VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI

A Mini Project Report

On

PORTFOLIO

Submitted in Partial fulfillment of requirement for the
MOBILE APPLICATION DEVELOPMENT LABORATORY WITH MINI PROJECT (18CSMP68**)**
In

Computer Science & Engineering

By

UMAKANTH N	3VC19CS173
SUNDEEP HIREMAT	3VC19CS159

Faculty In-Charge

Vinutha Prashanth
Assistant Professor,
CSE Department, RYMEC

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
ACCREDITED BY NATIONAL BOARD OF ACCREDITATION
RAO BAHADUR Y MAHABALESHWARAPPA ENGINEERING COLLEGE
CANTONMENT, BALLARI-583104, KARNATAKA

2021-22

VEERASHAIVA VIDYAVARDAHKA SANGHA'S

RAO BAHADUR Y MAHABALESHWARAPPA ENGINEERING COLLEGE

(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM & APPROVED BY AICTE, NEWDELHI)
BALLARI – 583104, KARNATAKA

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ACCREDITED BY NATIONAL BOARD OF ACCREDITATION

CERTIFICATE

This is to certify that project work entitled **"PORTFOLIO"** is a bonafied Work carried out by **UMAKANTH N(3VC19CS173), SUNDEEP HIREMAT (3VC19CS159)** of 6th Semester in Partial fulfillment of requirement for **MOBILE APPLICATION DEVELOPMENT LABORATORY WITH MINI PROJECT (**18CSMP68**)** during the year **2021-22.**

Signature of Staff - Incharge
Mrs. VINUTHA PRASHANTH
Assistant Professor,
Dept. of CSE, RYMEC

Signature of HOD
Dr. GIRISHA H
HOD, Dept. of CSE
RYMEC.

Name of Examiners: **Signature with Date**

1.

2.

ACKNOWLEDGEMENT

The satisfaction and excitement that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success.

We are highly indebted **Mrs. VINUTHA PRASHANTH** for guidance and constant supervision as well as for providing necessary information regarding the project & also for support in completing the project.

We wish to place on record our grateful thank to **Dr. H. GIRISHA**, Head of the Department, Computer Science and Engineering RYMEC, Ballari for providing encouragement and guidance.

We thank and appreciate the help of my friends in developing the project and people who have willingly helped me out with their abilities.

PROJECT ASSOCIATES

UMAKANTH N: 3VC19CS173

SUNDEEP HIREMAT: 3VC19CS159

ABSTRACT

This mini project on **PORTFOLIO** is a collection of student data that displays a brief description of two students related to personal ,academic information where one can find out the details of students

This mini project has user interactions where it takes the input from keyboard strokes or mouse. The first When Submit is pressed or clicked it will take to the next page where one can select the student they are interested to view details .

CONTENTS

CHAPTERS

1. INTRODUCTION

1.1. Introduction to Android

Android operating system is the largest installed base among various mobile platforms across the globe. Hundreds of millions of mobile devices are powered by **Android** in more than 190 countries of the world. The company named **Open Handset Alliance** developed Android for the first time that is based on the modified version of the Linux kernel and other open-source software. **Google** sponsored the project at initial stages and in the year 2005, it acquired the whole company. In September 2008, the first Android-powered device was launched in the market. Android dominates the mobile OS industry because of the long list of features it provides. It's user-friendly, has huge community support, provides a greater extent of customization, and a large number of companies build Android-compatible smart phones. As a result, the market observes a sharp increase in the demand for developing Android mobile applications, and with that companies need smart developers with the right skill set. At first, the purpose of Android was thought of as a mobile operating system. However, with the advancement of code libraries and its popularity among developers of the divergent domain, Android becomes an absolute set of software for all devices like tablets, wearables, set-top boxes, smart TVs, notebooks, etc.

Features of Android

Android is a powerful open-source operating system that open-source provides immense features and some of these are listed below.

- Android Open Source Project so we can customize the OS based on our requirements.
- Android supports different types of connectivity for GSM, CDMA, Wi-Fi, Bluetooth, etc. for telephonic conversation or data transfer.
- Using wifi technology we can pair with other devices while playing games or using other applications.
- It contains multiple APIs to support location-tracking services such as GPS.
- We can manage all data storage-related activities by using the file manager.

