VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JNANA SANGAMA", Belagavi-590018

A Mini Project Report on

"WEBVIEW PDF CONVERTER APPLICATION"

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR

THE AWARD OF DEGREE OF

BACHELOR OF ENGINEERING IN
INFORMATION SCIENCE AND ENGINEERING

SUBMITTED BY

NISHANTH N GOWTHAM(1JB20IS039)

ROHITH HANDE H (1JB21IS401)

Under the Guidance of

Dr. PAVITHRA BAI S
Assistant Professor,
Dept. of ISE, SJBIT
Bengaluru-60

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SJB INSTITUTE OF TECHNOLOGY

BGS HEALTH AND EDUCATION CITY, KENGERI, BENGALURU-560060, KARNATAKA, INDIA.

2022-2023

|| Jai Sri Gurudev ||

Sri Adichunchanagiri Shikshana Trust ®

SJB INSTITUTE OF TECHNOLOGY

BGS Health & Education City, Kengeri, Bengaluru – 560 060

Department of Information Science & Engineering

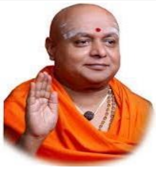
CERTIFICATE

Certified that the Mini-project work entitled "**WEBVIEW PDF CONVERTER APP**"**,** is bonafide work carried out by **NISHANTH N GOWTHAM** (**1JB20IS039) and ROHITH HANDE H (1JB21IS401),** bonafide students of **SJB Institute of Technology,** in partial fulfilment for 6th semester in **INFORMATION SCIENCE AND ENGINEERING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY**, **BELAGAVI** during the academic year **2022-23.** It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report isdeposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said degree.

Dr. Pavithra Bai **Associate Professor** **Dept. of ISE, SJBIT**	**Dr. Shashidhara H R** **Professor & Head** **Dept. of ISE, SJBIT**

EXTERNAL VIVA

Name of the Examiners	**Signature with Date**
1. ____________________	____________________
2. ____________________	____________________



ACKNOWLEDGEMENT

I would like to express my profound thanks to His Divine Soul **Padmabhushan Sri Sri Sri Dr. Balagangadharanatha Maha Swamiji** and His Holiness **Jagadguru Sri Sri Sri Dr. Nirmalanandanatha Maha Swamiji** for providing me an opportunity to pursue my academics in this esteemed institution.

I would also like to express my profound thanks to **Revered Sri Sri Dr. Prakashnath Swamiji**, Managing Director, SJB Institute of Technology, for his continuous support in providing amenities to carry out this mini project in this admired institution.

I express my gratitude to **Dr. K. V. Mahendra Prashanth**, Principal, SJB Institute of Technology, for providing excellent facilities and academic ambience, which have helped me in satisfactory completion of mini project work.

I extend my sincere thanks to **Dr. Shashidhara H.R**, Head of the Department, Information Science and Engineering, for providing an invaluable support throughout the period of mini project work.

I wish to express heartfelt gratitude to my guide**, Dr. Pavithra Bai S,** Assistant Professor, for his valuable guidance, suggestions and cheerful encouragement during the entire period of this work.

Finally, I take this opportunity to extend my earnest gratitude and respect to my parents, Teaching & Non-teaching staffs of the department, the library staff and all my friends, for their continuous support and encouragement.

NISHANTH N GOWTHAM
(1JB20IS039)
ROHITH HANDE H
(1JB21IS401)

ABSTRACT

This project aims to develop an Android app that enables users to convert webpages into PDF files. The app provides a convenient and efficient way to capture online content and save it as a portable document format. With the increasing importance of digital content, this app fills a crucial gap in the market by allowing users to transform webpages into easily shareable and printable documents. The app will leverage web scraping techniques to extract the webpage's content, including text, images, and formatting. It will provide users with a user-friendly interface where they can enter the URL of the webpage they want to convert. The app will then retrieve the webpage's data and present it in a preview format, allowing users to review the content before generating the PDF file. To ensure versatility and customization, the app will offer various options for users to customize the PDF output. Users will have the ability to select specific sections of the webpage they want to include, adjust the layout and formatting, and choose between different PDF settings such as page size and orientation. Furthermore, the app will include features like offline mode, where users can save webpages for later conversion even without an internet connection. It will also support batch processing, enabling users to convert multiple webpages into PDF files simultaneously. Overall, this Android app project to convert webpages into PDF files addresses the growing need for efficient content capture and storage. By providing a simple yet powerful solution, it empowers users to transform online content into easily accessible and shareable PDF documents.

TABLE OF CONTENT

Chapter

LIST OF FIGURES

LIST OF SNAPSHOTS

Chapter 1

INTRODUCTION

1.1 Objective

Sometimes there is a need to save some articles available on the internet in the form of PDF files. And to do so there are many ways, one can use any browser extension or any software or any website to do so. But in order to implement this feature in the android app, one can't rely on other software or websites to do so. So to implement this amazing feature in the android app

1.2 Importance and scope

Sometimes there is a need to save some articles available on the internet in the form of PDF files. And to do so there are many ways, one can use any browser extension or any software or any website to do so. But in order to implement this feature in the android app, one can't rely on other software or websites to do so. The app can help the user to save their Internet webview information in the pdf format.

