



Agencija za
strukovno obrazovanje
i obrazovanje odraslih

Razvijajmo
kompetencije
zajedno!



4. Dani strukovnih nastavnika 2020

Razvoj android aplikacija kroz prizmu
različitih predložaka dizajna

Doc. dr. sc. Tomislav Jakopec



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

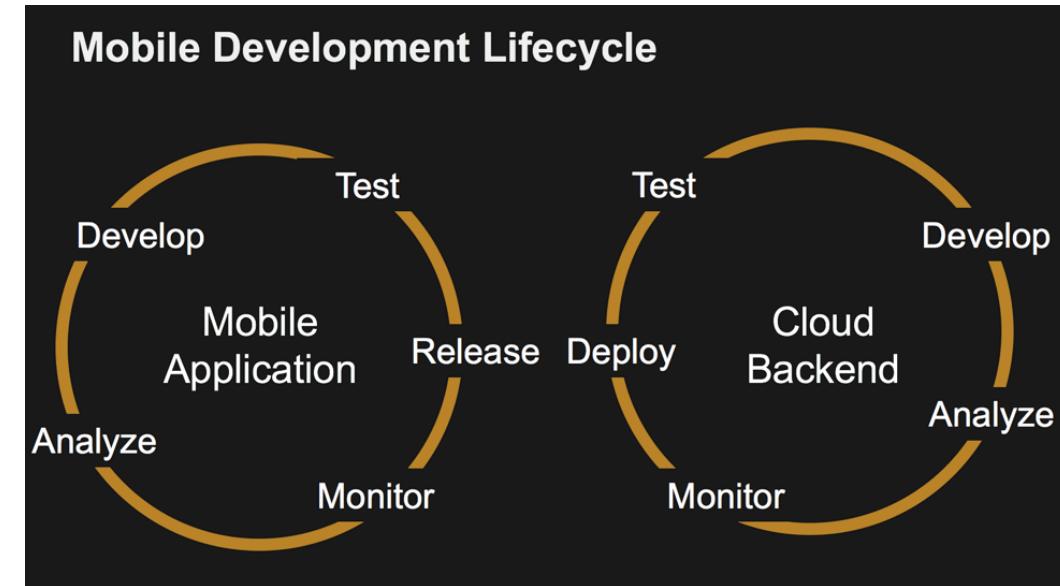
- Načini razvoja mobilnih aplikacija
- REST API
- Android preduvjeti
- Hello world aplikacija
- Alati (biblioteke) za lakši rad
- Principi u razvoju Android aplikacija
- MVP predložak dizajna
- MVVM predložak dizajna
- Pitanja



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

- Razvoj „platformskih“ (native) aplikacija
- Razvoj više platformskih aplikacija
 - Hibridne aplikacije
 - Među platformske aplikacije
 - Progresivne mrežne aplikacije
- Razvoj mobilnih aplikacija oblaku



Izvori:

<https://aws.amazon.com/mobile/mobile-application-development/>

<https://www.guru99.com/mobile-app-development-tools.html>

<https://appinventor.mit.edu/>

<https://hackernoon.com/list-of-best-mobile-app-development-tools-in-current-industry-6881cc6bd8d2>

<https://mopinion.com/mobile-development-tools-an-overview/>

<https://railsware.com/blog/native-vs-hybrid-vs-cross-platform/>

<https://www.upwork.com/hiring/mobile/should-you-build-a-hybrid-mobile-app/>



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Razvoj „platformskih“ (native) aplikacija

Android

- jezici
 - **Java**
 - **Kotlin**



- Razvojni alati
 - **Android studio**
 - MS Visual studio
 - Eclipse for Android Developers
 - Android IDE
 - IntelliJ-IDEA

The screenshot shows the official Android Studio website. At the top, there's a navigation bar with links for Developers, Platform, Android Studio (which is underlined), Google Play, Jetpack, Docs, and News. Below the navigation is a search bar and language selection. The main content area features a large green Android robot icon and the text "android studio". It says "Android Studio provides the fastest tools for building apps on every type of Android device." There are two prominent download buttons: "DOWNLOAD ANDROID STUDIO" and "DOWNLOAD OPTIONS". Below these are links for "8.4.1 for Windows 64-bit (871 MB)" and "RELEASE NOTES". The bottom part of the screenshot shows the Android Studio IDE interface with a project structure on the left and code editor windows on the right.



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



Razvoj „platformskih“ (native) aplikacija



iOS

- jezici
 - Objective-C
 - Swift
- Razvojni alati
 - XCode
 - MS Visual studio
 - AppCode
 - Atom

A screenshot of the Xcode 11 IDE interface. The top navigation bar includes links for Developer, Discover, Design, Develop, Distribute, Support, Account, and a search icon. Below the navigation bar, the main window shows the Xcode 11 interface with a sidebar containing project files and a central editor window displaying Swift code. On the right side, there are three iPhone simulators showing the Earth's globe as the home screen. The overall theme is dark mode.



Razvoj „platformskih“ (native) aplikacija

4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!

Windows Phone

- jezik
 - C#
- Razvojni alati
 - MS Visual studio



Microsoft | Visual Studio | Visual Studio 2019 | Features | Editions | Downloads | More | Free Visual Studio | All Microsoft | Search | Sign in

Mobile app development

Build cloud-connected cross platform mobile apps and games for iOS, Android, and Windows

[Download Visual Studio](#)

[Documentation](#)

A single toolchain across mobile and cloud



Native apps with .NET

Visual Studio tools for Xamarin

Create rich native apps using .NET across iOS, Android & Windows. Share code across device platforms with .NET Standard. Share UI with Xamarin.Forms and XAML for maximum code reuse. With 100% of the native APIs exposed, you have full access to device capabilities.



REST API



**REpresentational State Transfer
Application Program Interface**

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Izvor:

<https://developers.google.com/photos/library/guides/about-restful-apis>

<https://medium.com/google-cloud/building-api-services-a-beginners-guide-7274ae4c547f>

<https://devhints.io/rest-api>

HTTP METODE

GET

Zahtjev vraća resurs bez dodatnih aktivnosti

POST

Zahtjev kreira resurs (datoteka, baza podataka, API poziv)

PUT

Zahtjev mijenja stanje resursa (datoteka, baza podataka, API poziv)

DELETE

Zahtjev briše resurs (datoteka, baza podataka, API poziv)



