

Zadatak nedelja 5

Napomena: Ovaj zadatak je poslednji za prvu kontrolnu tačku (KT1). Kod za KT1 treba da uključi zadatke iz svih prethodnih nedelja.

Rok za predaju: 11.11. u 23:59h.

Pitanja: Studenti mogu postavljati tehnička pitanja vezana za zadatak do 08.11. (uključujući i taj dan).

Poeni: KT1 nosi maksimalno 18 poena.

1. Implementacija stabla uz JTree

U levom panelu aplikacije, potrebno je implementirati konkretno stablo koje se oslanja na Composite šablon rađen prošle nedelje, koristeći JTree vizuelnu komponentu. Stablo treba da omogući sledeće funkcionalnosti:

- **Dodavanje čvorova** (npr. dodavanje novih stambenih objekata ili soba),
- **Izmenu imena čvorova**,
- **Brisanje čvorova**.

Sve akcije treba omogućiti sa dva mesta:

- **Toolbar** (alatna traka),
- **Glavni meni** (meni na vrhu aplikacije).

Napomene:

- Imenovanje čvorova može se realizovati pomoću **TreeCellEditor-a** i/ili kroz odgovarajuće akcije u meniju ili toolbaru.
- Iz menija i toolbar-a treba omogućiti promenu ili postavljanje **autora, naziva i putanje projekta**.
- Stablo implementirajte kao komponentu (interfejs sa metodama + klasa koja ga implementira) pod nazivom `DraftTree` (`DraftTreeImplementation`). Akcije poput **kreiranja čvorova** (`createNode`), **dodavanja dece** (`addChild`), i **generisanja stabla** (`generateTree`) realizovati preko interfejsa dok je njihove implementacije potrebno uraditi u samoj klasi (`DraftTreeImplementation`)

2. Korisnički interfejs sa JTabbedPane komponentom

Proučite kako funkcioniše **JTabbedPane** (dokumentacija i primeri su dati ispod*). Na osnovu selekcije (dupli klik) na čvor projekta (**Project**) u stablu, potrebno je otvoriti tabove za svaku **sobu** (Room) u tom podstablu konkretnog projekta bilo da se ona nalazi kao dete u nekom objektu ispod (Building) ili direktno kao njegovo dete.

- **Tabovi** sadrže panele za uređivanje soba na kojima ćemo kasnije vršiti crtanje.

- **Tabovi za sobe** koje su u istom objektu (Building) treba da budu obojeni istom bojom, dok ukoliko se nalaze u različitim objektima treba da budu različite boje (boju možete random praviti i čuvati njenu rgb vrednost u modelu objekta (Building) pa odatle farbati sve tabove koji su njegova deca)
- Na desnom panelu aplikacije, dodajte **labelu** u kojoj će biti prikazan **naziv trenutno otvorenog projekta kao i objekta u kome se soba nalazi** (staviti '/' ako nije u objektu soba).
- Ispod naziva projekta, prikažite i ime autora. Ovo omogućava prikaz rada **Observer** šablona prilikom promena.

3. Povezivanje DraftRepository-a sa desnim panelom (Observer šablon)

Desni deo aplikacije treba da koristi **Observer** šablon kako bi reagovao na promene u stablu:

- **Ceo desni panel** (ProjectView) treba da osluškuje promene na projektu i stambenim objektima u njemu (zbog labela koje tamo stoje).
- **Pojedinačni tab** (RoomView) treba da osluškuje promene na pojedinačnoj sobi i/ili nekom njenom parentu (moguće je više rešenja).

Kada dođe do promene (npr. promena naziva elementa, autora ili brisanje/dodavanje čvorova u stablu), **Observer** treba odmah da ažurira prikaz na desnom panelu i tabovima (ako se obriše soba, njen tab se automatski zatvara, ako se doda nova onda se pojavi i novi tab i slično).

4. Korišćenje FactoryMethod šablona za kreiranje čvorova

Implementirajte kreiranje čvorova u stablu koristeći **FactoryMethod** šablon. U klasi `DraftTreeImplementation`, prilikom kreiranja novih čvorova, pozivajte odgovarajuću **fabriku** iz `DraftRepository-a`.

`Factory` može biti podkomponenta `DraftRepository` ili neke druge klase modela po vašem izboru.

5. Obrađivanje grešaka korisnika u projektu

Potrebno je obraditi sve greške koje korisnik može napraviti u projektu preko `MessageGenerator` komponente koja je implementirana u prošlom zadatku (Korisnik pokuša da izbriše Project Explorer i slično...) Takođe, potrebno je odraditi proveru prilikom kreiranja novih elemenata ili preimenovanja starih tako da ne postoje dva čvora u stablu na istoj putanji koji se isto zovu (isto kao File System na bilo kom operativnom sistemu)

NAPOMENA

OBAVEZNO JE POŠTOVANJE MVC ARHITEKTURE TOKOM CELOG PROJEKTA! GUBIĆE SE POENI U SLUČAJU NJENOG KRŠENJA.

Dokumentacija i primeri za JTabbedPane:

<https://docs.oracle.com/javase/tutorial/uiswing/components/tabbedpane.html>

<https://www.onlinetutorialspoint.com/java/java-swing-jtabbedpane-example.html>