1.3 Android Studio

Android Studio is an integrated development environment (IDE) for developing for the Android platform. It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0.

Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. Based on JetBrains' IntelliJ IDEA software, Android Studio is designed specifically for Android development. It is available for download on Windows, Mac OS X and Linux, and replaced Eclipse Android Development Tools (ADT) as Google's primary IDE for native Android application development.

1.4 Android Architecture

We studied the Android system architecture. Android system is a Linux-based system, Use of the software stack architecture design patterns. The Android architecture consists of four layers: Linux kernel, Libraries and Android runtime, Application framework and

Applications. Each layer of the lower encapsulation, while providing call interface to the upper.

1.5 Methodology

This project is made by using Android studio, Virtual emulator and Photoshop. The programming languages used for building the application are Java and XML .User interface is handled using XML codes. Backend programming is handled mainly through set of java codes.

The other libraries required are:

- Android SDK tools
- ARM EABI v7a System Image
- Intel x86 Atom_64 System Image
- Google APIs
- Google APIs ARM EABI v7a System Image
- GPU Debugging tools
- Intel x86 Emulator Accelerator (HAXM installer)
- Google USB drivers
- Google play services
- Android SDK platform tools
- Android SDK build tools
- SDK platform
- Android Support Library
- Google Web drivers

1.6 Project structure

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view. This view is organized by modules to provide quick access to your project's key source files. All the build files are visible at the top level under Grade Scripts and each app module contains the following folders:

- Manifests: Contains the AndroidManifest.xml file.
- Java: Contains the Java source code files, including JUnit test code.

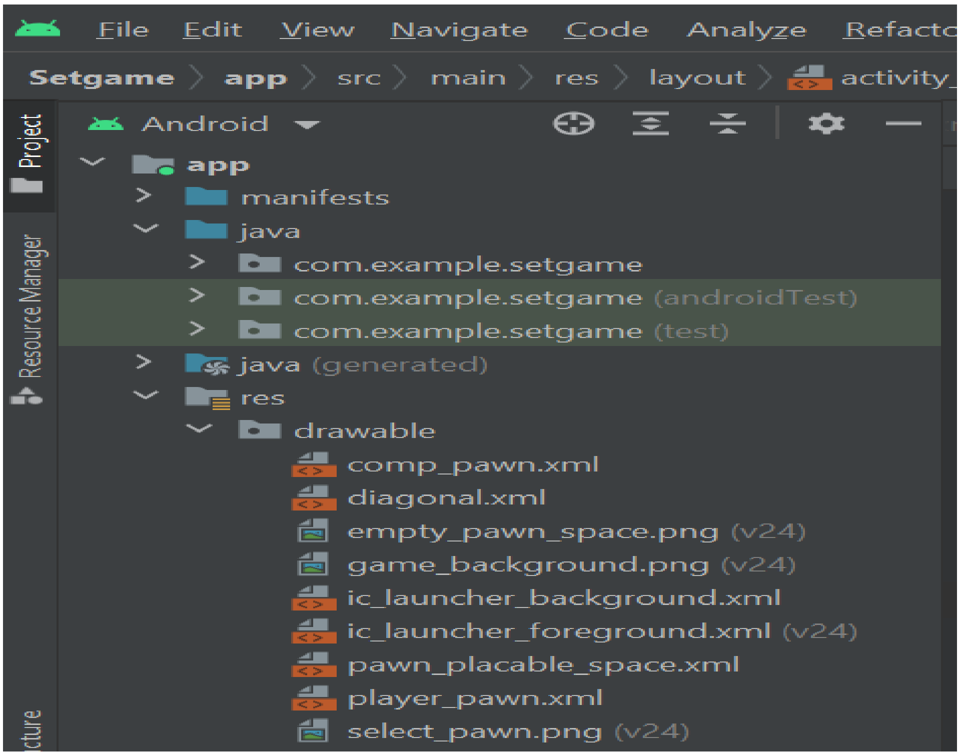


Fig 1.1: The project files in android view

Chapter 2

SYSTEM REQUIREMENT

2.1 INTRODUCTION

Requirements are during early stages of a system development as a specification of what should be implemented or as a constraint of some kind of on the system. They may be a user level facility description, a detailed specification of expected system behavior, a general system property, a specific constraint on the system, and information on how to carry out some computation or a constraint on the development of the system. The end product of the requirement analysis phase is a requirement specification. The requirement specification is a reconstruction of the result of this analysis phase.

Natural language is often used to write system requirements specifications. Further problems with natural language can arise when it is used for more detailed specification:

- Natural language understanding relies on the specification of the readers and writers using the same words for the same concept. This leads to misunderstandings because of the ambiguity of the natural language.
- A natural language requirements specification is over-flexible. You can say the same thing in completely different ways. It is up to the reader to find out when requirements are same and when they are distinct.

2.2 Functional Requirement

The functional requirements are the statement of services the system should provide, how system reacts to particular inputs and how system should behave in particular situation. It describes the functionality that the system provides. Our app requires:

- Username to provide a their name.

