

## Domaći zadatak 07 – Branislav Manojlović kp11

### KI301-DZ07

#### Tekst zadatka

Odabrati aplikaciju u Java programskom jeziku i izvršiti sledeće:

1. Opisati proces integracionog testiranja aplikacije,
2. Napraviti primer integracionog testiranja,
3. Izvršiti testiranje primenom odabranog test alata koristeći top-down tehniku.

#### Rešenje:

- 1) Proces integracionog testiranja.

**Interacija softvera** je integracija softverskih komponenti/jedinica u jednu funkcionalnu celinu. **Integraciono testiranje** uključuje pisanje malih delova koda (snippets) za potrebe testiranja integrisanih modula. Glavni **cilj** Integracionog testiranja je da se testiraju interfejsi između jedinica/modula. **Ideja** Integracionog testiranja je da prvo treba integrisati jedinično istestirane module jedan po jedan u složenije celine, a zatim treba testirati svaki ovaj složeni modul kao jednu jedinicu. Integraciono testiranje se može obaviti i white-box i black-box tehnikom.

Postoje dva **osnovna pristupa** u Integracionom testiranju:

- Odozdo-nagore (Bottom up approach),
- Odozgo-nadole (Top down approach).

**Koraci** koji su potrebni da se izvrše u Integracionom testiranju:

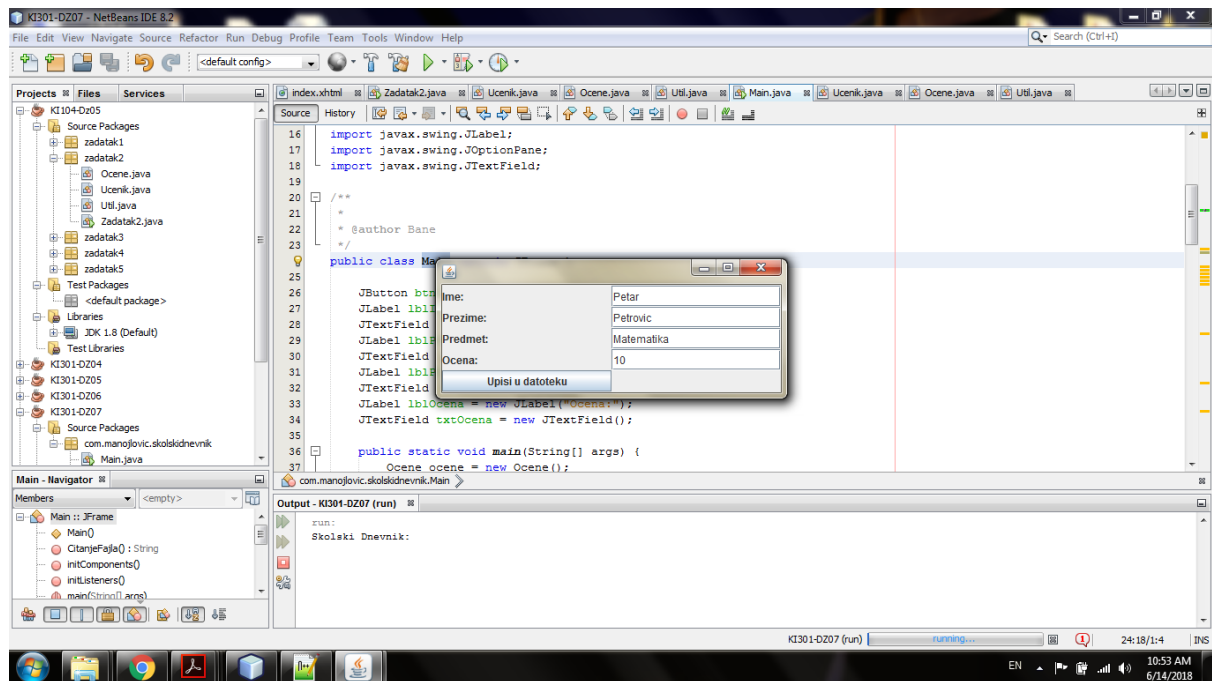
1. Proučiti arhitekturu aplikacije,
2. Identifikovati module,
3. Identifikovati šta svaki modul radi,
4. Identifikovati kako se podaci prenose, tj. razmenjuju preko pojedinih modula,
5. Identifikovati koji su ulazi i koji su izlazi iz cele aplikacije (tj. koji se podaci unose u sistem a koji se podaci primaju od sistema),
6. Identifikovati i napraviti test slučajeve.

