29
XHR	6.38%	12
Font	3.72%	7
HTML	1.60%	3
Redirect	0.53%	1
Total	100.00%	188

Koji sadržaj čini najveći dio stranice? Izvršiti analizu prednosti i mana web-stranice na osnovu njenog sadržaja.

Najveći dio stranice čine skripte sa čak 73.6%, dok iza njih slijede slike sa 9.47%. Iz izlaza iznad može se doći do zaključka da se stranica većinski sastoji od programske logike zbog postotka zastupljenosti js skripti, što je sasvim uredu. Što se tiče slika, one ne zauzimaju puno, tako da je i to sasvim uredu.

Zadatak 4. (Istraživanje)

Članovi tima koji učestvuju u istraživanju:

Ime i prezime: [Click here to enter text.](#)

Ime i prezime: [Click here to enter text.](#)

Ime i prezime: [Click here to enter text.](#)

Ovaj dio izvještaja potrebno je manuelno popuniti, ovisno o odabranim alatima za evidenciju grešaka. Za testiranje i demonstriranje rada alata potrebno je koristiti ekvivalentne klase formirane u trećem dijelu prvog zadatka. Obavezno naznačiti koji član tima je učestvovao u kojem dijelu istraživanja.