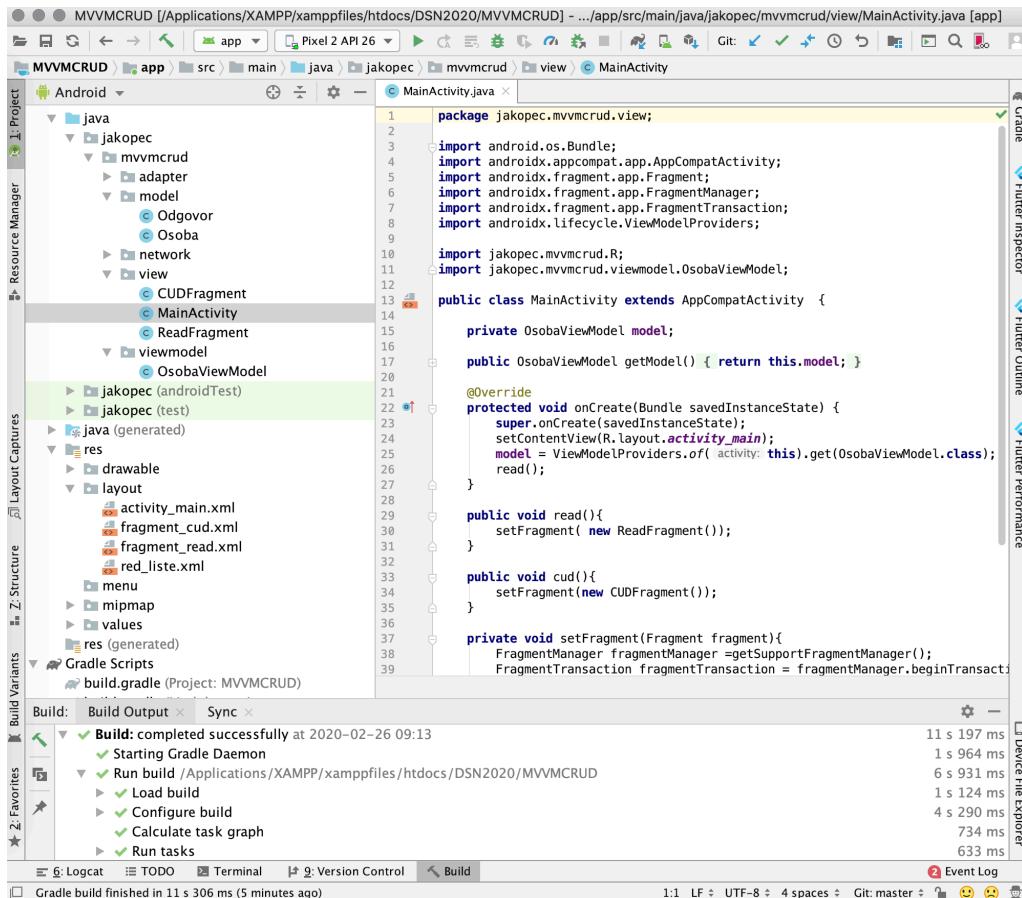
Android Preduvjeti



android studio <https://developer.android.com/studio>

4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!



```

package jakopec.mvvmcrud.view;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;
import androidx.lifecycle.ViewModelProviders;
import jakopec.mvvmcrud.R;
import jakopec.mvvmcrud.viewmodel.OsobaViewModel;

public class MainActivity extends AppCompatActivity {

    private OsobaViewModel model;

    public OsobaViewModel getModel() { return this.model; }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        model = ViewModelProviders.of(this).get(OsobaViewModel.class);
        read();
    }

    public void read(){
        setFragment( new ReadFragment());
    }

    public void cud(){
        setFragment(new CUDFragment());
    }

    private void setFragment(Fragment fragment){
        FragmentManager fragmentManager = getSupportFragmentManager();
        FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
        fragmentTransaction.replace(R.id.fragmentContainer, fragment);
        fragmentTransaction.commit();
    }
}

```



1. Android Studio's emulator
2. ARChon
3. Bliss OS
4. Bluestacks
5. GameLoop
6. Genymotion
7. LDPlayer
8. MEmu
9. Nox
10. Phoenix OS
11. PrimeOS
12. Remix OS Player
13. Xamarin
14. YouWave
15. Build your own

<https://www.androidauthority.com/best-android-emulators-for-pc-655308/>



Hello world aplikacija

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Welcome to Android Studio

Android Studio Version 3.5.1

- + Start a new Android Studio project
- Start a new Flutter project
- Open an existing Android Studio project
- Check out project from Version Control
- Profile or debug APK
- Import project (Gradle, Eclipse ADT, etc.)
- Import an Android code sample

Configure Get Help

Create New Project

Choose your project

Phone and Tablet Wear OS TV Android Auto Android Things

- Add No Activity
- Basic Activity
- Empty Activity
- Bottom Navigation Activity
- Fragment + ViewModel
- Fullscreen Activity
- Master/Detail Flow
- Navigation Drawer Activity

Empty Activity

Creates a new empty activity

Cancel Previous Next Finish

Create New Project

Configure your project

Name: HelloWorld

Package name: jakopc.helloworld

Save location: /Users/tjakopc/AndroidStudio/HelloWorld

Language: Java

Minimum API level: API 22: Android 5.1 (Lollipop)

Your app will run on approximately 80.2% of devices.

This project will support instant apps

Use androidx.* artifacts

Creates a new empty activity

Cancel Previous Next Finish

HelloWorld (~/AndroidStudio/HelloWorld) - .../app/src/main/java/jakopc/helloworld/MainActivity.java [app]

