

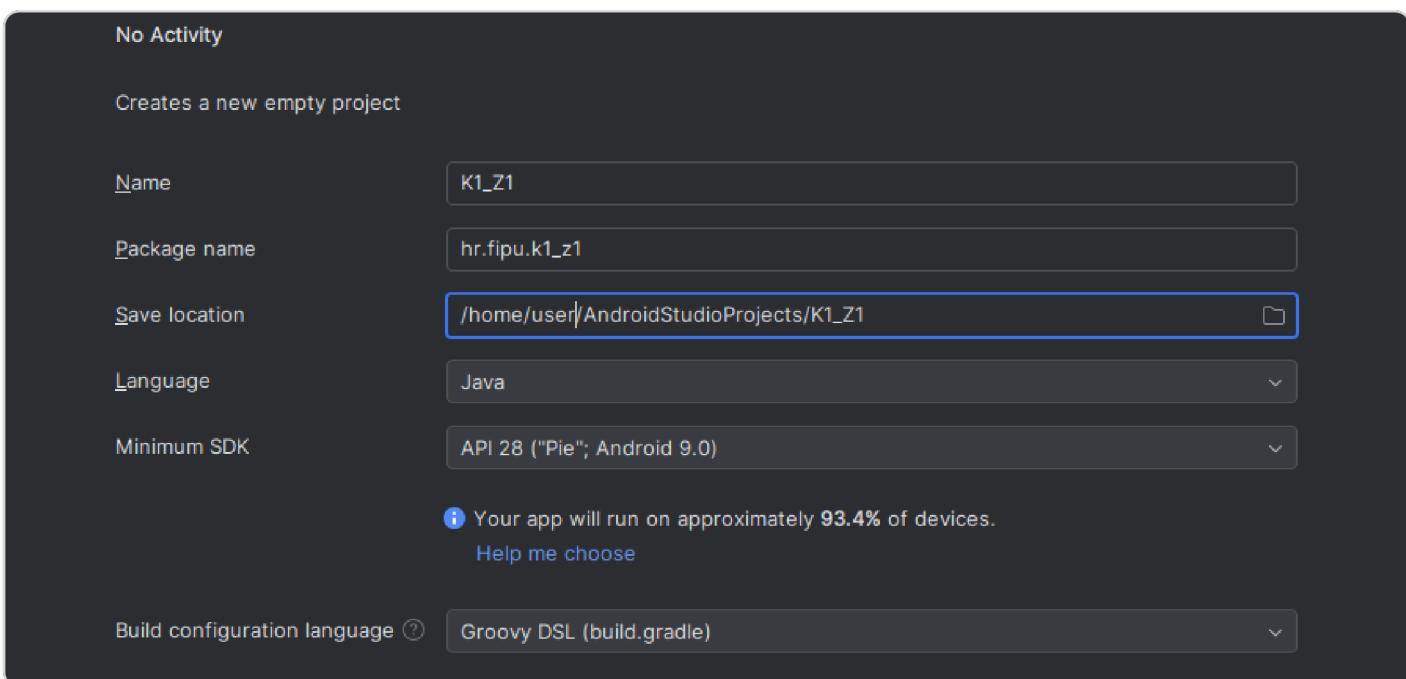
# Mobilne aplikacije

## Primjer kolokvija #1

Kolokvij nosi ukupno **25 bodova** i piše se **120 minuta**.

Za svaki projekt koristite zasebnu Android Studio instancu. Postavke projekta neka budu:

- Template: **No Activity**
- Project name: **za svaki zadatok zasebno, npr. K1\_Z1, K1\_Z2, K1\_Z3**
- Package name: **hr.fipu.(project\_name)**
- Save location: **po želji**
- Language: **Java**
- Minimum SDK: **API 28: (Pie) Android 9.0**
- Build configuration language: **Groovy DSL (build.gradle)**