- 2) Primer je aplikacija Školski dnevnik.

Izvršićemo Integraciono testiranje konzolne aplikacije Školski dnevnik. U ovom programu korisnik (učenik) bi trebalo da svoje ocene iz predmeta upisuje u fajl preko grafičkog interfejsa. Nakon unosa učenik bi trebalo da može na konzoli da pogleda unešene ocene kao i njihov krajnji prosek. Aplikacija je dostavljena u fajlu: **KI301-DZ07.rar**.

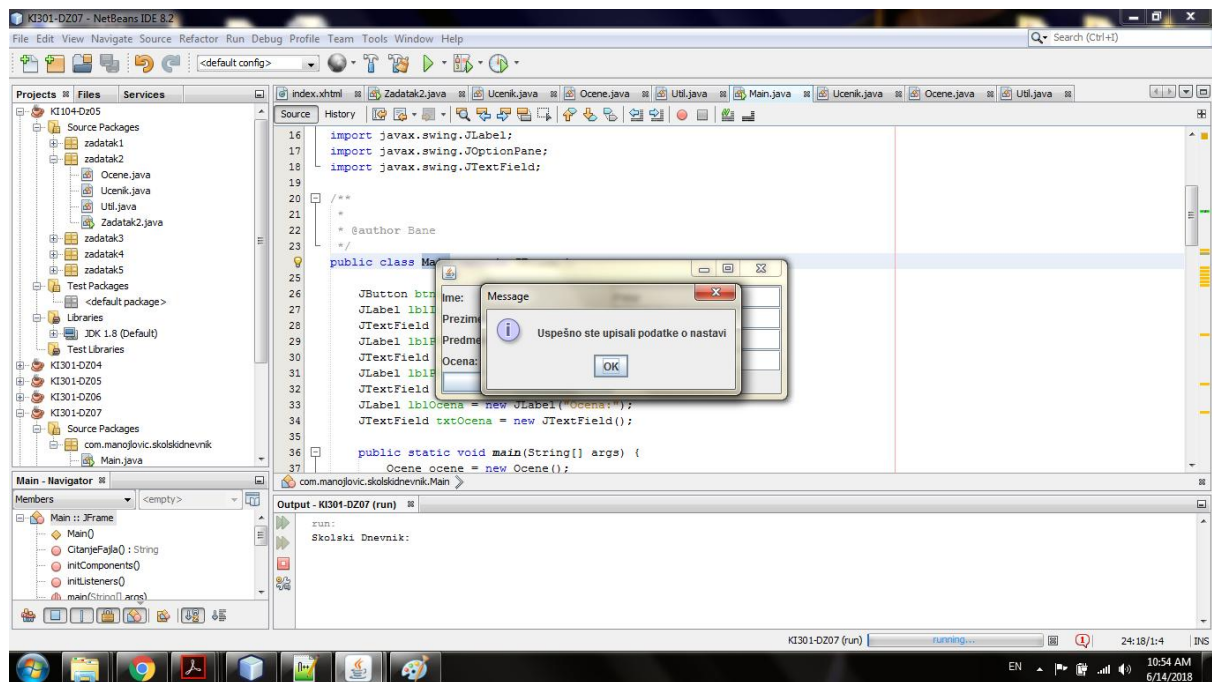
Aplikacija je realizovana klasama: Main.java, Util.java, Ucenik.java, Ocene.java.