2.3 Android Studio

- 64-bit Microsoft Windows 8/10
- 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
- 1280 x 800 minimum screen resolution.

The process of building applications and games is resource-intensive, especially CPU. If your CPU is too old then your computer will be very hot and possibly overloaded. Therefore, we recommend using an Intel Core i5-8400 or a better CPU. In addition, the use of SSD also makes your work process much quicker and more comfortable than using a HDD.

2.4 Android Studio Emulator

Android Studio is the default development platform for Android application. It comes with collections of tools that help developers to make their apps and games specifically for Android devices. It also provides a built-in emulator which is used to test your app takes a few minutes to start.

Minimum system requirements

- Windows, Linux or Mac
- 64-bit distribution capable of running 32-bit applications
- 3 GB GAM • 2 GB hard disk space
- Minimum resolution needs to be 1280 x 800

2.5 Java System requirement

- Windows 10 (8u51 and above)
- Windows 8.x (Desktop) • Windows 7 SP1
- Windows Vista SP2
- Windows Server 2008 R2 SP1 (64-bit)
- Windows Server 2012 and 2012 R2 (64-bit)
- RAM: 128 MB
- Disk space: 124 MB for JRE; 2 MB for Java Update
- Processor: Minimum Pentium 2 266 MHz processor

Chapter 3

SYSTEM DESIGN

3.1 Date flow diagram

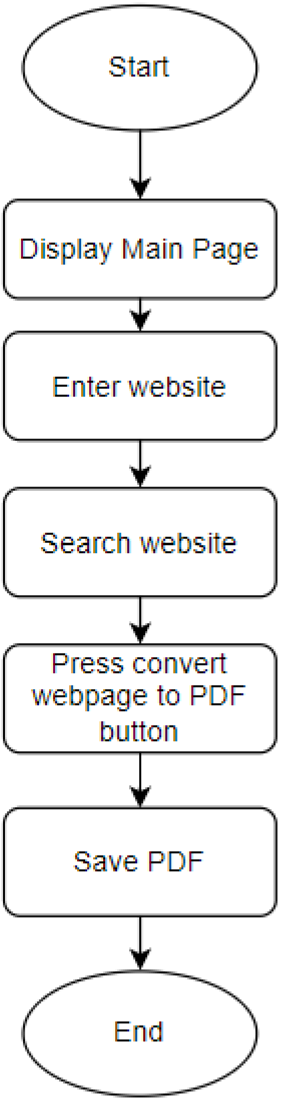


Fig 3.1: Data flow diagram

Chapter 4

IMPLENTATION

4.1. Java

There are several ways to create apps for Android devices, but their commended method for most developers is to write native apps using Java and the Android SDK. Java for Android apps is both similar and quite different from other types of Java applications. If you have experience with Java (or a similar language) then you'll probably feel comfortable diving right into the code and learning how to use the Android SDK to make your app run. But if you're new to programming or object- oriented languages then you'll probably want to get familiar with the syntax of the Java language and how to accomplish basic programming tasks before learning how to use the Android SDK.

4.2 XML Convert Web View To PDF Page Implementation

```
android:textColor="#fff"
android:textSize="10sp"
android:textStyle="bold" />
</LinearLayout>
<EditText
android:id="@+id/etoutput"
android:layout_width="359dp"
android:layout_height="180dp"
android:layout_below="@+id/llout"
android:layout_marginTop="63dp"
android:alpha="0.8"
android:background="#ffffff"
android:gravity="start"
android:hint="output here"
android:padding="10sp"
android:textSize="20sp"
android:textStyle="bold" />
</RelativeLayout>
```

4.3 Convert Web View To PDF Java Implementation

Main Activity.java

```
import android.content.Context;

import android.os.Build;

import android.os.Bundle;

import android.print.PrintAttributes;

import android.print.PrintDocumentAdapter;

import android.print.PrintJob;

import android.print.PrintManager;

import android.view.View;

import android.webkit.WebView;

import android.webkit.WebViewClient;

import android.widget.Button;

import android.widget.Toast;

import androidx.annotation.RequiresApi;

import androidx.appcompat.app.AppCompatActivity;

  public class MainActivity extends AppCompatActivity {

   // creating object of WebView

  WebView printWeb;

  @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

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

```
// Initializing the WebView
  final WebView webView = (WebView) findViewById(R.id.webViewMain);
Initializing the Button
   Button savePdfBtn = (Button) findViewById(R.id.savePdfBtn);
 / Setting we View Client
  webView.setWebViewClient(new WebViewClient() {
      @Override
      public void onPageFinished(WebView view, String url) {
          super.onPageFinished(view, url);
          // initializing the printWeb Object
          printWeb = webView;
      }
  });
   // loading the URL
  webView.loadUrl("https://www.google.com");
 // setting clickListener for Save Pdf Button
  savePdfBtn.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
          if (printWeb != null) {
              if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
                  // Calling createWebPrintJob()
                  PrintTheWebPage(printWeb);
```