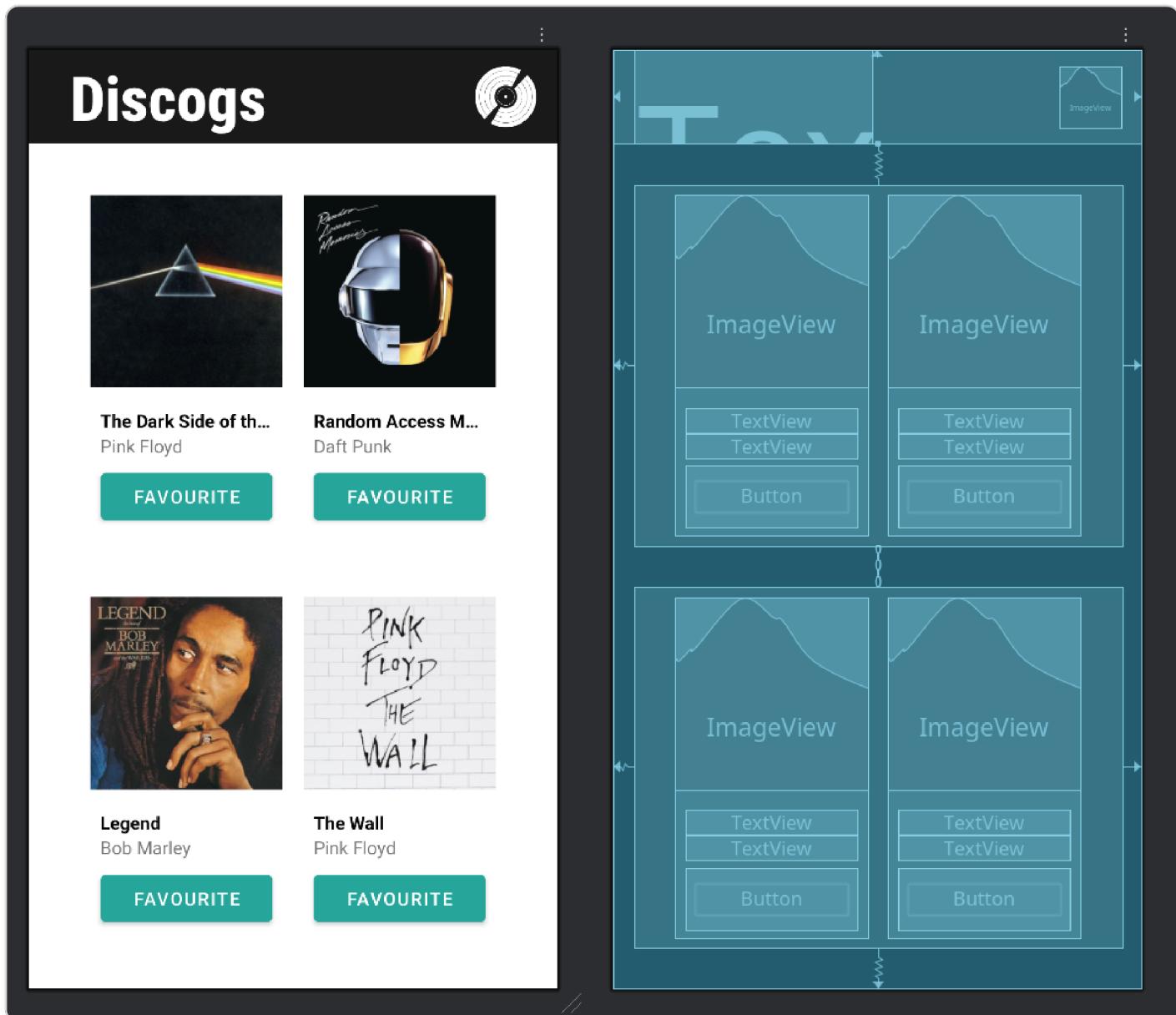
Postavke svakog projekta

**Napomena:** Pri rješavanju svih zadataka možete koristiti službenu Android dokumentaciju i materijale s predavanja. Nije dozvoljeno koristiti generative AI alate (*ChatGPT, GitHub Copilot, itd.*).

**Napomena:** Svaki zadatok predajte kao zaseban projekt, ne predajete sve zadatke u jednom projektu.

## Zadatak 1. (5 bodova)

Zadana je sljedeća slika korisničkog sučelja za Android aplikaciju, rekreirajte je koristeći **constraint layout** kao glavni layout:



Zadatak 1. - GUI za rekreirati

**Napomena:** Slike korištene u zadatku nalaze se u mapi `zadatak_1` kada preuzmete ovaj primjer kolokvija.

## Specifikacije korisničkog sučelja:

### Header

- Background boja: #191919

### Naslov "Discogs"

- Veličina teksta: 48sp
- Boja: Bijela
- Font stil: Bold
- Font obitelj: sans-serif-condensed-medium

### Logo

- Dimenzije: 48dp × 48dp

### Slike albuma

- Dimenzije: 15dp × 15dp

### Naslovi albuma

- Veličina teksta: 16sp
- Boja: Crna
- Font stil: Bold
- Redak: Single line

### Autori albuma

- Veličina teksta: 16sp
- Boja: Siva

### Dugma

- Background tint: #26A69A

**Napomena:** Razmak između elemenata neka bude sličan kao na slici, ali nije potrebno da bude identičan. Postoji više načina da se postigne sličan izgled.

