

## 8. Osmo laboratorijska vježba

### 8.1. Dodatne funkcionalnosti s JavaFX-om

Svrha laboratorijske vježbe je proširivanje JavaFX aplikacije iz sedme laboratorijske vježbe s dodatnim mogućnostima proširivanja liste komponenata unošenjem karakteristika komponenti u zasebnom dijalogu. Demonstriranje funkcionalnosti aplikacije može se vidjeti na sljedećem Youtube linku: <https://www.youtube.com/watch?v=CUY9zF5tExk>.

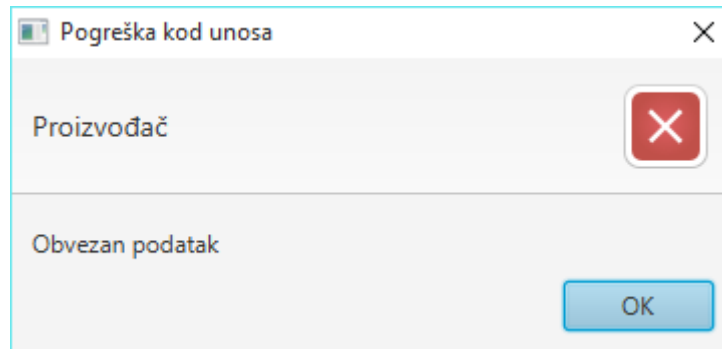
### 8.2. Zadatak za pripremu

Nastaviti razvoj aplikacije iz sedme laboratorijske vježbe, oblikovati dodatne ekrane JavaFX grafičkog sučelja aplikacije i funkcionalnosti spremanja novih podataka u datoteke prema koracima u nastavku.

1. Kopirati projekt iz sedme laboratorijske vježbe i preimenovati ga na način da mu se promijeni redni broj u „8“.
2. U aplikaciju dodati četiri izbornika: „Matične ploče“, „Procesori“, „Radne memorije“ i „Tvrdi diskovi“. U svaku od njih dodati opciju „Dodaj novu/novi“. Element koji predstavlja izbornik potrebno je postaviti u „Top“ područje „BorderLayout“ organizatora rasporeda komponenti, a poruku „Odaberite komponente koje želite dodati u konfiguraciju računala“ prebaciti u naslov prozora.
3. Za svaku opciju izbornika kreirati novi prozor koji će omogućavati unos nove komponente. Prozor je potrebno dizajnirati pomoću Scene Buildera, a FXML datoteku potrebno je učitati i prikazati na odabir opcije unutar izbornika. Na primjer, za prikaz ekrana za unos nove matične ploče potrebno je napisati odgovarajuću metodu unutar klase „PocetniEkranController“ kao što je „prikaziEkranZaNovuMaticnuPlocu“ i povezati je s akcijom izbornika „Unos nove“. Primjer te metode je u nastavku:

```
public void prikaziEkranZaNovuMaticnuPlocu() throws IOException {  
    FXMLLoader loader = new  
    FXMLLoader(getClass().getResource("NovaMaticnaPloca.fxml"));  
    Scene scene = new Scene((Parent) loader.load());  
    Stage stage = new Stage();  
    stage.setTitle("Unos nove matične ploče");  
    stage.setScene(scene);  
    stage.show();  
}
```

4. Za svaki ekran koji služi za unos podataka o matičnoj ploči, procesoru, radnoj memoriji i tvrdom disku kreirati odgovarajuću „controller“ klasu koja će sadržavati sve elemente grafičkog sučelja pridružene „@FXML“ anotaciji i metodu koja će služiti za „validiranje“ unesenih podataka. U slučaju da korisnik ne unese neku od vrijednosti na ekranu, potrebno mu je prikazati dijalog s opisom pogreške, npr.:



Slika 1. Dijalog s opisom pogreške kod unosa

U slučaju da je potrebno dodati „comboBox“ komponentu koja unaprijed nudi samo predefinirane vrijednosti (kao što je maksimalan broj memorijskih modula ili kapacitet memorijskog modula), to je moguće obaviti na sljedeći način:

```
@FXML
ComboBox<Integer> kapacitet;

@FXML
public void initialize() {
    kapacitet.getItems().addAll(Arrays.asList(1, 2, 4, 8));
}
```

5. Kreirati novi paket „hr.java.vjezbe.datoteke“ i u njemu klasu „Datoteke“ koja će sadržavati metode za spremanje svih vrsta komponenata u pripadajuće datoteke, npr. „spremiMaticnuPlocu“, „spremiProcesor“, „spremiRadnuMemoriju“ i „spremiTvrdiDisk“.
6. Kako bi se nova komponenta dodala u „ObservableArrayList“ objekt nakon spremanja u datoteku, mora biti dostupna za pozivanje u bilo kojoj klasi. To se može postići tako da se unutar klase „PocetniEkranController“ postavi statička varijabla:

```
public static ObservableList<Komponenta> komponente = null;
```

koja se inicijalno popunjava s podacima iz datoteke i prikazuje u komponenti za prikaz podataka o matičnim pločama:

```
komponente = FXCollections.observableArrayList(listaKomponenti);
leftListView.setItems(komponente);
```

Nakon spremanja komponente u datoteku, potrebno je pozvati i metodu za dodavanje te nove komponente i u „ObservableList“ listu, npr.:

```
Datoteke.spremiMaticnuPlocu(maticnaPloca);
PocetniEkranController.komponente.add(maticnaPloca);
```

#### MOGUĆNOSTI PROŠIRENJA ZADATAKA:

1. Umjesto četiri metode za spremanje komponenti napisati jednu generičku metodu koja omogućava spremanje svih komponenti.
2. Implementirati funkcionalnost pomoću koje će aplikacija nakon poruke „Uspješno spremljena komponenta“ ugasiti i prozor na kojem su se definirali podaci za novu komponentu.

3. Implementirati funkcionalnost da se nakon dodavanja nove komponente ista ne doda na kraj popisa svih komponenti, nego nakon posljednje komponente iste vrste (na primjer, ako se doda matična ploča, ona mora biti dodana nakon posljednje matične ploče, a ne na kraj popisa). To implementirati za sve vrste komponenata.

NAPOMENE:

1. Svi detalji koji nisu definirani potrebno je proizvoljno implementirati.
2. FXML datoteke moraju se nalaziti unutar paketa „hr.java.vjezbe.javaafx“ kako bi se mogle dohvaćati korištenjem metode npr. „getClass().getResource("NovaMaticnaPloca.fxml")“. Ako se nalaze na drugoj putanji, mora se promijeniti relativna putanja.