### Pokretanje aplikacije:



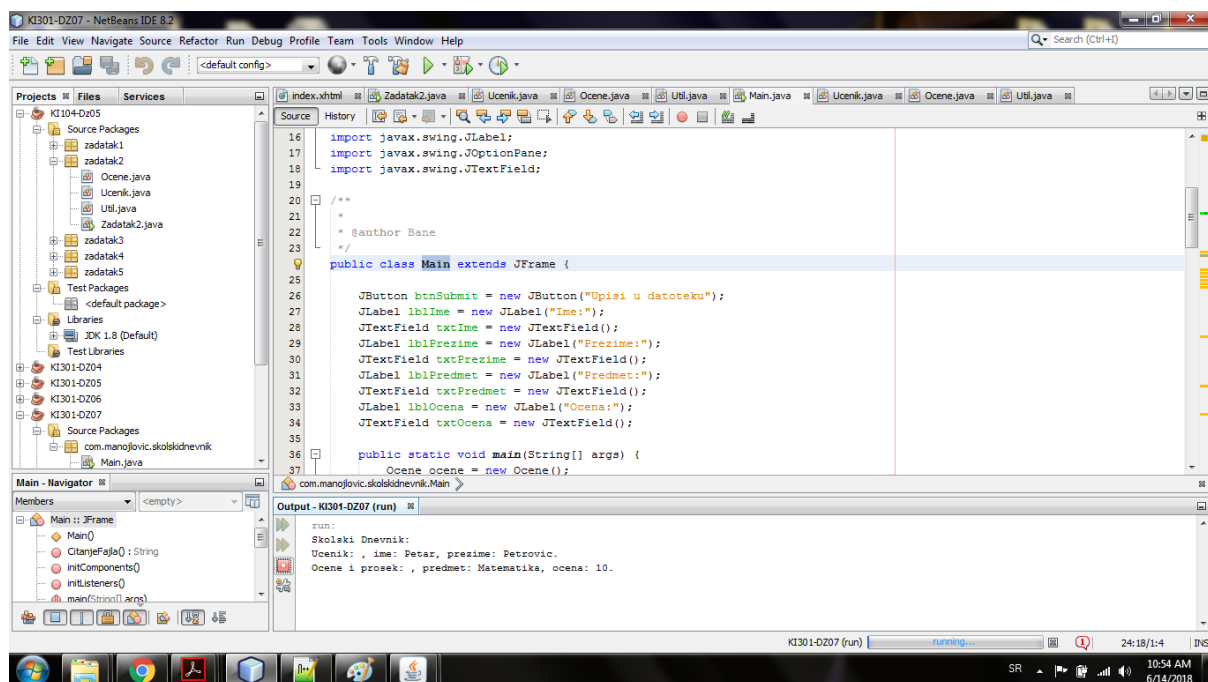
### Slika 1. Unos podataka

Potvrda o uspesnom unosu podataka:



