

## 8. Osmo laboratorijska vježba

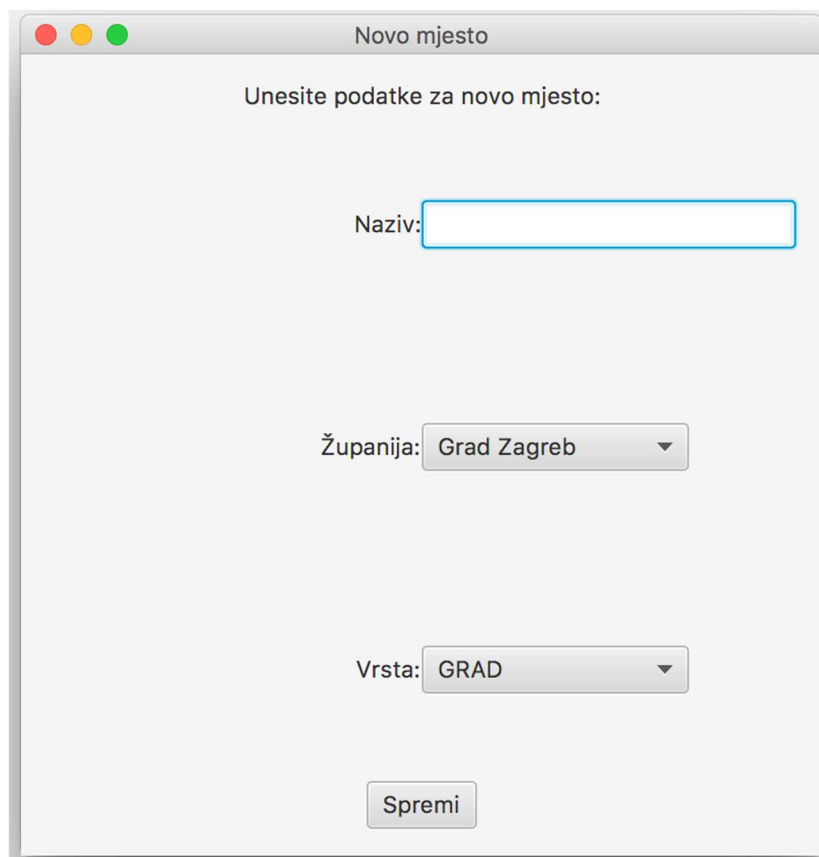
### 8.1. JAVA FX – NAPREDNO KORIŠTENJE

Svrha laboratorijske vježbe je proširivanje JavaFX aplikacije iz prethodne laboratorijske vježbe. Aplikacija mora podržavati funkcionalnosti za unos podataka u aplikaciju. Na Youtube video poveznici „<https://youtu.be/zq3Pbw4HmGU>“ se nalazi primjer unosa podataka o novim mjestima te je po uzoru na to potrebno implementirati unos podataka i za države, županije, senzore te mjerne postaje.

### 7.2. ZADATAK ZA PRIPREMU

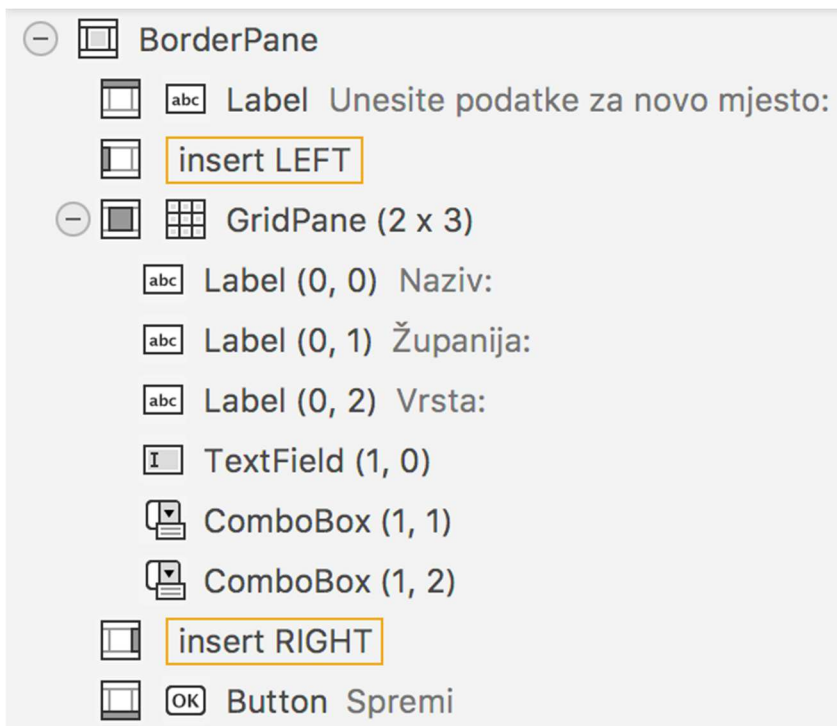
Nastaviti razvoj aplikacije iz prošle laboratorijske vježbe i oblikovati ekrane grafičkog sučelja aplikacije prema koracima u nastavku.