- It contains a wide range of media supports like AVI, MKV, FLV, MPEG4, etc. to play or record a variety of audio/video.
- It also supports different image formats like JPEG, PNG, GIF, BMP, MP3, etc.
- It supports multimedia hardware control to perform playback or recording using a camera and microphone.
- Android has an integrated open-source WebKit layout-based web browser to support User Interfaces like HTML5, and CSS3.
- Android supports multi-tasking means we can run multiple applications at a time and can switch between them.
- It provides support for virtual reality or 2D/3D Graphics.

Programming Languages used in Developing Android Applications

1. **Java**
2. **Kotlin**

Developing the Android Application using Kotlin is preferred by Google, as Kotlin is made an official language for Android Development, which is developed and maintained by JetBrains. Previously before Java is considered the official language for Android Development. Kotlin is made official for Android Development in Google I/O 2017.

Advantages of Android Development

- The Android is an open-source Operating system and hence possesses a vast community for support.
- The design of the Android Application has guidelines from Google, which becomes easier for developers to produce more intuitive user applications.
- Fragmentation gives more power to Android Applications. This means the application can run two activities on a single screen.
- Releasing the Android application in the Google play store is easier when it is compared to other platforms.

Disadvantages of Android Development

- Fragmentation provides a very intuitive approach to user experience but it has some drawbacks, where the development team needs time to adjust to the various screen sizes of mobile smart phones that are now available in the market and invoke the particular features in the application.
- The Android devices might vary broadly. So the testing of the application becomes more difficult.

- As the development and testing consume more time, the cost of the application may increase, depending on the application's complexity and features.

1.2. Installation of android studio and SDK

Installing Android software is probably the most challenging part of this project. It takes times - from 30 minutes to *n* hours to forever - depending on your luck, your programming knowledge, and your PC. You probably need a fairly decent PC (with 8GB RAM) and 10GB of free disk space to run the Android emulator!!! Running on "actual" Android phone/tablet requires much lesser resources.

Step 0: Pre-Installation Check List

1. Before installing Android SDK, you need to install Java Development Kit (JDK). Read "How to install JDK". Ensure that your JDK is at or above 1.8. You can check your JDK version with command "javac -version" (compiler) and "java -version"(runtime).
2. Uninstall older version(s) of "Android Studio" and "Android SDK", if any.
3. The installation and many operations take a LONG time to complete. Do NOT stare at your screen or at the ceiling. Browse through the "Android for Developers" @ https://developer.android.com.
4. We need to install two HUGE packages:
    a. Android Studio (IDE) (about 1.6 GB), which is an Integrated Development Environment (IDE) based on IntelliJ (a popular Java IDE); and
    b. Android SDK (Software Development Kit) (about 5 GB) for developing and running Android apps.

Step 1: Install "Android Studio IDE"

Reference: "Install Android Studio" @ https://developer.android.com/studio/install.

(For Windows)

1. Check that environment variable JAVA_HOME is set to the JDK installation directory via command "set JAVA_HOME".
2. Check the system requirements for Android Studio/SDK - For Windows 10, 8GB of RAM, 8GB of disk space, and 1280x800 minimum screen resolution. Take note that you should have enough space on C drive. Insufficient space on C drive will take you many days.

3. Goto "Android Studio" under "Android Developers" @ https://developer.android.com/studio ⇒ Click "Download Android Studio" (Android Studio Bumblebee 2021.x.x for Windows 64-bit (872MiB)) to download the executable installer "android-studio-2021.x.x.xx-windows.exe".
4. Run the downloaded installer ⇒
    a. In "Choose Components", select "Android Studio" and "Android Virtual Device" (space required: 2.7GB).
    b. In "Configuration Settings Install Location", accept the default "C:\Program Files\Android\Android Studio".
    c. In "Choose Start Menu Folder", accept the default ⇒ Install.
    d. Launch Android Studio. Continue to Step 2.