MainActivity

```
package jakopc.helloworld;
import ...
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Project Manager

Build Variants

Build Sync

Build: Synced successfully at 2019-10-23 08:39

- Starting Gradle Daemon
- Run build /Users/tjakopc/AndroidStudio/HelloWorld
- Load build
- Configure build
- Calculate task graph
- Run tasks

Event Log

* daemon started successfully



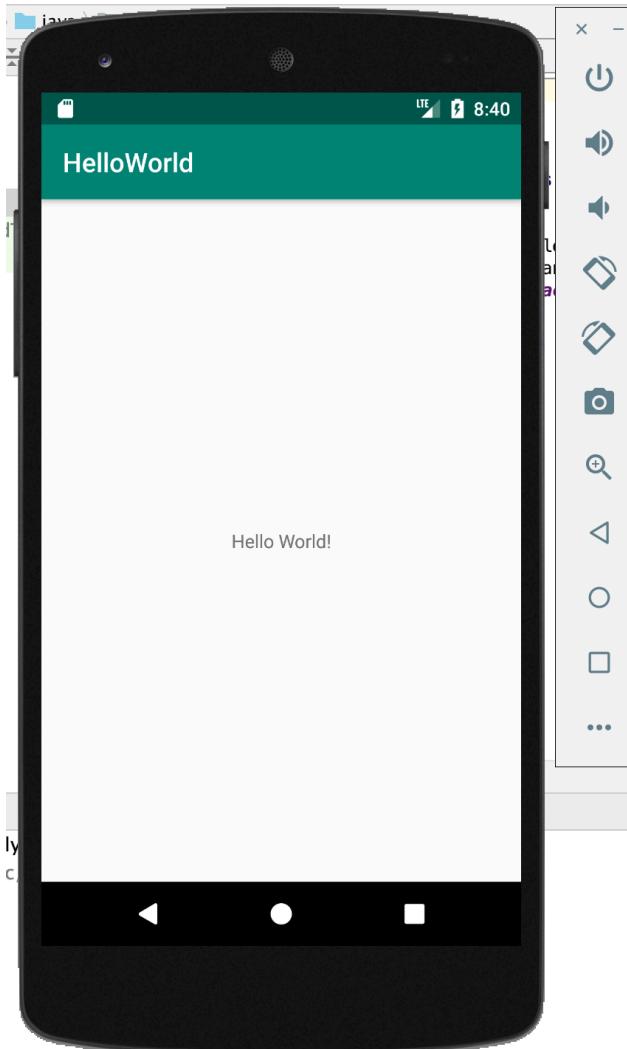


Hello world aplikacija



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



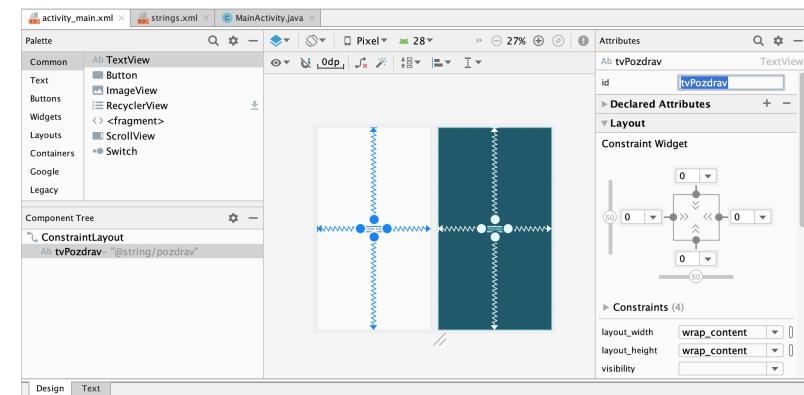
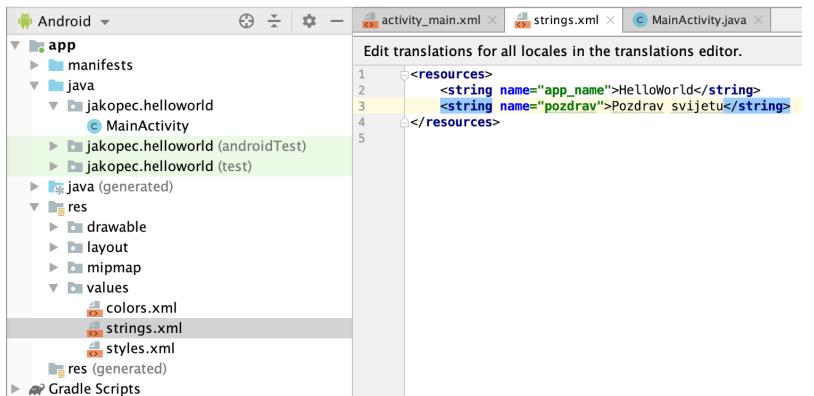
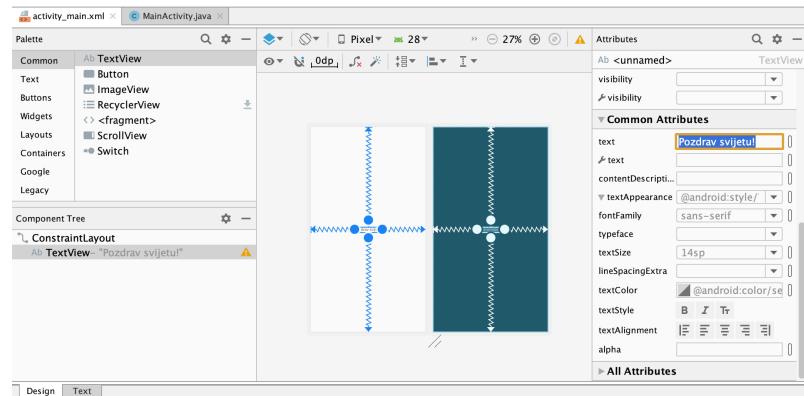


Hello world aplikacija



4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!



```
1 package jakopec.helloworld;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10        setContentView(R.layout.activity_main);
11        TextView tvPozdrav = findViewById(R.id.tvPozdrav);
12        tvPozdrav.setText("Pogramski pozdrav");
13    }
14
15 }
```

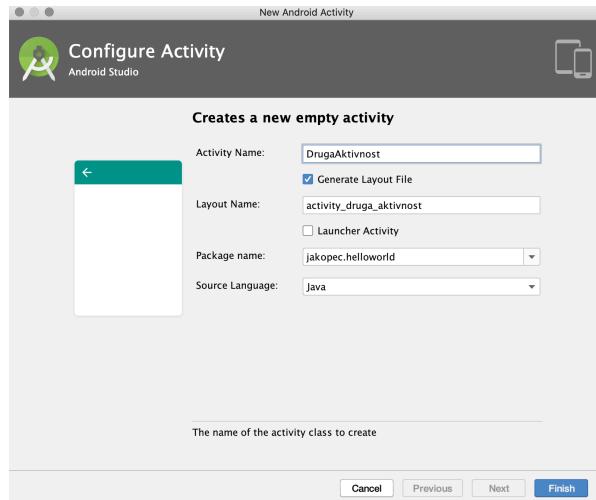
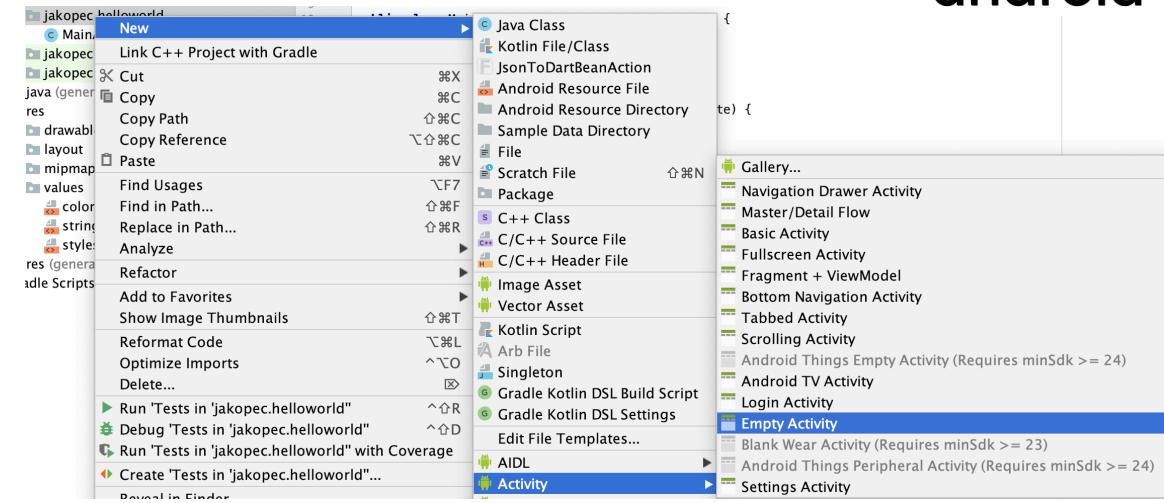
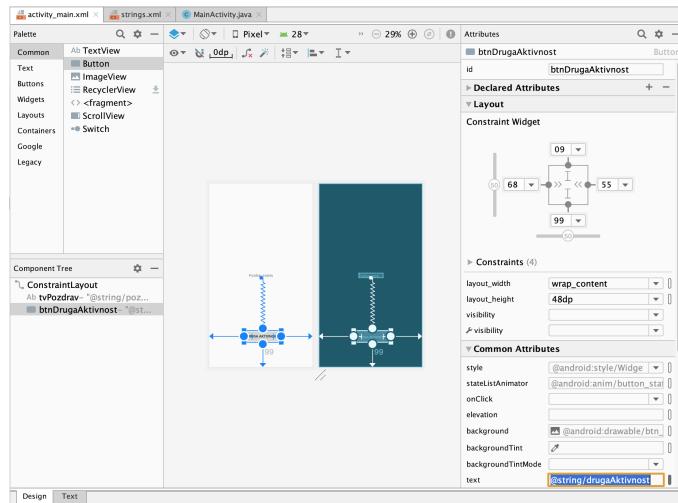


Hello world aplikacija



4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!