1. Kopirati „JavaFX“ projekt iz prošle laboratorijske vježbe nazvati ga prema vlastitom prezimenu sa sufiksom „8“.
2. Kreirati novu „FXML“ datoteku i korištenjem Scene Buildera otvoriti tu „FXML“ datoteku (korištenjem desne tipke miša i odabirom opcije „Open with Scene Builder“) dizajnirati ekran aplikacije za unos novog mjesta na način da izgleda ovako:



Slika 1. Izgled ekrana aplikacije za unos novog mjesta

Prijedlog rasporeda i hijerarhije izgleda ovako:



Slika 2. Hijerarhija komponenti

3. Kreirati klasu „DodajMjestoController“ koja će se povezati s ekranom za dodavanje novog mjesta. Klasa mora osim varijabli za povezivanje s elementima grafičkog sučelja sadržavati i metodu koja će se pozivati za spremanje podataka o novom mjestu u datoteku. U toj metodi potrebno je provjeriti je li korisnik ispravno unio sve podatke. Ako nije, aplikacija treba korisnika upozoriti što je potrebno promijeniti, a ako je sve u redu, aplikacija mora spremiti podatke u odgovarajuću tekstualnu datoteku. Ekran za unos novog mjesta mora se prikazivati u novom dijalogu te se automatski ugasiti nakon uspješnog spremanja novog mjesta. Osim toga, potrebno je automatski osvježiti listu mjesta da novo mjesto bude vidljivo na popisu. Primjer isječka koda kojim je to moguće implementirati je dan u nastavku (varijable „naziv“, „zupanija“, „ime“ i „vrstaMjesta“ su preuzete iz elemenata koji se nalaze na grafičkom sučelju):

```
String naziv = nazivTextfield.getText();
Zupanija zupanija = zupanijaCombobox.getValue();
VrstaMjesta vrstaMjesta = vrstaCombobox.getValue();
Mjesto mjesto = new Mjesto(naziv, zupanija, vrstaMjesta);
File mjestoFile = new File("dat/mjesta.txt");
int noviId = getZadnjiId() + 1;
try (FileWriter writer = new FileWriter(mjestoFile, true);) {
    writer.write(noviId + "\n");
    writer.write(naziv + "\n");
    writer.write(vrstaMjesta.toString() + "\n");
    writer.write(zupanija.getId() + "\n");
}
```

```
Alert alert = new Alert(AlertType.INFORMATION);
alert.setTitle("Uspješno spremanje mjesta!");
alert.setHeaderText("Uspješno spremanje mjesta!");
alert.setContentText("Uneseni podaci za mjesto su uspješno
spremljeni.");
Stage stage = (Stage) spremiButton.getScene().getWindow();
stage.close();
MjestaController.dodajNovoMjesto(mjesto);
} catch (IOException e) {
    logger.error("Pogreška kod spremanja podataka!", e);
    e.printStackTrace();
}
```

Metoda „dodajNovoMjesto“ u „MjestaController“ klasi može biti implementirana na način da se u statički objekt klase „ObservableList“ doda novokreirani objekt klase „Mjesto“:

```
public static void dodajNovoMjesto(Mjesto novoMjesto) {
    observableListaMjesta.add(novoMjesto);
}
```

Obzirom da se ID parametar povećava inkrementalno, posljednji ID može se dohvatiti na način da se uzme veličina Liste ili Mape:

```
public int getZadnjiId() {
    return svaMjesta.values().size();
}
```

Prikazivanje novog ekrana za unos podataka može se implementirati kreiranjem vlastitog „Stage“ objekta na sljedeći način:

```
public void prikaziEkranZaNovoMjesto() {
    try {
        BorderPane novoMjestoPane = FXMLLoader.load(Main.class
            .getResource("DodajMjestoEkran.fxml"));
        Scene scene = new Scene(novoMjestoPane, 600, 400);

        scene.getStylesheets().add(getClass().getResource(
            "application.css").toExternalForm());

        Stage stage = new Stage();
        stage.setScene(scene);
        stage.show();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

4. Na sličan način kao što je to implementirano za mjesta, istu funkcionalnost za dodavanje novih zapisa potrebno je implementirati i za države, senzore, županije te mjerne postaje, uz nužnu provjeru popunjenosti i ispravnosti podataka.

#### NAPOMENE:

1. Nakon svake promjene u „Scene Builder“ okruženju napraviti „Refresh“ projekta unutar Eclipse razvojnog okruženja.