## Zadatak 2. (12 bodova)

1. Izradite klasu `Knjiga` koja ima sljedeće atribute:

- o `String naslov`
- o `String autor`
- o `int godinaIzdavanja`
- o `String opis`

• Napišite konstruktor koji prima sve atribute kao parametre i inicijalizira ih. Također, implementirajte metode za dohvaćanje (gettere) svih atributa.

2. Izradite klasu `Knjiznica` koja sadrži listu objekata tipa `Knjiga`. Implementirajte sljedeće metode:

- o `void dodajKnjigu(Knjiga knjiga)` - dodaje knjigu u knjižnicu
- o `void ukloniKnjigu(String naslov)` - uklanja knjigu iz knjižnice prema naslovu
- o `Knjiga nadjiKnjigu(String naslov)` - pronađi i vraća knjigu prema naslovu
- o `List<Knjiga> dohvatiSveKnjige()` - vraća listu svih knjiga u knjižnici

3. `MainActivity` klasa treba prikazivati listu svih knjiga. Korisničko sučelje treba sadržavati:

- o `EditText` za unos naslova knjige za brisanje ili otvaranje detalja
- o Dugme za dodavanje nove knjige koje otvara `DodajKnjiguActivity`
- o Dugme za brisanje knjige
  - Prilikom klika na dugme, knjiga čiji je naslov unesen u `EditText` treba biti obrisana iz knjižnice
  - Prikazati poruku o uspješnom brisanju knjige
  - Nakon brisanja, ažurirati prikaz liste knjiga
  - Ako knjiga nije pronađena, prikazati odgovarajuću poruku
- o Dugme za prikaz detalja knjige koje otvara `DetaljiKnjigeActivity`
  - Prikazati detalje knjige čiji je naslov unesen u `EditText`
  - Ako knjiga nije pronađena, prikazati odgovarajuću poruku
- o `TextView` za prikaz liste knjiga (*naslov i autor*)

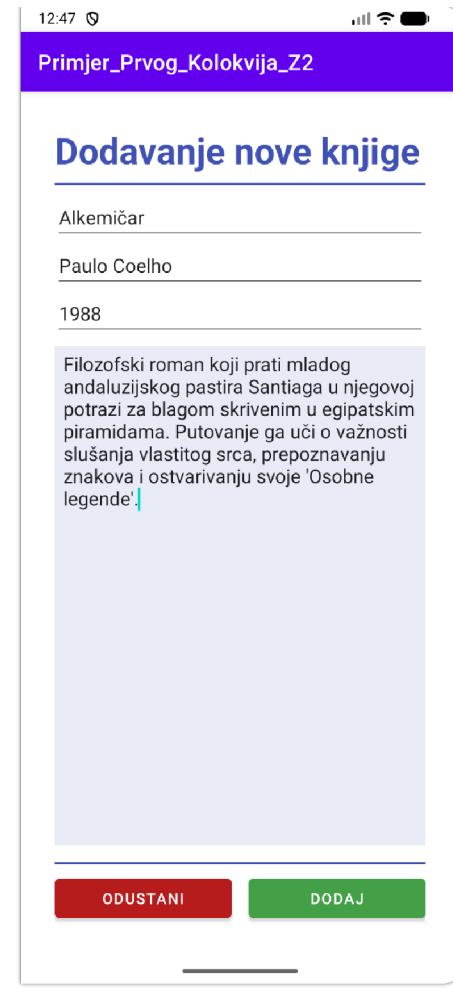
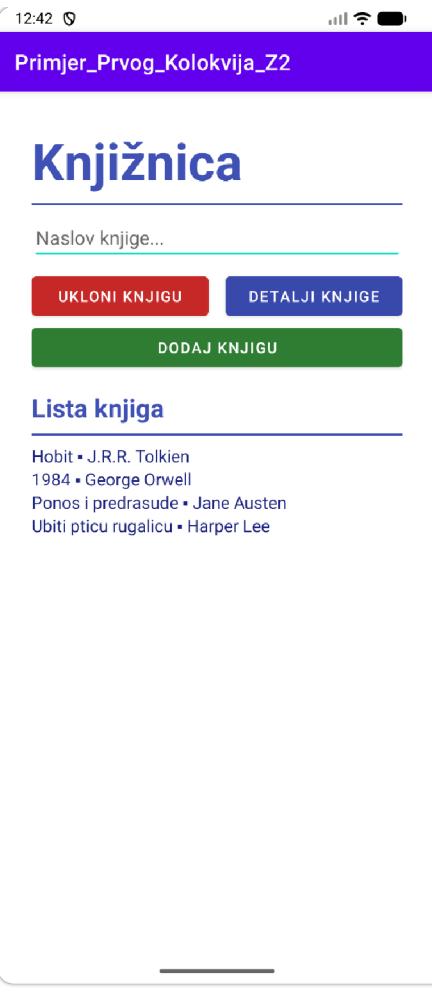
4. `DodajKnjiguActivity` klasa treba omogućiti korisniku da unese detalje nove knjige putem korisničkog sučelja i doda je u knjižnicu.