```
1 package jakopec.helloworld;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     private TextView tvPozdrav;
8     private Button btnDrugaAktivnost;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14
15        tvPozdrav = findViewById(R.id.tvPozdrav);
16        tvPozdrav.setText("Pogramski pozdrav");
17
18        btnDrugaAktivnost = findViewById(R.id.btnDrugaAktivnost);
19        btnDrugaAktivnost.setOnClickListener(new View.OnClickListener() {
20            public void onClick(View v) {
21                otvoridrugaAktivnost();
22            }
23        });
24
25    }
26
27    private void otvoridrugaAktivnost() {
28        Intent intent = new Intent(getApplicationContext(), DrugaAktivnost.class);
29        startActivity(intent);
30    }
31
32
33
34
35
36
37
38 }
```

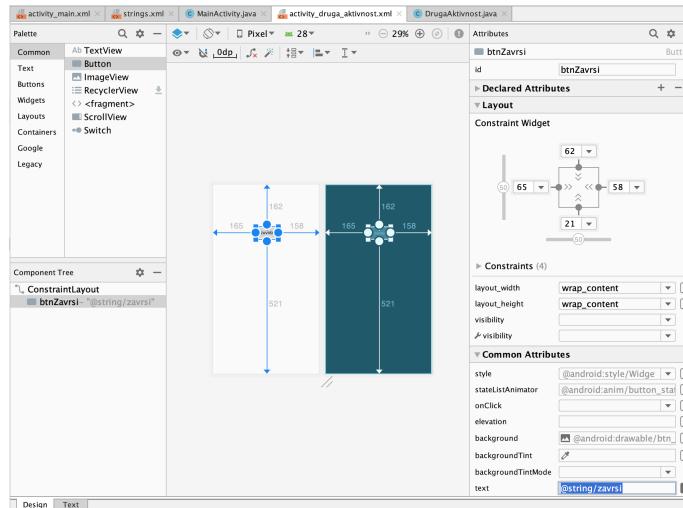


Hello world aplikacija

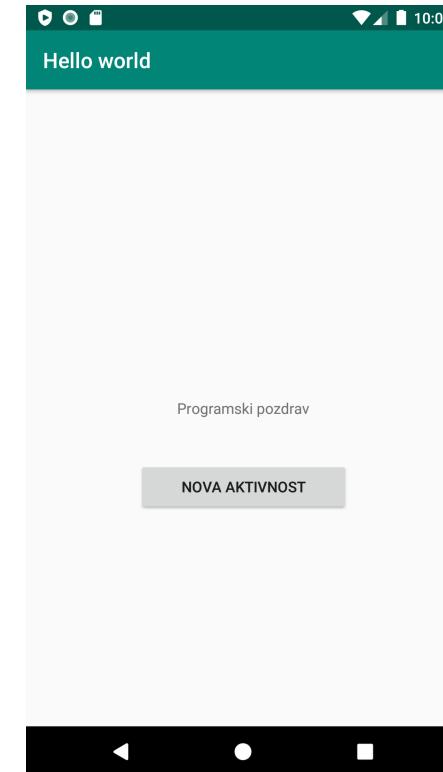


4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!



```
1 package jakopec.helloworld;
2
3 import ...
4
5 public class DrugaAktivnost extends AppCompatActivity {
6
7     private Button btnZavrsi;
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_druga_aktivnost);
13
14        btnZavrsi = findViewById(R.id.btnZavrsi);
15
16        btnZavrsi.setOnClickListener(new View.OnClickListener() {
17            @Override
18            public void onClick(View view) {
19                zavrsi();
20            }
21        });
22
23    }
24
25    private void zavrsi() { finish(); }
26
27
28
29
30
31
32 }
```



Programski pozdrav

NOVA AKTIVNOST



ZAVRŠI



Hello world aplikacija



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!





Osnovni preuvjeti

manifest.xml

4. Dani strukovnih
nastavnika 2020

```
<uses-permission android:name="android.permission.INTERNET" />
```

build.gradle -> android

```
compileOptions {  
    sourceCompatibility JavaVersion.VERSION_1_8  
    targetCompatibility JavaVersion.VERSION_1_8  
}
```

Razvijajmo
kompetencije
zajedno!



Lombok

dodatak za IDE koji omogućuje kreiranje standardnih mogućnosti putem anotacija (get, set, konstruktor,...)

Preduvjeti:

Instalirati Lombok plugin za Android studio



4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!

Izvor:

<https://projectlombok.org/>

Pojo

```
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
public class Osoba implements Serializable {

    private int id;
    private String ime;
    private String prezime;
    private String urlSlika;
}
```

U build.gradle -> dependencies dodati zavisnosti

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'androidx.appcompat:appcompat:1.1.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test.ext:junit:1.1.1'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
    compileOnly 'org.projectlombok:lombok:1.18.10'
    annotationProcessor 'org.projectlombok:lombok:1.18.10'
}
```

Korištenje

```
osoba.get|
```

- I m getId()
- I m getIme()
- I m getPrezime()
- I m getUrlSlika()



Razvijajmo
kompetencije
zajedno!