```
                    } else {

                        // Showing Toast message to user

                        Toast.makeText(MainActivity.this, "Not available for device below Android LOLLIPOP", Toast.LENGTH_SHORT).show();

                    }

                } else {

                    // Showing Toast message to user

                    Toast.makeText(MainActivity.this, "WebPage not fully loaded", Toast.LENGTH_SHORT).show();

                }

            }

        });

    }

   // object of print job

    PrintJob printJob;

  // a boolean to check the status of printing

    boolean printBtnPressed = false;

    @RequiresApi(api = Build.VERSION_CODES.LOLLIPOP)

    private void PrintTheWebPage(WebView webView) {

                  // set printBtnPressed true

        printBtnPressed = true;

       // Creating  PrintManager instance

        PrintManager printManager = (PrintManager) this

                .getSystemService(Context.PRINT_SERVICE);
       // setting the name of job
        String jobName = getString(R.string.app_name) + " webpage" + webView.getUrl();
       // Creating  PrintDocumentAdapter instance
        PrintDocumentAdapter printAdapter =
webView.createPrintDocumentAdapter(jobName);
       // Create a print job with name and adapter instance
        assert printManager != null;
        printJob = printManager.print(jobName, printAdapter,
              new PrintAttributes.Builder().build());
    }
  @Override
   protected void onResume() {
        super.onResume();
        if (printJob != null && printBtnPressed) {
            if (printJob.isCompleted()) {
                // Showing Toast Message
                Toast.makeText(this, "Completed", Toast.LENGTH_SHORT).show();
            } else if (printJob.isStarted()) {
                // Showing Toast Message
                Toast.makeText(this, "isStarted", Toast.LENGTH_SHORT).show();
                               } else if (printJob.isBlocked()) {
                // Showing Toast Message
                Toast.makeText(this, "isBlocked", Toast.LENGTH_SHORT).show();
           } else if (printJob.isCancelled()) {

                // Showing Toast Message

               Toast.makeText(this, "isCancelled", Toast.LENGTH_SHORT).show();

           } else if (printJob.isFailed()) {

                // Showing Toast Message

                Toast.makeText(this, "isFailed", Toast.LENGTH_SHORT).show();

            } else if (printJob.isQueued()) {

                // Showing Toast Message

               Toast.makeText(this, "isQueued", Toast.LENGTH_SHORT).show();

            }

            // set printBtnPressed false

            printBtnPressed = false;