- o Korisničko sučelje treba sadržavati polja za unos naslova, autora, godine izdavanja i opisa knjige, te dugme za dodavanje knjige
- o Nakon unosa podataka i klika na dugme, nova knjiga treba biti dodana u knjižnicu unutar `MainActivity` koristeći **extra** proslijedene podatke nakon zatvaranja `DodajKnjiguActivity` i povratka na `MainActivity`
  - Prikazati poruku o uspješnom dodavanju knjige
  - Spriječiti dodavanje knjige ako je već postoji knjiga s istim naslovom, te prikazati odgovarajuću poruku

5. DetaljiKnjigeActivity klasa treba prikazati detalje knjige čiji je naslov unesen u MainActivity.
- Korisničko sučelje treba sadržavati TextView za prikaz naslova, autora, godine izdavanja i opisa knjige
  - Dugme za povratak na MainActivity

Podaci:

```
knjiznica.dodajKnjigu(new Knjiga("Hobit", "J.R.R. Tolkien", 1937,  
    "Fantastični roman koji prati avanture hobita Bilba Bagginsa dok putuje  
    Međuzemljem kako bi povratio blago koje je zmaj Smaug oteo patuljcima. Knjiga  
    je uvod u bogati svijet 'Gospodara prstenova' i predstavlja čitateljima mnoge  
    rase i mesta koja će kasnije postati ključna."));  
knjiznica.dodajKnjigu(new Knjiga("1984", "George Orwell", 1949,  
    "Distopijski roman koji opisuje život u totalitarnoj državi Oceaniji, gdje  
    Veliki Brat sve nadzire, a misao je zločin. Glavni lik, Winston Smith, radi za  
    Ministarstvo istine i potajno se buni protiv režima, što ga dovodi u veliku  
    opasnost."));  
knjiznica.dodajKnjigu(new Knjiga("Ponos i predrasude", "Jane Austen", 1813,  
    "Ljubavni roman koji istražuje složene društvene odnose i pitanja braka,  
    morala i odgoja u Engleskoj početkom 19. stoljeća kroz priču o Elizabeth Bennet  
    i gospodinu Darcyju. Kroz duhovite dijaloge i složene likove, Austen kritizira  
    društvene konvencije svog vremena."));  
knjiznica.dodajKnjigu(new Knjiga("Ubiti pticu rugalicu", "Harper Lee", 1960,  
    "Roman smješten na američkom Jugu tijekom Velike depresije, koji se bavi  
    ozbiljnim temama rasne nepravde i gubitka nevinosti, promatranim očima  
    djevojčice Scout Finch. Njezin otac, Atticus Finch, odvjetnik je koji brani  
    crnca lažno optuženog za silovanje bjelkinje, što izaziva bijes u njihovoj  
    zajednici."));  
knjiznica.dodajKnjigu(new Knjiga("Alkemičar", "Paulo Coelho", 1988,  
    "Filozofski roman koji prati mladog andaluzijskog pastira Santiaga u  
    njegovoj potrazi za blagom skrivenim u egipatskim piramidama. Putovanje ga uči  
    o važnosti slušanja vlastitog srca, prepoznavanju znakova i ostvarivanju svoje  
    'Osobne legende'."));
```



Zadatak 2. – primjer sučelja

**Napomena:** Izgled korisničkog sučelja nije strogo definiran, ali treba sadržavati sve tražene elemente i biti funkcionalan.