Butter Knife

vezivanje polja i metoda u Android view klasama

Preduvjet:

U build.gradle -> dependencies dodati zavisnosti

```
implementation 'com.jakewharton:butterknife:10.2.1'  
annotationProcessor 'com.jakewharton:butterknife-compiler:10.2.1'
```

Korištenje

```
@BindView(R.id.lista)  
RecyclerView recyclerView;  
  
@Override  
protected void onCreate(Bundle savedInstanceState)  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    ButterKnife.bind(target: this);  
}
```

Nema potrebe za: `RecyclerView recyclerView = findViewById(R.id.lista);`

Izvor:

<https://jakewharton.github.io/butterknife/>

Izgled klase MainActivity

```
public class MainActivity extends AppCompatActivity implements AdapterListe.ItemClickListener {  
  
    @BindView(R.id.lista)  
    RecyclerView recyclerView;  
  
    AdapterListe adapter;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        ButterKnife.bind(target: this);  
        recyclerView.setLayoutManager(new LinearLayoutManager(context: this));  
        adapter = new AdapterListe(context: this);  
        adapter.setClickListener(this);  
        recyclerView.setAdapter(adapter);  
    }  
  
    @Override  
    public void onItemClick(View view, int position) {  
    }  
}
```



Retrofit

HTTP klijent

Preduvjet:

U build.gradle -> dependencies dodati zavisnosti

```
implementation 'com.squareup.retrofit2:retrofit:2.3.0'  
implementation 'com.squareup.retrofit2:converter-gson:2.3.0'
```

Korištenje

Sučelje

```
public interface OsobeRESTSucelje {  
  
    @GET("osobe")  
    Call<Odgovor> dohvatiOsobe();  
}
```

Implementacija

```
private void ucitaj(){  
    Retrofit retrofit = new Retrofit.Builder()  
        .baseUrl(REST_URL)  
        .addConverterFactory(GsonConverterFactory.create())  
        .build();  
    OsobeSucelje osobeSucelje = retrofit.create(OsobeSucelje.class);  
    Call<Odgovor> call = osobeSucelje.dohvatiOsobe();  
    call.enqueue(new Callback<Odgovor>() {  
        @Override  
        public void onResponse(Call<Odgovor> call, Response<Odgovor> response) {  
            adapter.setPodaci(response.body().getOsobe());  
            adapter.notifyDataSetChanged();  
        }  
  
        @Override  
        public void onFailure(Call<Odgovor> call, Throwable t) {  
            Toast.makeText(context: MainActivity.this, text: "Nešto nije u redu",  
                           Toast.LENGTH_LONG).show();  
        }  
    });  
}
```

Izvor:

<https://square.github.io/retrofit/>



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Picasso

Preuzimanje i prilagodba slika

Preduvjet:

U build.gradle -> dependencies dodati zavisnosti

```
implementation 'com.squareup.picasso:picasso:2.71828'
```

Korištenje

```
Picasso.get().load(osoba.getUrlSlika()).into(slika);
```

Izvor:

<https://square.github.io/picasso/>



Ostalo

- Room (Pohrana na uređaju)
- Robolectric (Testiranje)
- Mockito (Testiranje)
- Glide (Rad s slikama)
- Parceler (Generiranje koda)
- IcePick (Stanje instanci)
- LeakCanary (Hvatanje curenja memorije)
- Espresso (Testiranje)
- Robolectric (Testiranje)
- Dagger 2 (eng. Dependency injector)
- EventBus (Komunikacija)
-

Razvijajmo
kompetencije
zajedno!



Alati (biblioteke) za lakši rad



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!





SOLID principi

Single responsibility principle

Open/closed principle

Liskov substitution principle

Interface segregation principle

Dependency inversion principle

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Postiže se ponovo upotrebljiv,
lagan za održavanje i proširiv kod koji se lako testira.



Android aplikacije su kompleksne, ovisne o Android OS.

Razvoj kompleksnih android aplikacija zahtjeva definiranje više slojeva radi lakšeg razvoja, testiranja i održavanja

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Izvori:

<https://developer.android.com/jetpack/docs/guide>

<https://deviq.com/separation-of-concerns/>

<https://medium.com/@nicholas.rose/driving-your-ui-from-a-persistent-model-6136ce73c9>



Odvajanje zaduženja (Separation of concerns)

Sličan principu pojedinačne odgovornosti - Single Responsibility Principle (SOLID)

Odgovornosti Klase definiramo po principu razloga za promjenu.

Ako mislimo da postoji više od jednog razloga za promjenu klase, ta klasa tada ima više odgovornosti. Cilj je da jedna klasa bude pojedinačno odgovorna.

Jednostavnim jezikom:

Kada dolazimo u bolnicu ne postoji jedna osoba koja sve odradi. Primi nas medicinska sestra koji napravi trijažu, zatim nas pregleda liječnik, odemo na snimanje gdje nas obradi medicinski tehničar itd.

Primili smo zdravstvenu uslugu od sustava koji se sastoji od više dionika. Tako moramo graditi i aplikacije

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Izvori:

<https://developer.android.com/jetpack/docs/guide>

<https://deviq.com/separation-of-concerns/>

<https://softwareengineering.stackexchange.com/questions/32581/how-do-you-explain-separation-of-concerns-to-others>

<https://deviq.com/single-responsibility-principle/>

<https://ivandrevianko.com/wp-content/uploads/2013/10/Agile-Principles-Patterns-and-Practices-in-C.pdf>



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



MODERNIZACIJA SUSTAVA
STRUČNOG USAVRŠAVANJA
NASTAVNIKA STRUKOVNIH PREDMETA

Odvojiti izgled (UI) od modela

Model su komponente koje su odgovorne za upravljanje podacima.

Podaci bi trebali biti pohranjeni na uređaju iz dva razloga:

1. U slučaju zatvaranja aplikacije od strane Android OS ne gube se podaci
2. Aplikacija nastavlja raditi u slučaju gubitka veze s serverom

Izgled i model su neovisni jedan od drugoga pa ih treba odvojiti

Odvajanjem izgleda i modela aplikacija je prikladna za automatsko testiranje

Držati UI kod jednostavnim i oslobođenim od poslovne logike kako bi ga se lakše održavalo

Izvori:

<https://developer.android.com/jetpack/docs/guide>

<https://medium.com/@nicholas.rose/driving-your-ui-from-a-persistent-model-6136ce73c9>