The "Android Studio IDE" will be installed in "C:\Program Files\Android\Android Studio", and the "Android SDK" in "c:\Users*username*\AppData\Local\Android\Sdk".

(For macOS)

1. Check the system requirements - macOS 10.14 or higher, 8GB of RAM, 8GB of disk space, and 1280x800 minimum screen resolution.
2. Goto "Android Developer" under "Android Developers" @ https://developer.android.com/index.html ⇒ Click "Download Options" ⇒ For **Intel** processor, choose "Mac (64-bit) Android-studio-2021.x.x.xx-mac.dmg (928MiB)"; for **Apple M1 ARM** processor, choose "Mac (64-bit, ARM) Android-studio-2021.x.x.x-mac_arm.dmg (925MiB)".
3. Launch the downloaded ".dmg" installation file ⇒ You may watch a short video @ https://developer.android.com/studio/install.
4. Drag and drop Android Studio into the "Applications" folder. Continue to Step 2.

The "Android SDK" will be installed in "~/Library/Android/sdk", where ~ denotes your home directory.

Note: If you see a warning that says "the package is damaged and should be moved to the trash", goto "System Preferences" ⇒ Security & Privacy ⇒ under "Allow applications downloaded from" ⇒ select "Anywhere". Then run again.

Step 2: Installing Android SDK

This step takes a long time as you need to download about 3GB of zip data, and expand to 5 GB of disk data, even for the minimum configuration.

Note: You can actually copy the SDK from another computer with the same OS.

(For Windows and macOS)

1. Launch Android Studio ⇒ It will run the "setup" wizard for the first launch.
   a. Choose "do not import previous settings", and wait...
   b. In "Welcome", choose "next".
   c. In "Install Type", choose "Standard" (default).
   d. In "Select UI Theme", choose one that you like (or default).
   e. In "Verify Settings", take note of the SDK directory (by default @ c:\Users*username*\AppData\Local\Android\Sdk for Windows, "~/Library/Android/sdk" for macOS):
   f. In "License Agreement", Accept ALL items ⇒ Finish ⇒ Wait (this step will take quite some time)
2. (For Windows) Use "File Explorer" to check the SDK installed directory. Take note that the "AppData" is a hidden directory. You need to choose "View" ⇒ Uncheck "Hidden Items" to see this directory.
   (For macOS) Use "Finder" to check the SDK installed directory.
3. You can also use "Android Studio" to check the SDK packages installed by selecting "More Actions" (or "Configure" in older versions, or "Tools") ⇒ "SDK Manager" ⇒ "Android SDK" (sidebar):
   - Under "SDK Platforms" tab:
     - Android API 32
   - Under "SDK Tools" tab:
     - Android SDK Build Tools 33-rc1
     - Android Emulator
     - Android SDK Platform-Tools (33.0.0)
     - Intel x86 Emulator Accelerator (HAXM installer)

2. SYSTEM REQUIREMENTS

2.1. Software requirements

- **Operating system:** Windows 10 or higher
- **Integrated Development Environment:** Android Studio (Latest Version)
- **Development Kits:** Java JDK and Android SDK

2.2.Hardware requirements

- 64-bit Microsoft® Windows® 8/10/11
- x86_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor
- 8 GB RAM or more
- 8 GB of available disk space minimum (IDE + Android SDK + Android Emulator)
- 1280 x 800 minimum screen resolution

3. ABOUT THE PROJECT

3.1. INTRODUCTION TO THE PROJECT

Portfolio is a collection of student work that can demonstrate learning and be used as an effective assement tool. The portfolio can range from simple collection of teacher identified student assignments to more complex student-driven product demonstrating learning standards with teacher evaluating. A portfolio can be completed as short term project or a comprehensive one that spans over the year.