        }

    }

}
```

Chapter 5

TEST CASES

5.1 Test Cases Table

Test Case Id	Test Scenario	Test Case	Test Data	Expected Result	Post Condition	Actual Result	Status (Pass/Fail)
TC_1	Entering the URL	Search for website	URL	Should open the website specified by URL	Destination website	As expected	Pass
TC_2	Clicking Convert button	Click on convert website to PDF button	User click	Should open print page	Renders print activity view	As expected	Pass
TC_3	On clicking save button	PDF	User click	Should open file manager in device	Renders file manager	As expected	Pass

Chapter 6

RESULTS

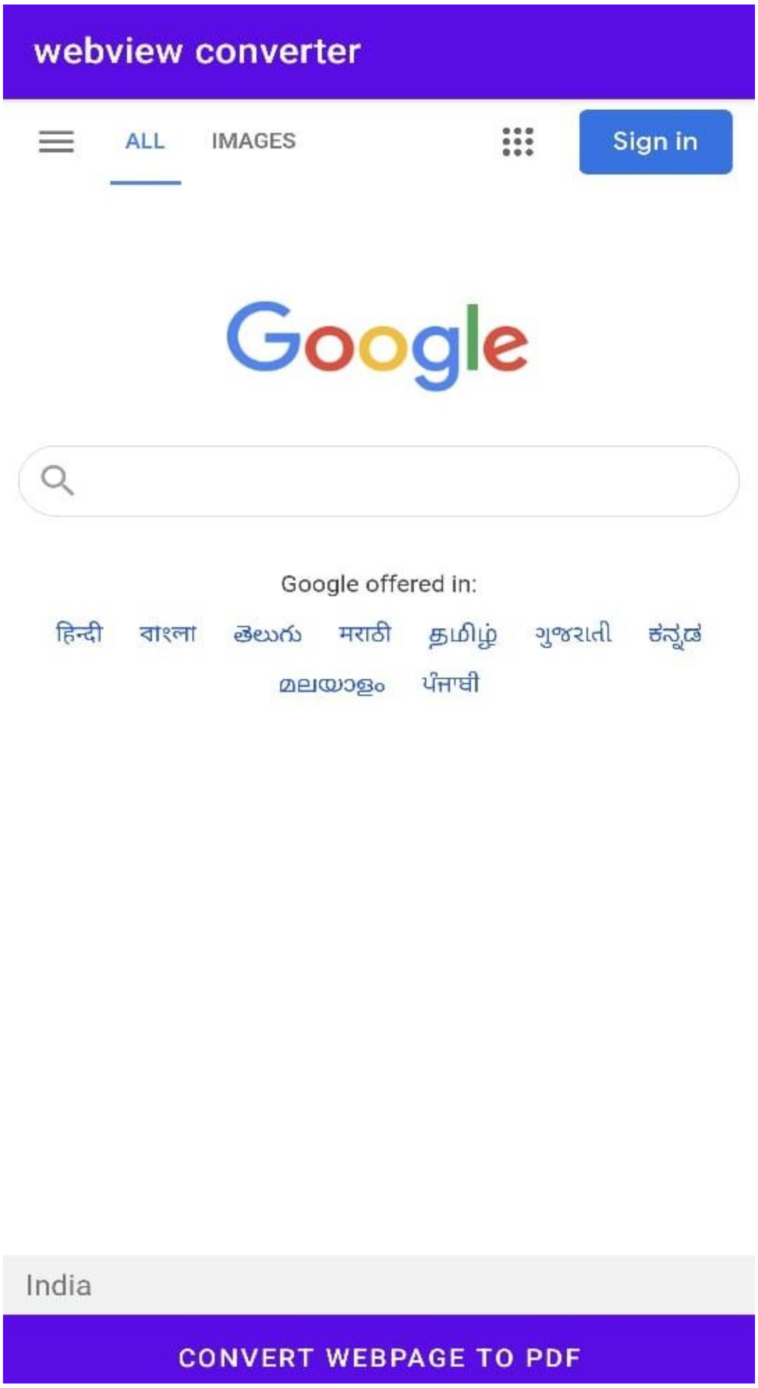
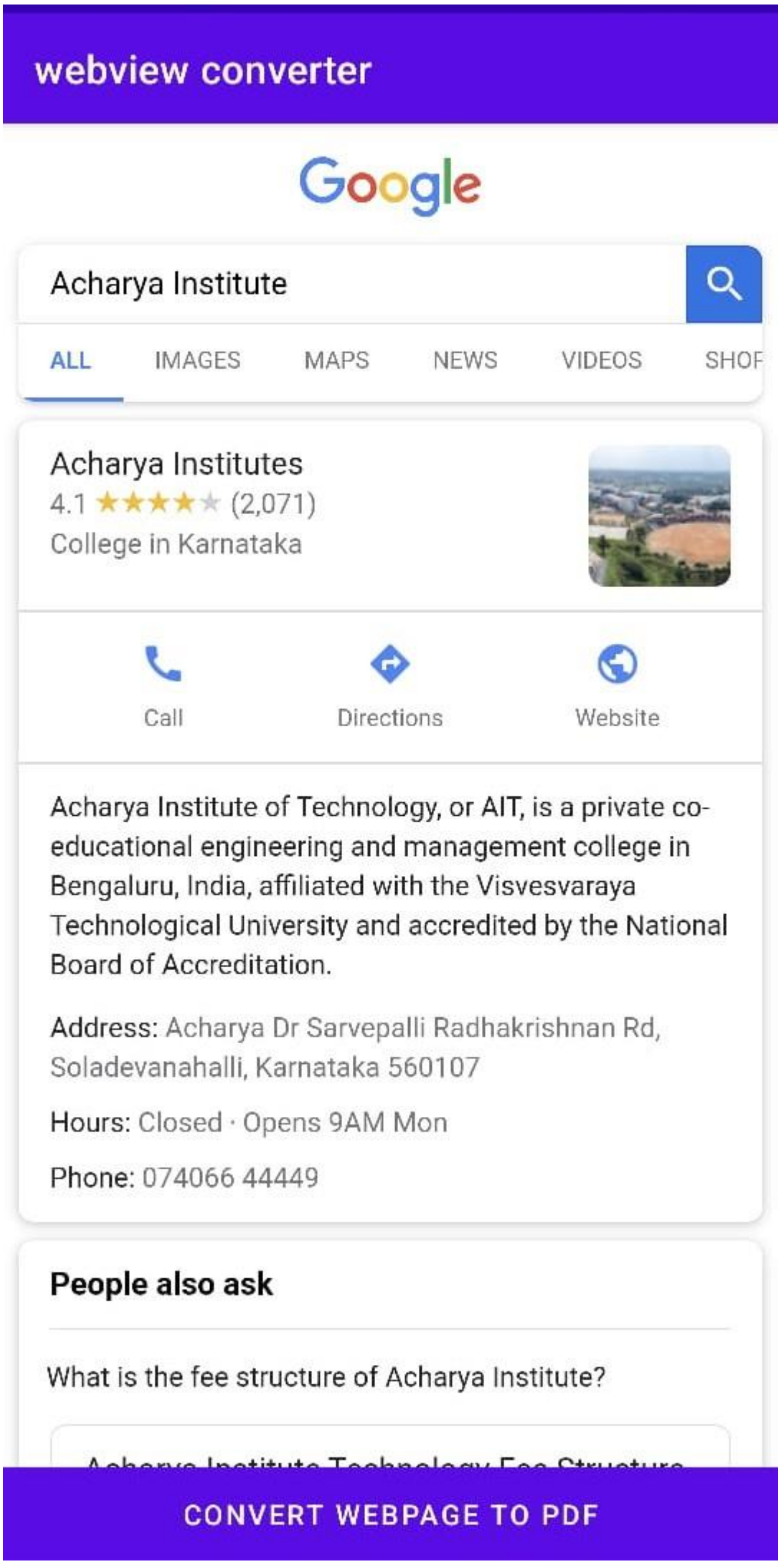


Fig 6.1: Home Page



Acharya Institute

Acharya Institutes
4.1 ★★★★☆ (2,071)
College in Karnataka

Acharya Institute of Technology, or AIT, is a private co-educational engineering and management college in Bengaluru, India, affiliated with the Visvesvaraya Technological University and accredited by the National Board of Accreditation.

Address: Acharya Dr Sarvepalli Radhakrishnan Rd, Soladevanahalli, Karnataka 560107

Hours: Closed · Opens 9AM Mon

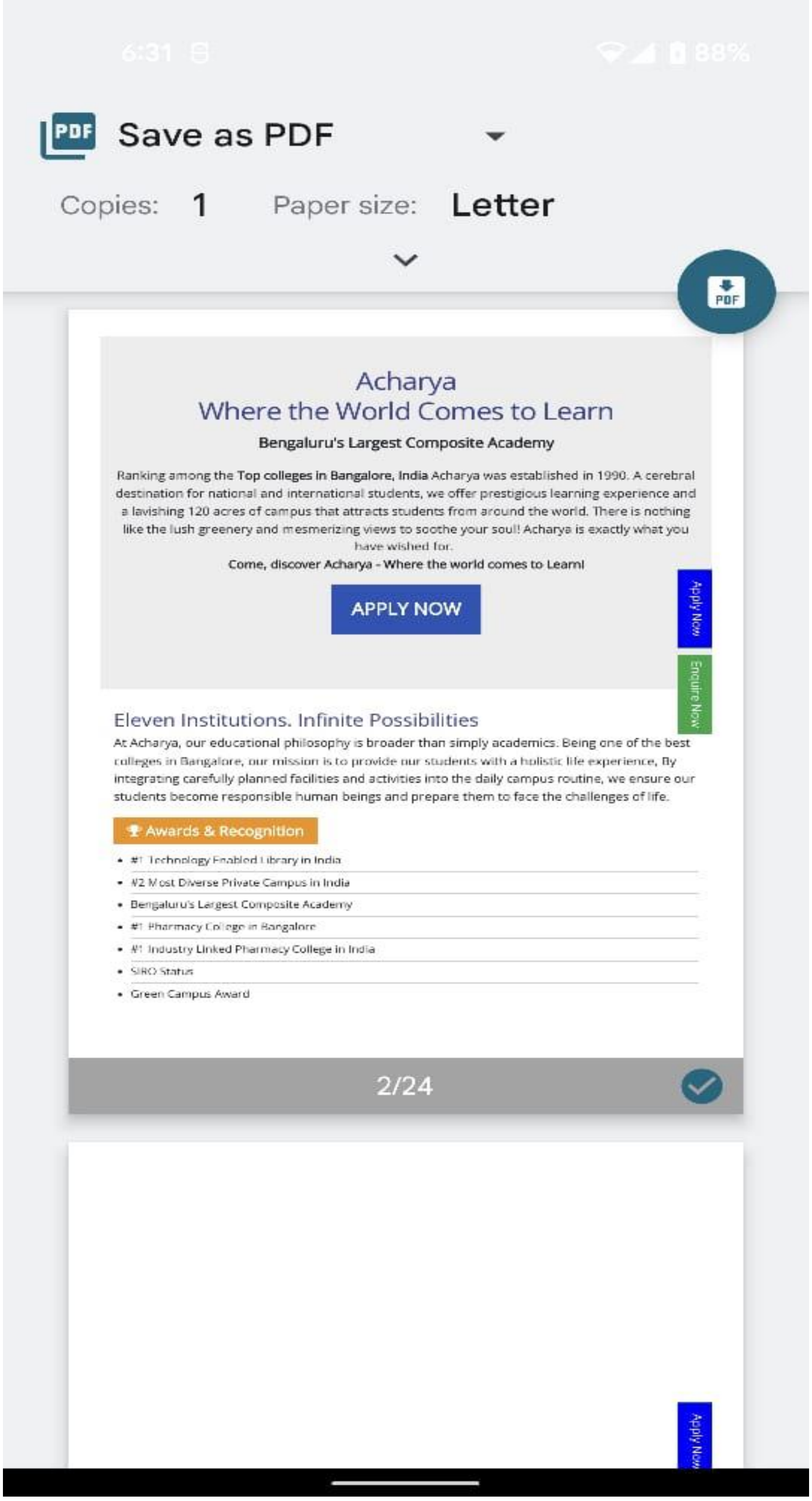
Phone: 074066 44449

People also ask