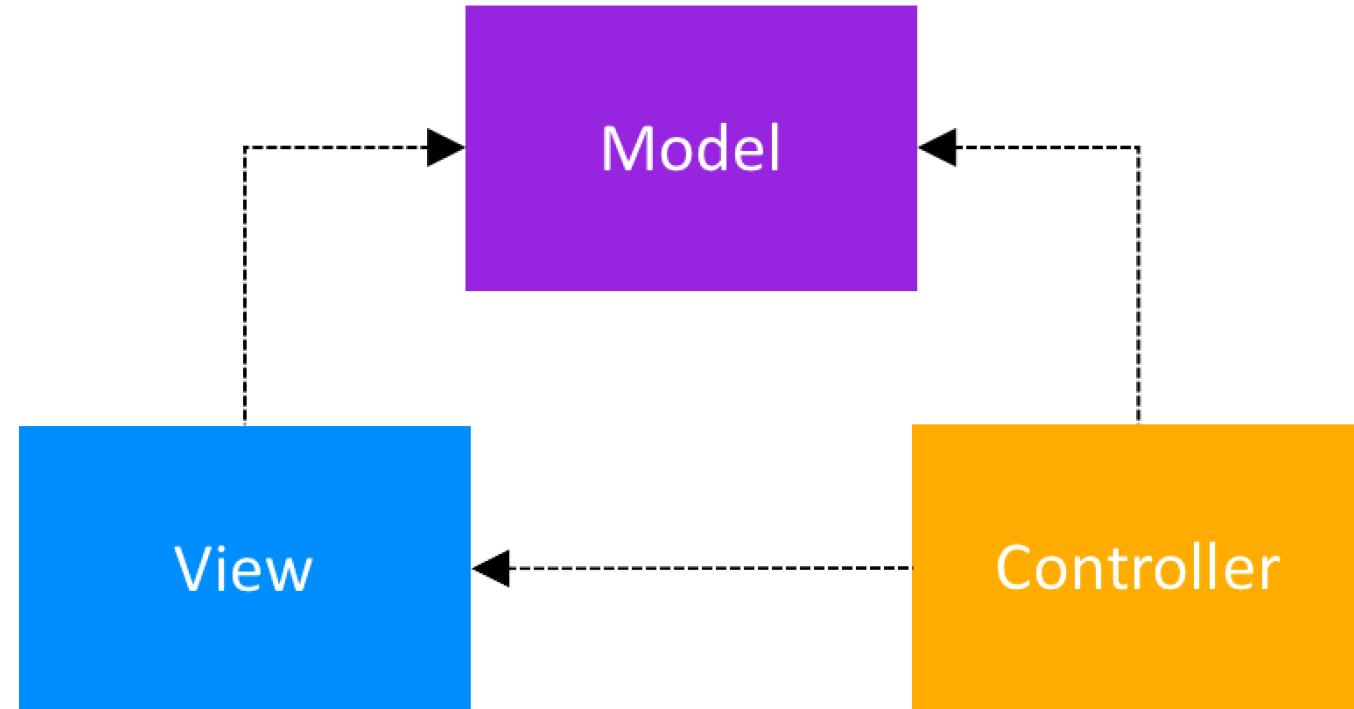
<https://proandroiddev.com/mvvm-architecture-viewmodel-and-livedata-part-1-604f50cda1>



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

Predložak dizajna MVC



Izvor:

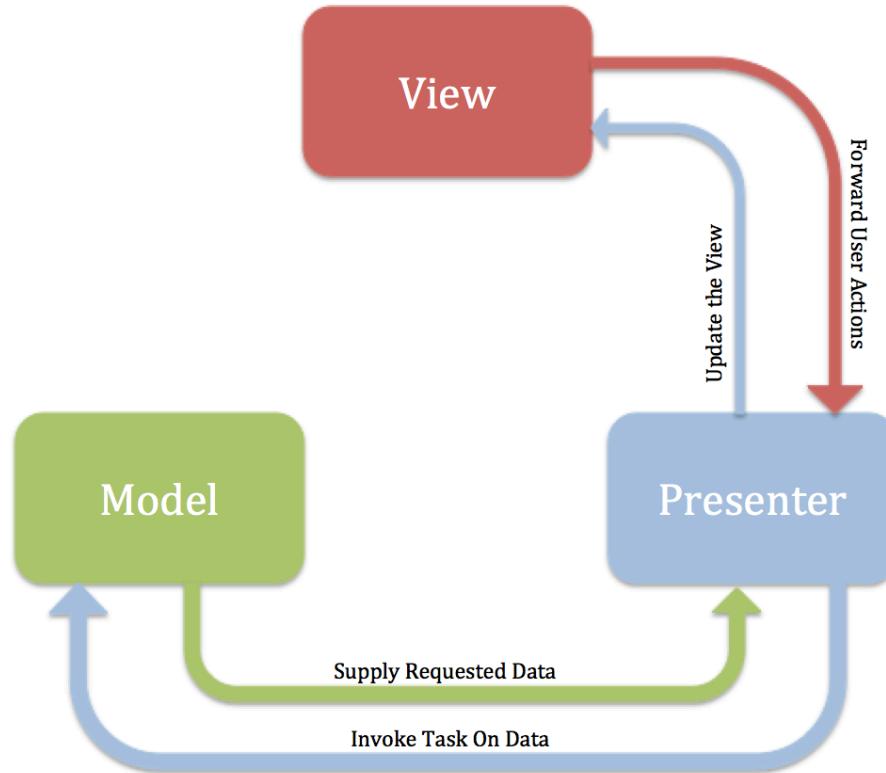
<https://medium.com/upday-devs/android-architecture-patterns-part-1-model-view-controller-3baecef5f2b6>



Predložak dizajna MVP

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



Izvor:

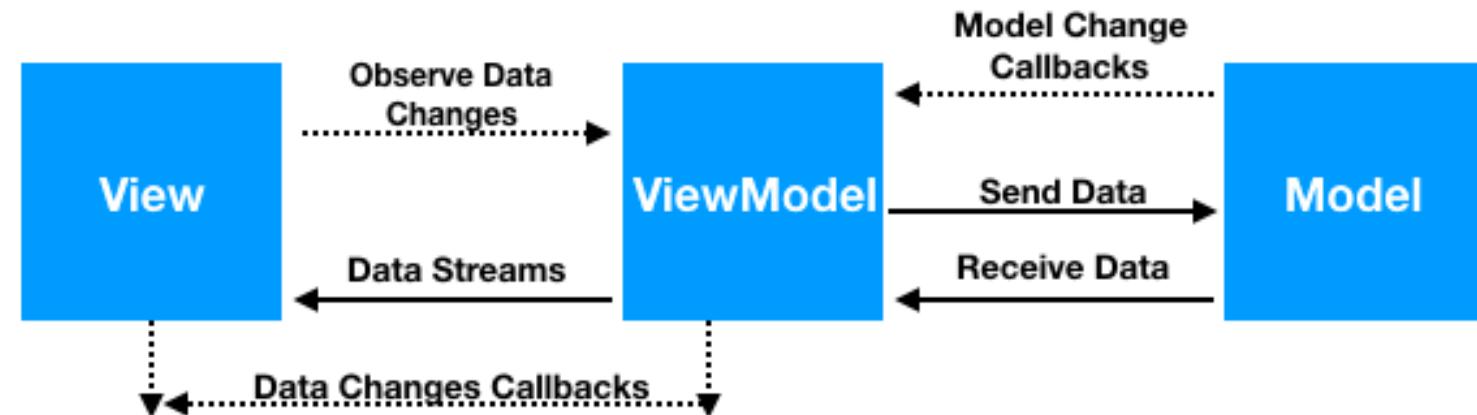
<https://medium.com/cr8resume/working-with-mvp-and-retrofit-2-in-android-b771b8369ec0>



Predložak dizajna MVVM

4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!