## Zadatak 3. (8 bodova)

Zadan je sljedeći xml layout:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingStart="40px"
    android:paddingLeft="40px"
    android:paddingTop="256px"
    android:paddingEnd="40px">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Self removing blue buttons!"
        android:textSize="24sp" />

    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="#C3C3C3"
        android:text="Click Me"
        android:textColor="#000000" />

    <LinearLayout
        android:id="@+id/buttonContainer"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" />
</LinearLayout>
```

Rekreirajte ga programski u `MainActivity` klasi koristeći Java kôd, bez korištenja XML layout datoteke.

- Postavite elemente kao u zadanom XML-u.
  - Pritiskom na glavno dugme ("Click Me"), dodajte plavo dugme u `LinearLayout` container
  - Svako plavo dugme treba imati event listener koji će ga izbrisati kada se na njega klikne

1:15



## Primjer\_Prvog\_Kolokvija\_Z3

Self removing blue buttons!

CLICK ME

4. UKLONI ME

5. UKLONI ME

9. UKLONI ME

10. UKLONI ME

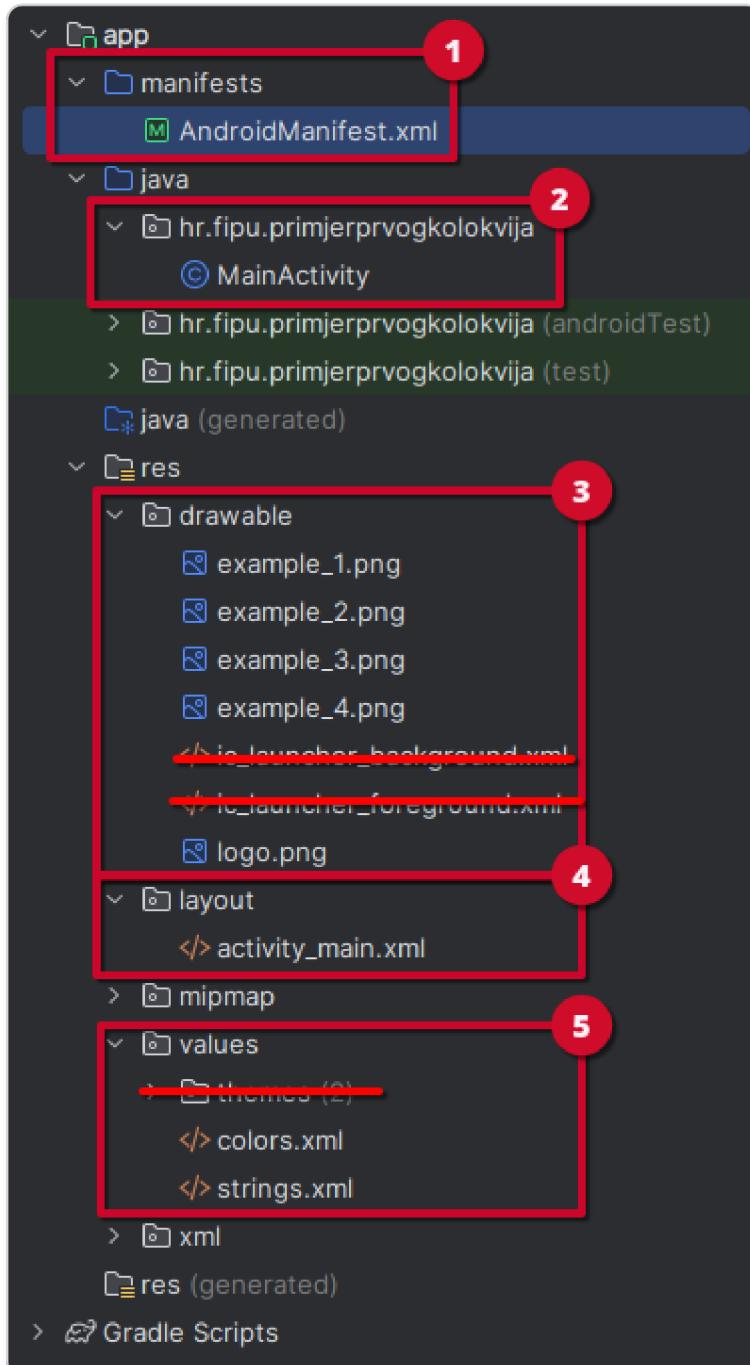
12. UKLONI ME

13. UKLONI ME

## Predajte sljedeće:

- Za svaki zadatak zasebno predajete .zip datoteku koja sadrži strukturu projekta:

1. app/manifests/ - sadrži `AndroidManifest.xml` datoteku
2. app/java/direktorij/ - sadrži `.java` datoteke
3. app/res/drawable/ - sadrži slike korištene u aplikaciji
4. app/res/layout/ - sadrži `.xml` datoteke za layoute
5. app/res/values/ - sadrži `colors.xml` & `strings.xml` datoteke



*Primjer mapa i datoteka za predati*

**Napomena:** Svaki zadatak predajte kao zaseban projekt, ne predajte sve zadatke u jednom projektu.