4. DESIGN AND IMPLEMENTATION

Source Code:

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView2"
        android:layout_alignParentStart="true"
        android:layout_marginStart="42dp"
        android:layout_marginBottom="15dp"
        android:fontFamily="@font/alegreya"
        android:text="@string/associ"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textColor="#F45B50"
        android:textSize="20sp" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="320dp"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView5"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:layout_marginStart="51dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="36dp"
        android:fontFamily="@font/bigshot_one"
        android:text="@string/title"
        android:textAlignment="center"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textColor="#E91E63"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="336dp"
        android:layout_height="wrap_content"
        android:layout_above="@+id/hmbtn"
        android:layout_alignParentStart="true"
        android:layout_marginStart="40dp"
```

```
        android:layout_marginBottom="102dp"
        android:fontFamily="@font/acme"
        android:text="@string/std2"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textColor="#4CAF50"
        android:textSize="16sp" />

    <ImageView
        android:id="@+id/appnam"
        android:layout_width="102dp"
        android:layout_height="102dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentTop="true"
        android:layout_marginStart="25dp"
        android:layout_marginTop="29dp"
        android:contentDescription="@string/app_name"
        android:scaleType="fitXY"
        android:src="@drawable/rymec"
        tools:ignore="ImageContrastCheck"
        tools:layout_editor_absoluteX="64dp"
        tools:layout_editor_absoluteY="134dp" />

    <TextView
        android:id="@+id/textView"

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView4"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:layout_marginStart="68dp"
        android:layout_marginEnd="68dp"
        android:layout_marginBottom="64dp"
        android:fontFamily="@font/average"
        android:text="@string/clgname"
        android:textAlignment="center"
        android:textAppearance="@style/Theme.AppCompat"
        android:textColor="#C62BE1"
        android:textSize="48sp"
        android:textStyle="bold" />

    <ImageView
        android:id="@+id/rymeclogo"
        android:layout_width="93dp"
        android:layout_height="97dp"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true"
        android:layout_marginTop="33dp"
        android:layout_marginEnd="22dp"
        android:contentDescription="@string/app_name"
        android:scaleType="fitXY"

        android:src="@drawable/vtu"
```

```
        tools:layout_editor_absoluteX="64dp"
        tools:layout_editor_absoluteY="134dp"
        tools:srcCompat="@tools:sample/avatars" />


    <Button
        android:id="@+id/hmbtn"
        android:layout_width="124dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="142dp"
        android:layout_marginEnd="144dp"
        android:layout_marginBottom="67dp"

        android:text="@string/btnname" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="283dp"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView3"
        android:layout_alignParentStart="true"
        android:layout_marginStart="42dp"
        android:layout_marginBottom="22dp"
        android:fontFamily="@font/acme"
        android:text="@string/std1"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textColor="#4CAF50"
        android:textSize="16sp" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    private Button button;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);


        button = findViewById(R.id.hmbtn);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openPage2();
            }
        });

    }
    public void  openPage2(){
        Intent intent=new Intent(this,MainActivity2.class);
        startActivity(intent);
    }
}
```

activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity2"
    >

    <com.google.android.material.tabs.TabLayout
        android:id="@+id/tab"
        android:layout_width="413dp"
        android:layout_height="50dp"
        android:background="#92E8E0"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.489"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023">

        <com.google.android.material.tabs.TabItem
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/tab1" />

        <com.google.android.material.tabs.TabItem
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
        android:text="@string/tab2" />


    </com.google.android.material.tabs.TabLayout>

    <androidx.viewpager.widget.ViewPager
        android:id="@+id/view"
        android:layout_width="match_parent"
        android:layout_below="@+id/tab"
        android:layout_height="match_parent">


    </androidx.viewpager.widget.ViewPager>




</LinearLayout>
```

MainActivity2.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import android.os.Bundle;

import com.google.android.material.tabs.TabLayout;

public class MainActivity2 extends AppCompatActivity {
    TabLayout tab;
    ViewPager viewPager;


    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        tab=findViewById(R.id.tab); //tab id
        viewPager=findViewById(R.id.view); // viewer screen id

        Page_Adder adapter = new Page_Adder(getSupportFragmentManager());
        viewPager.setAdapter(adapter);

        tab.setupWithViewPager(viewPager);

    }
}
```

fragment_view_std1.xml

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import android.os.Bundle;

import com.google.android.material.tabs.TabLayout;

public class MainActivity2 extends AppCompatActivity {
    TabLayout tab;
    ViewPager viewPager;


    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        tab=findViewById(R.id.tab); //tab id
        viewPager=findViewById(R.id.view); // viewer screen id

        Page_Adder adapter = new Page_Adder(getSupportFragmentManager());
        viewPager.setAdapter(adapter);

        tab.setupWithViewPager(viewPager);

    }
}
```