Izvor:

<https://www.journaldev.com/20292/android-mvvm-design-pattern>



MVC vs. MVP vs. MVVM

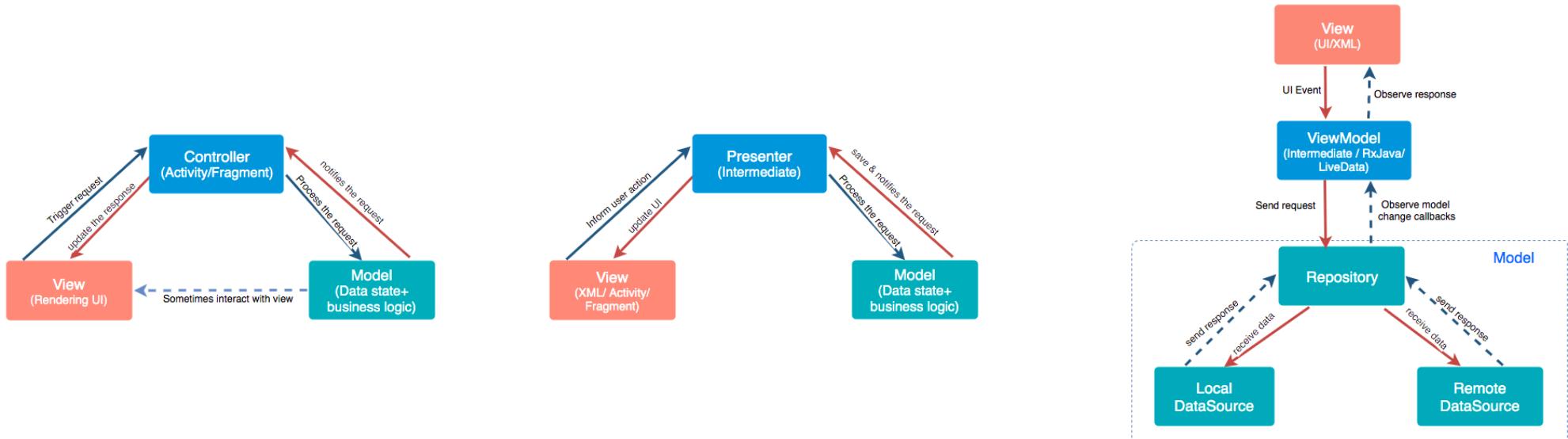
Ne postoji jasno pravilo što i kako koristiti. OVISI O ZAHTJEVIMA APLIKACIJE!

4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

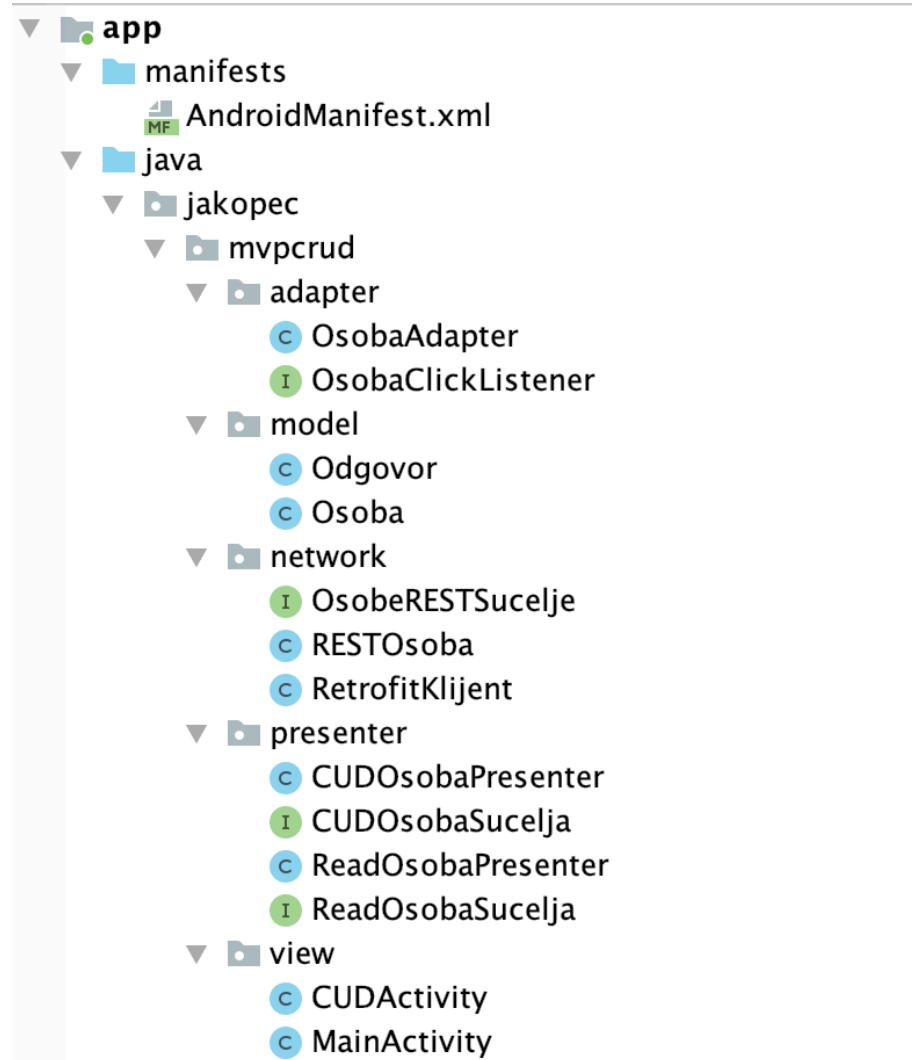
Izvor:

<https://blog.mindorks.com/mvc-mvp-mvvm-architecture-in-android>





Model View Presenter Osoba CRUD arhitektura



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



MVP predložak dizajna



OsobaClickListener.java

```
public interface OsobaClickListener {  
    void onItemClick(Osoba osoba);  
}
```

MainActivity.java

```
recyclerView.setAdapter(new OsobaAdapter(new OsobaClickListener() {  
    @Override  
    public void onItemClick(Osoba osoba) { detalji(osoba); }  
}));
```

4. Dani strukovnih
nastavnika 2020

OsobaAdapter.java

```
private OsobaClickListener osobaClickListener;  
  
public OsobaAdapter(OsobaClickListener osobaClickListener) {  
    this.osobaClickListener=osobaClickListener;  
}  
  
red.itemView.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) { osobaClickListener.onItemClick(o); }  
});
```

Razvijajmo
kompetencije
zajedno!