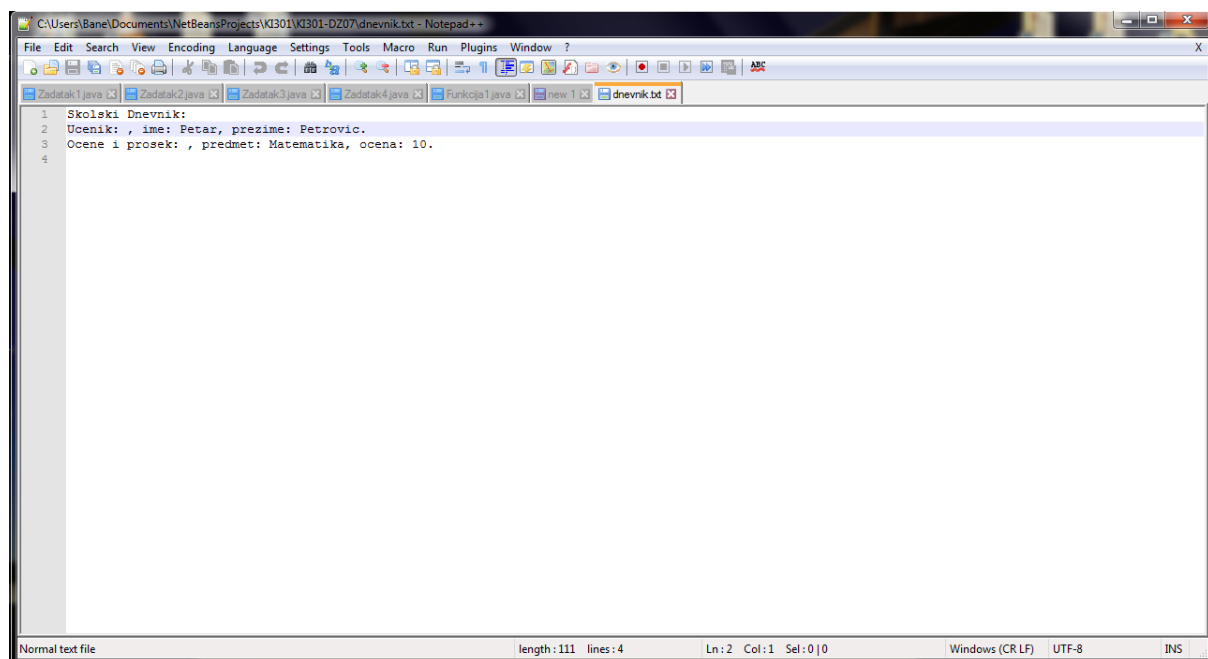
Slika 2. Podaci su uspesno uneti

Ispis sadržaja iz fajla dnevnik.txt:



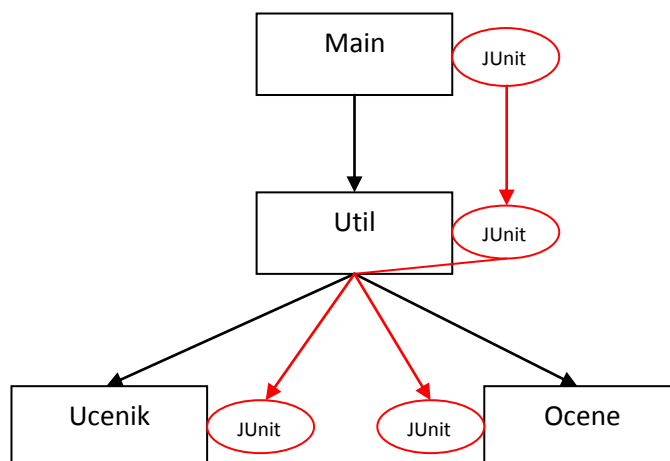
Slika 3. Ispis sadržaja fajla dnevnik.txt na konzoli

Prikaz sadržaja fajla:



Slika 4. Sadržaj fajla dnevnik.txt

3) Primenjen pristup je Intergaciono testiranje **odozgo-nadole** (top-down).



Slika 5. Dijagram Integracionong testiranja Top-down pristupom

4) Testiranje ćemo izvršiti primenom JUnit framework-a u NetBeans random okruženju, budući da je kompletna aplikacija pisana u prog. jez. Java.

#### **Redosled i postupak testiranja:**

1. Prvo se testira Main.java fajl i metode koje se u njemu nalaze: **CitanjeFajla()**, **initListeners()**. Pri tome se kao predefinisane vrednosti hardkoduju podaci koje će metoda CitanjeFajla() prikazati na konzoli, zato što trenutno modul-klasa Util.java nije implementirana, pa je potrebno simulirati njen rad pomoću Skela (Stub). **Skele** su komadi programa koji prihvataju ulaze/zahteve od gornjeg modula i vraćaju rezultat, tj. odgovor, koriste se u Top-down tehnikama Integracionog testiranja.
2. U drugom koraku se testira Util.java fajl i metoda **upisiUFajl()** koja u njemu postoji. U ovom trenutku još uvek nisu implementirane klase-moduli Ucenik.java i Ocene.java, pa se za njihovo simuliranje koriste Skele sa hardkodovanim podacima.
3. U trećem koraku se integrišu Main.java i Util.java klase jer su one kreirane i jedinično istestirane. Nakon što je jedinično testiranje ovih dvaju klasa urađeno i nakon što je izvršena njihova intergacija, vrši se njihovo testiranje kao funkcionalne celine, pri čemu se i dalje umesto klasa-modula Ucenik.java i Ocena.java koriste Skele. Nakon što je ovo testiranje urađeno, prelazi se na sledeći korak.
4. Sada se vrši jedinično testiranje klasa Ucenik.java, koja je su sada konačno završena i može da se testira kao funkcionalna celina. U njoj se testira metoda: **toString()**.
5. Poslednja klasa koja se jedinično testira je Ocena.java i u njoj se isto testira metoda: **toString()**.
6. Nakon što su klase Ucenik.java i Ocena.java istestirane kao posebne celine, one se sada integrišu sa klasama Main.java i Util.java i testiraju kao jedan sistem.

**MSc Branislav Manojlović**  
**Novi Sad 13.06.2018.**