ViewStd1.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import android.os.Bundle;

import com.google.android.material.tabs.TabLayout;

public class MainActivity2 extends AppCompatActivity {
    TabLayout tab;
    ViewPager viewPager;


    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
```

```
        tab=findViewById(R.id.tab); //tab id
        viewPager=findViewById(R.id.view); // viewer screen id

        Page_Adder adapter = new Page_Adder(getSupportFragmentManager());
        viewPager.setAdapter(adapter);

        tab.setupWithViewPager(viewPager);

    }
}
```

fragment_view_std2.xml

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import android.os.Bundle;

import com.google.android.material.tabs.TabLayout;

public class MainActivity2 extends AppCompatActivity {
    TabLayout tab;
    ViewPager viewPager;


    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        tab=findViewById(R.id.tab); //tab id
        viewPager=findViewById(R.id.view); // viewer screen id

        Page_Adder adapter = new Page_Adder(getSupportFragmentManager());
        viewPager.setAdapter(adapter);

        tab.setupWithViewPager(viewPager);

    }
}
```

ViewStd2.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;
```

```
import androidx.viewpager.widget.ViewPager;

import android.os.Bundle;

import com.google.android.material.tabs.TabLayout;

public class MainActivity2 extends AppCompatActivity {
    TabLayout tab;
    ViewPager viewPager;


    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        tab=findViewById(R.id.tab); //tab id
        viewPager=findViewById(R.id.view); // viewer screen id

        Page_Adder adapter = new Page_Adder(getSupportFragmentManager());
        viewPager.setAdapter(adapter);

        tab.setupWithViewPager(viewPager);

    }
}
```

Page_Adder.java

```
package com.example.mad_project;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentPagerAdapter;

public class Page_Adder extends FragmentPagerAdapter {
    public Page_Adder(@NonNull FragmentManager fm) {
        super(fm);
    }



    @Override
    public Fragment getItem(int position) {
      if (position==0){
          return new ViewStd1();
      }else{
          return new ViewStd2();
        }

    }
```

```
    @Override
    public int getCount() {
        return 2;
    }

    @Override
    public CharSequence getPageTitle(int position) {
        if (position==0){
            return "Student1";
        }else {
            return "Student2";
        }
    }

}
```

activity_pdf.xml

```
package com.example.mad_project;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentPagerAdapter;

public class Page_Adder extends FragmentPagerAdapter {
    public Page_Adder(@NonNull FragmentManager fm) {
        super(fm);
    }



    @Override
    public Fragment getItem(int position) {
      if (position==0){
          return new ViewStd1();
      }else{
          return new ViewStd2();
        }

    }

    @Override
    public int getCount() {
        return 2;
    }

    @Override
    public CharSequence getPageTitle(int position) {
        if (position==0){
            return "Student1";
```

```
        }else {
            return "Student2";
        }
    }

}
```

PdfActivity.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import com.github.barteksc.pdfviewer.PDFView;

public class PdfActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pdf);

        PDFView pdfView=findViewById(R.id.pdf);
        pdfView.fromAsset("uma.pdf").load();


    }
}
```

activity_pdf2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".PdfActivity2">

    <com.github.barteksc.pdfviewer.PDFView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/pdf2"
        />
```

```
</RelativeLayout>
```

PdfActivity2.java

```
package com.example.mad_project;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import com.github.barteksc.pdfviewer.PDFView;

public class PdfActivity2 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pdf2);


            PDFView pdfView2=findViewById(R.id.pdf2);
            pdfView2.fromAsset("sun.pdf").load();