MVP predložak dizajna



OsobeRESTSucelje.java

```
public interface OsobeRESTSucelje {  
  
    @GET("osobe?kljuc=tjakopec")  
    Call<Odgovor> dohvatiOsobe();  
  
    @POST("osobe")  
    Call<Odgovor> dodajOsobu(@Body Osoba o);  
  
    @PUT("osobe/{id}")  
    Call<Odgovor> promjeniOsobu(@Path("id") int id, @Body Osoba o);  
  
    @DELETE("osobe/{id}")  
    Call<Odgovor> obrisiOsoba(@Path("id") int id);  
}
```

4. Dani strukovnih nastavnika 2020

Razvijajmo kompetencije zajedno!

RetrofitKlijent.java

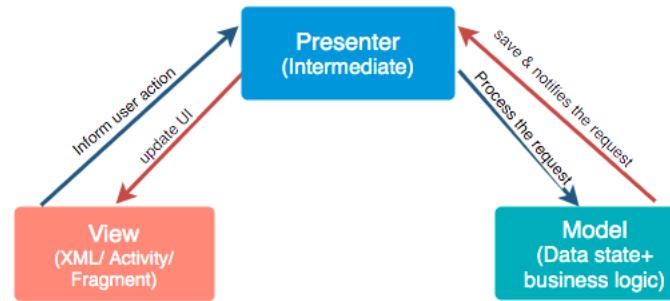
```
public class RetrofitKlijent {  
  
    public static final String REST_URL = "https://oziz.ffos.hr/DSN2020/";  
  
    private static Retrofit retrofit;  
  
    public static Retrofit getRetrofitInstance() {  
        if (retrofit == null) {  
            retrofit = new Retrofit.Builder()  
                .baseUrl(REST_URL)  
                .addConverterFactory(GsonConverterFactory.create())  
                .build();  
        }  
        return retrofit;  
    }  
}
```

RESTOsoba.java

```
OsobeRESTSucelje service;  
  
public RESTOsoba(){  
    service = RetrofitKlijent.getRetrofitInstance().create(OsobeRESTSucelje.class);  
}  
  
@Override  
public void dohvatiOsobe(ReadOsobaSucelja.Model.PoZavrsetku poZavrsetku) {...}  
  
@Override  
public void dodajOsobu(CUDOsobaSucelja.Model.PoZavrsetku poZavrsetku, Osoba osoba) {...}  
  
@Override  
public void promjeniOsobu(CUDOsobaSucelja.Model.PoZavrsetku poZavrsetku, Osoba osoba) {...}  
  
@Override  
public void obrisiOsobu(CUDOsobaSucelja.Model.PoZavrsetku poZavrsetku, Osoba osoba) {...}  
  
private void odradiPoslovniZadatak(Call<Odgovor> call, CUDOsobaSucelja.Model.PoZavrsetku poZavrsetku){...}
```



MVP predložak dizajna



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!



Izvori:

<https://handbook.infinum.co/books/android/MVP>

<https://www.grapecity.com/blogs/applying-mvp-in-android>

<https://medium.com/cr8resume/make-you-hand-dirty-with-mvp-model-view-presenter-eab5b5c16e42>

<https://medium.com/cr8resume/working-with-mvp-and-retrofit-2-in-android-b771b8369ec0>



MVP predložak dizajna



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!

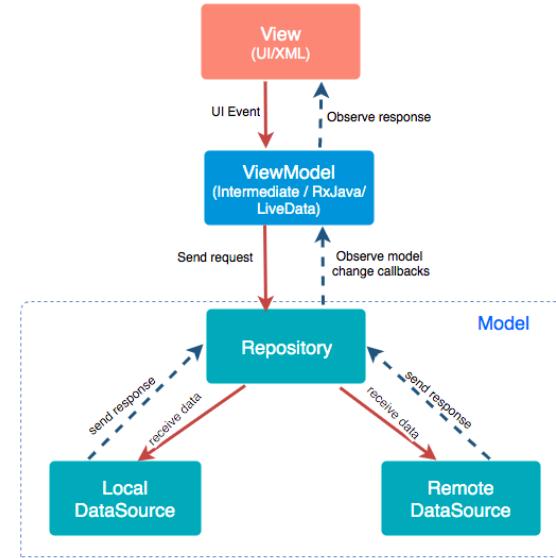




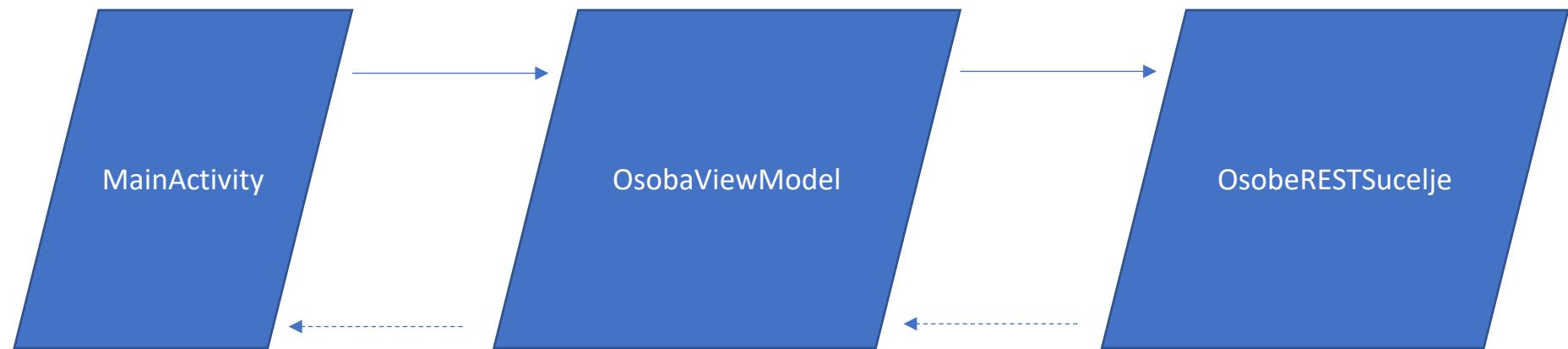
MVVM predložak dizajna



4. Dani strukovnih
nastavnika 2020



Razvijajmo
kompetencije
zajedno!





MVVM predložak dizajna



4. Dani strukovnih
nastavnika 2020

Razvijajmo
kompetencije
zajedno!





Agencija za
strukovno obrazovanje
i obrazovanje odraslih

Razvijajmo
kompetencije
zajedno!



Europska unija

"Zajedno do fondova EU"



Projekt je sufinancirala Europska unija iz Europskog socijalnog fonda.



4. Dani strukovnih nastavnika 2020
Razvoj android aplikacija kroz prizmu
različitih predložaka dizajna
Doc. dr. sc. Tomislav Jakopec