What is the fee structure of Acharya Institute?

Acharya Institute Technology Fee Structure

Fig 6.2:Search Webpage



Save as PDF

Copies: 1 Paper size: Letter

Acharya
Where the World Comes to Learn

Bengaluru's Largest Composite Academy

Ranking among the **Top colleges in Bangalore, India** Acharya was established in 1990. A cerebral destination for national and international students, we offer prestigious learning experience and a lavishing 120 acres of campus that attracts students from around the world. There is nothing like the lush greenery and mesmerizing views to soothe your soul! Acharya is exactly what you have wished for.

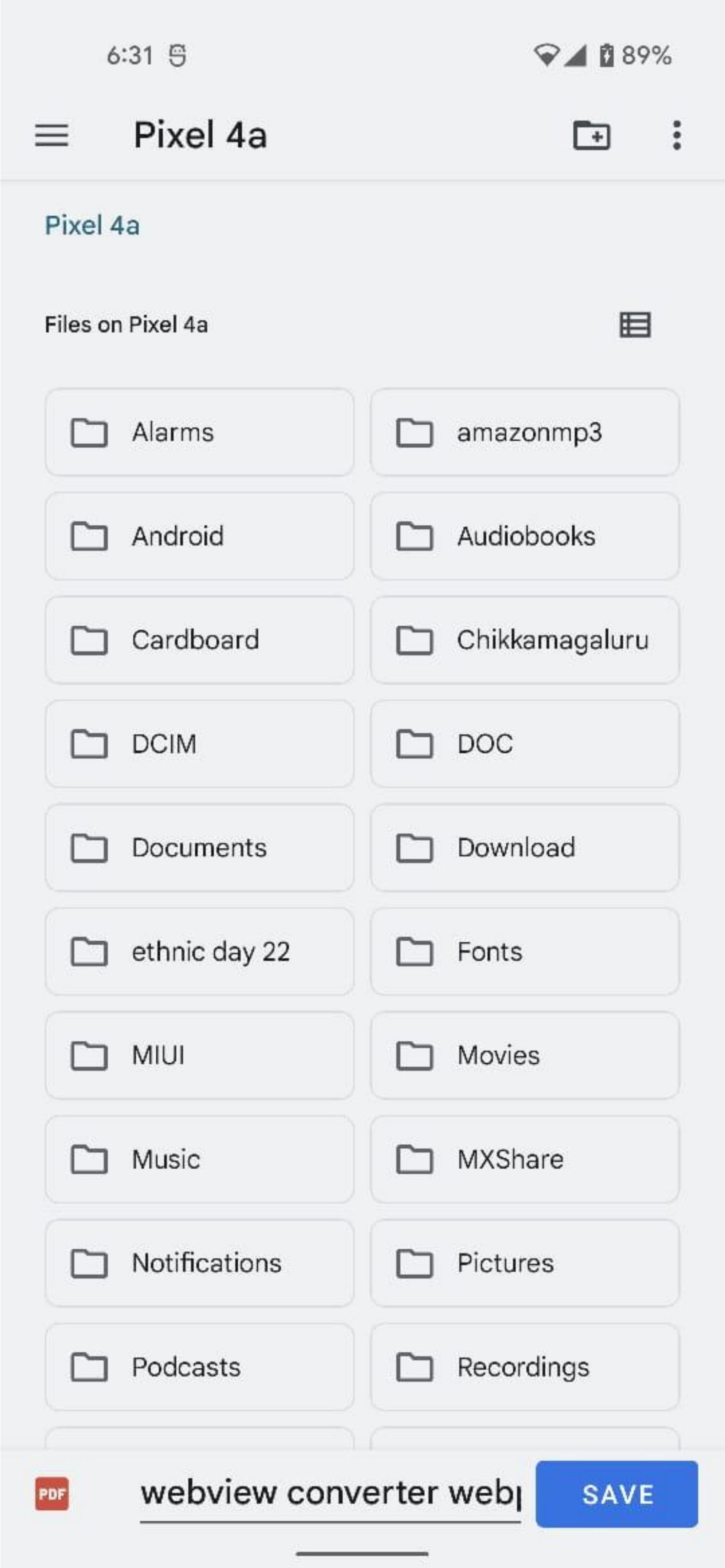
Come, discover Acharya - Where the world comes to Learn!

Eleven Institutions. Infinite Possibilities

At Acharya, our educational philosophy is broader than simply academics. Being one of the best colleges in Bangalore, our mission is to provide our students with a holistic life experience, By integrating carefully planned facilities and activities into the daily campus routine, we ensure our students become responsible human beings and prepare them to face the challenges of life.

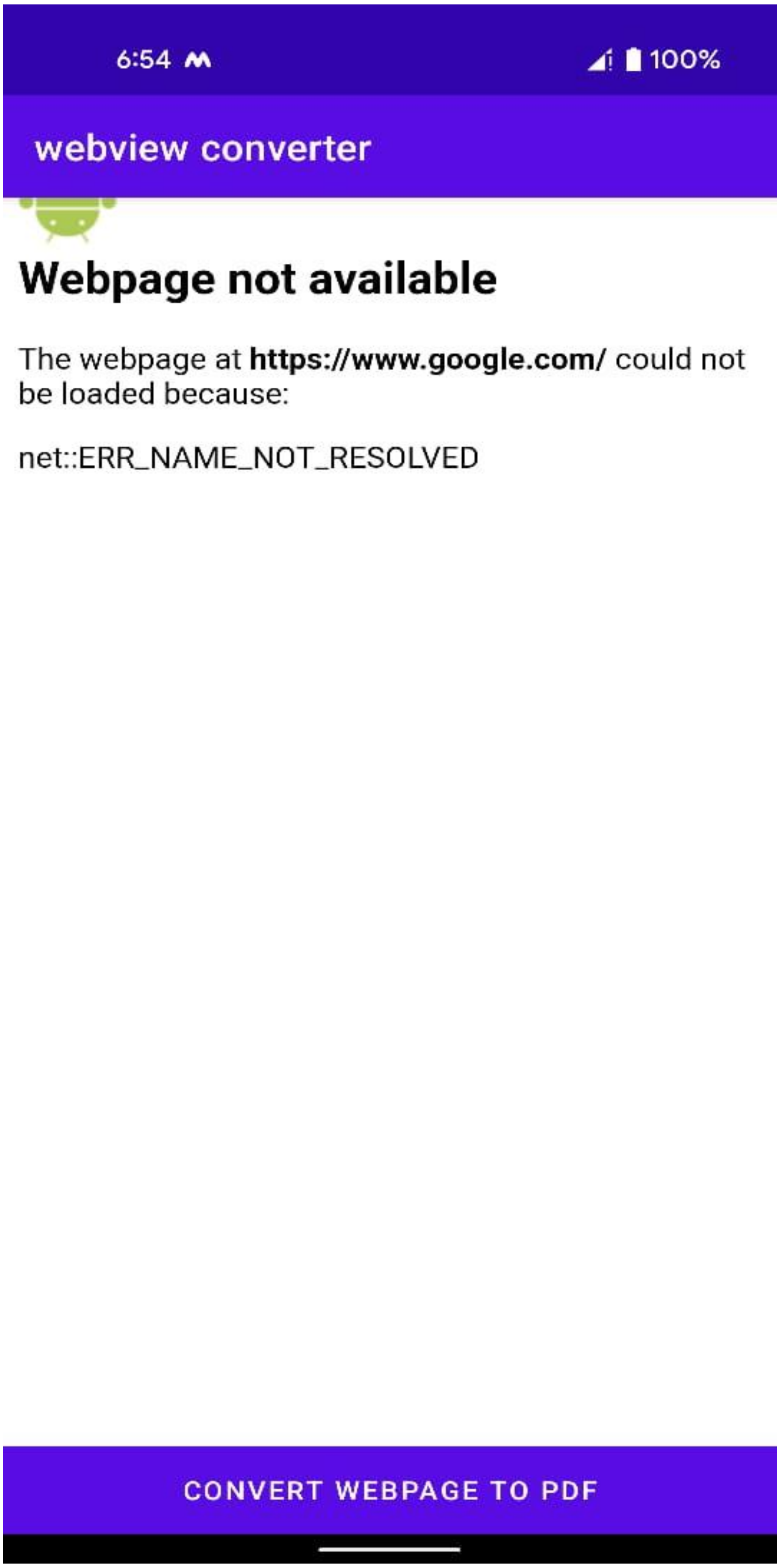