        }
    }
```

5. SNAPSHOTS

FIG: Display page

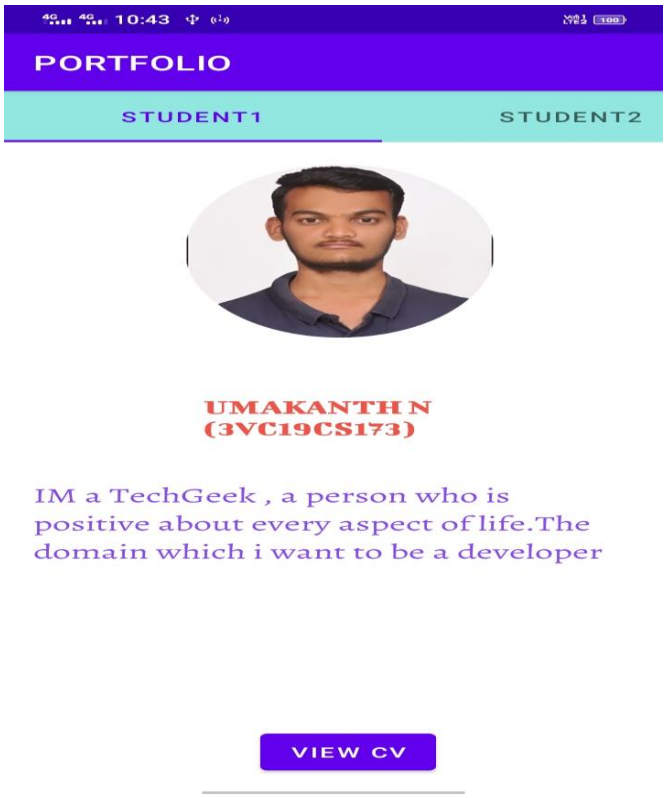

FIG: Selection of student page

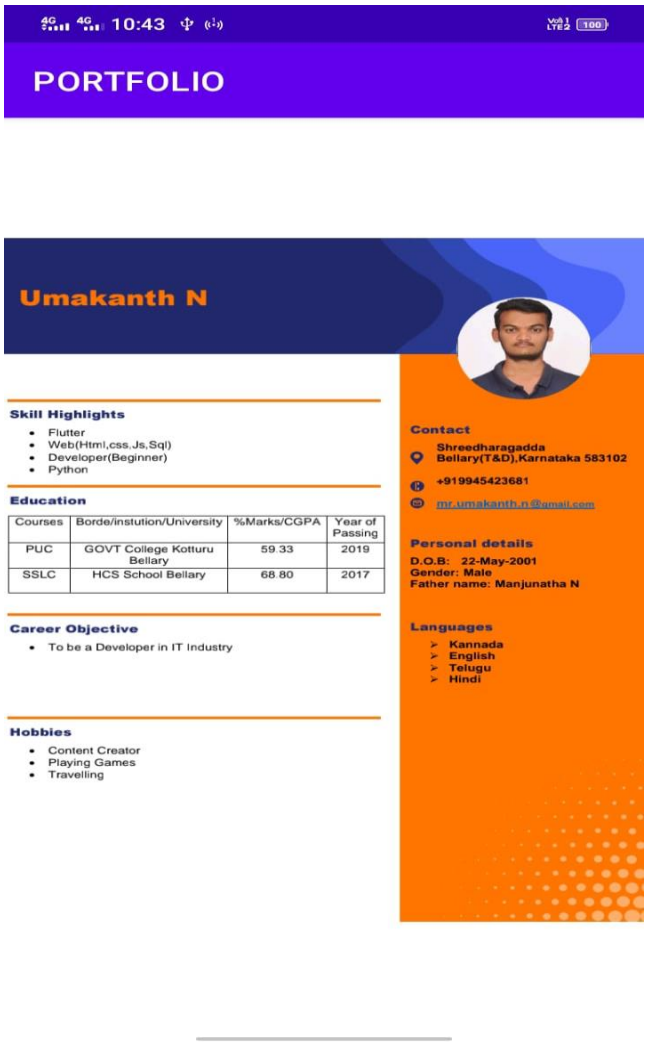

FIG: Student 1 CV

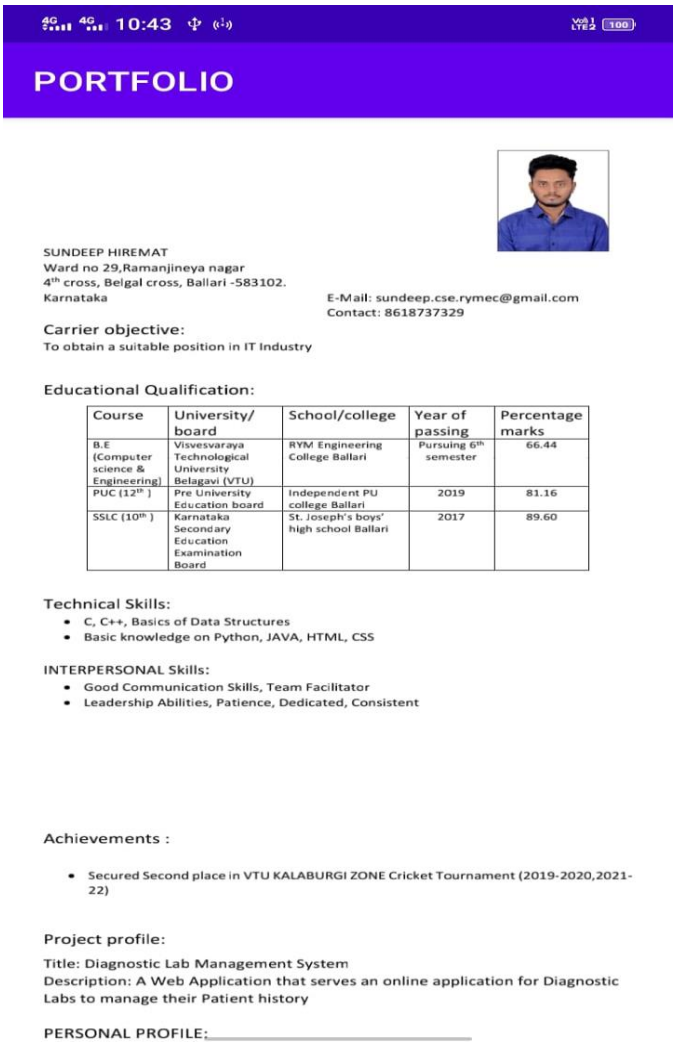

FIG: Student 2 CV

6. CONCLUSION

Portfolio is a very valuable and meaningful evaluation tool that effectively assesses student leaning. Along with the student reflection the data provides the valuable information about each student learns and what is important to him or her in the learning process

7. FUTURE ENHANCEMETS

Our future enhancement is this project is The portfolios have seen an unprecedented surge in popularity, they have also become the subject of controversy: learners often perceive little gain from writing reflections as part of their portfolios; scholars question the ethics of such obligatory reflection; and students, residents, teachers and scholars alike condemn the bureaucracy surrounding portfolio implementation in competency-based education. It could be argued that mass adoption without careful attention to purpose and format may well jeopardize portfolios' viability in health sciences education. Moreover, comprehensive portfolios, which are integrated into the curriculum and much more diverse in content than reflective portfolios, can serve as meaningful patient charts, providing doctor and patient with useful information to discuss well-being and treatment. In this sense, portfolios are also learner charts that comprehensively document progress in a learning trajectory which is lubricated by meaningful dialogue between learner and mentor in a trusting relationship to foster learning. If we are able to make such comprehensive and meaningful use of portfolios, then, yes, portfolios do have a bright future in education, business.

8. BIBLOGRAPHY

8.1. Text book references

1. Erik Hellman, **"Android Programming – Pushing the Limits",** 1st Edition, Wiley India Pvt Ltd,
2014. ISBN-13: 978-8126547197
2. Dawn Griffiths and David Griffiths, **"Head First Android Development",** 1st Edition, O'Reilly
SPD Publishers, 2015. ISBN-13: 978-9352131341
3. Bill Phillips, Chris Stewart and Kristin Marsicano, **"Android Programming: The Big NerdRanch Guide",** 3rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13: 978-0134706054

8.2. Web references

https://developer.android.com/studio/install

https://www3.ntu.edu.sg/home/ehchua/programming/android/Android_HowTo.html

https://github.com

http://www.geeksforgeeks.org