- #1 Technology Enabled Library in India
- #2 Most Diverse Private Campus in India
- Bengaluru's Largest Composite Academy
- #1 Pharmacy College in Bangalore
- #1 Industry Linked Pharmacy College in India
- SIRO Status
- Green Campus Award

Fig 6.3:Converting Webpage into PDF



webview converter web|

Fig 6.4:Saving File To Device



Webpage not available

The webpage at **https://www.google.com/** could not be loaded because:

net::ERR_NAME_NOT_RESOLVED

Fig 6.5: Error Page (When Internet Connection Not Available)

Chapter 7

CONCLUSION, FUTURE ENHANCHEMENTS AND REFERENCES

7.1 Conclusion

Web view converter to PDF is a app that helps students in saving the web view into PDF. The app is a self dependent activity that can engage user in saving PDF, development of visualizing ideas. Android as a full, open and free mobile device platform, with its powerful function and good user experience rapidly developed into the most popular mobile operating system. This report gives an overview of the different challenges and issues faced in android app development. The experience of developing an android app is quite challenging, motivating as well as satisfying.

7.2 Future enhancements

To further enhance the Android app project for converting webpages into PDF files, there are several improvements that can be considered. First, implementing a feature to capture dynamic web content, such as videos or interactive elements, would provide a more comprehensive conversion experience. This could involve using technologies like HTML5 parsing or browser automation to capture and embed dynamic elements within the generated PDF. Additionally, optimizing the app's performance and resource usage, especially when dealing with large or complex webpages, would improve user experience and ensure smooth conversion. Another enhancement could be the inclusion of a PDF editing functionality, allowing users to annotate, merge, or rearrange pages within the converted PDF files. Furthermore, integrating cloud storage options, such as Google Drive or Dropbox, would enable seamless synchronization and backup of the generated PDFs. Lastly, incorporating support for multiple languages and character encodings would ensure accurate conversion and compatibility with diverse web content.

7.3 References

- https://www.geeksforgeeks.org/
- https://www.javatpoint.com/
- https://stackoverflow.com/
- https://youtube.com/