

MINI PROJECT

**(2021-22)**

**“Handwriting Recognition”**

Project Report

**Institute of Engineering & Technology**

**Submitted By -**

###### Piyush Keshari (191500538)

###### Shivansh Kulshrestha (191500772)

###### Rohan Kumar (191500669)

###### Shishank Agrawal (191500576)

###### Aditya Srivastava (191500063)

**Under the Supervision Of Mr. Mandeep Singh Technical Trainer**

**Department of Computer Engineering & Applications**

Department of Computer Engineering and Applications GLA University, 17 km. Stone NH#2, Mathura-Delhi Road, Chaumuhan, Mathura – 281406 U.P (India)

## Declaration

I/we hereby declare that the work which is being presented in the Bachelor of technology. Project **“Handwriting Recognition”**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my/our own work carried under the supervision of **Ms. Ruchi Gupta, Technical Trainer, Dept. of CEA,GLA University.**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

**Sign**: *PiyushKeshari* **Sign**: *ShivanshKulshrestha*

**Name of Candidate**: Piyush Keshari **Name of Candidate**: Shivansh Kulshrestha

**University Roll No**.:191500538 **University Roll No**.:191500772

**Sign:** *ShishankAgrawal* **Sign**: *RohanKumar*

**Name of Candidate:** Shishank Agrawal **Name of Candidate**: Rohan Kumar

**University Roll No**.:191500576 **University Roll No**.:191500669

**Sign:** *AdityaSrivastava*

**Name of Candidate**: Aditya Srivastava

**University Roll No:** 191500063

Department of Computer Engineering and Applications GLA University, 17 km. Stone NH#2, Mathura-Delhi Road, Chaumuhan, Mathura – 281406 U.P (India)

## Certificate

###### This is to certify that the project entitled “Handwriting Recognition”, carried out in Mini Project – I Lab, is a bonafide work by Piyush Keshari, Shivansh Kulshrestha, Shishank Agrawal, Rohan Kumar and Aditya Srivastava and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

**Signature of Supervisor:**

**Name of Supervisor:** Ms. Ruchi Gupta

**Date:**

**Training Certificates**

* **Shivansh Kulshrestha**



* + **Piyush Keshari**



* + **Shishank Agrawal**





* + **ROHAN KUMAR**
  + 
  + **ADITYA SRIVASTAVA**

****

Department of Computer Engineering and Applications GLA University, 17 km. Stone NH#2, Mathura-Delhi Road, Chaumuha, Mathura – 281406 U.P (India)

**ACKNOWLEDGEMENT**

Presenting the ascribed project paper report in this very simple and official form, we would like to place my deep gratitude to GLA University for providing us the instructor Ms Ruchi Gupta, our technical trainer and supervisor.

She has been helping us since Day 1 in this project. She provided us with the roadmap, the basic guidelines explaining on how to work on the project. She has been conducting regular meeting to check the progress of the project and providing us with the resources related to the project. Without her help, we wouldn’t have been able to complete this project.

And at last but not the least we would like to thank our dear parents for helping us to grab this opportunity to get trained and also my colleagues who helped me find resources during the training.

###### Thanking You

**Sign**: *Piyush* **Sign**: *ShivanshKulshrestha*

**Name of Candidate**: Piyush Keshari **Name of Candidate**: Shivansh Kulshrestha

**University Roll No**.:191500538 **University Roll No**.:191500772

**Sign:** *ShishankAgrawal* **Sign**: *RohanKumar*

**Name of Candidate:** Shishank Agrawal **Name of Candidate**: Rohan Kumar

**University Roll No**.:191500576 **University Roll No**.:191500669

**Sign:** *AdityaSrivastava*

**Name of Candidate**: Aditya Srivastava

**University Roll No:** 191500063

**ABSTRACT**

###### In this project, we are creating a web application, which has its basic idea derived from machine learning and its called Handwritten character recognition. This application will provide us a platform to upload images of numbers drawn to be identified by our system. All that the users have to do is upload an image of a number that they have drawn on MS Paint (or any other drawing tool), and our application will predict which number it is, by learning from its previous results. The application divides the image into pixels, which act as a matrix and then predicts which number its most likely to be. Whatever may be the size of the image our system scales down the image into the pixel ratio of 28 \* 28(784 pixels) and then tests it to tell the value. The user can also take images from their phones and upload them on our site to get the required digit recognition. Neural networks have lately been applied in a variety of pattern recognition applications. Because various people's handwritings varies, it might be difficult to detect handwritten characters. Handwritten character identification is a field of pattern recognition that has gotten a lot of attention in recent decades. In handwritten character recognition, neural networks play a crucial role.

###### Artificial intelligence and is changing people’s life all over the world. The influence of Artificial intelligence are expected to increase because of the advance changes of technology and the way it is supposed to reduces the work and increase the accuracy and efficiency of this work. Furthermore designing solutions for the problems that we may face in future is essential. Like this application, machine learning is definitely the way of the future.

**CONTENTS**

Cover Page i

[Declaration ii](#_TOC_250030)

[Certificate iii](#_TOC_250029)

Training Certificate… iv

[Acknowledgement vii](#_TOC_250028)

[Abstract viii](#_TOC_250027)

Content ix

[List Of figures xi](#_TOC_250026)

List Of tables xii

[Chapter 1Introduction 1](#_TOC_250025)

* + [1.1 Context… 1](#_TOC_250024)
  + [1.2 Motivation 1](#_TOC_250023)
  + [1.3 Objective 2](#_TOC_250022)
  + [1.4 Existing System 2](#_TOC_250021)
  + 1.4 Sources 3

Chapter 2 Software Requirement Analysis 4

* + [2.1 Impact Of Books On Daily Life 4](#_TOC_250020)
  + [2.2 Problem Statement… 5](#_TOC_250019)
  + [2.3 Hardware and Software Requirements 6](#_TOC_250018)
  + [2.4 Modules and Functionalities 6](#_TOC_250017)
  + [2.5 Bookopedia on Android Application……………,,,,,, 7](#_TOC_250016)

Chapter 3 Software Design… 8

* + [3.1 Use Case Diagram 8](#_TOC_250015)
  + 3.2 Data Flow Diagram 11
  + [3.3 Sequence Diagram 12](#_TOC_250014)

[Chapter 4 Technology Used… 13](#_TOC_250013)

* + [4.1 Android… 13](#_TOC_250012)
  + [4.2 Version of Android 14](#_TOC_250011)
  + [4.3 Tools and Languages 15](#_TOC_250010)
  + [4.4 Basic Terminology 16](#_TOC_250009)

[Chapter 5 Implementation and User Interface 19](#_TOC_250008)

* + 5.1 Implementation of Bookopedia 19
  + [5.2 User Interface 23](#_TOC_250007)

[Chapter 6 Testing 33](#_TOC_250006)

* + [6.1 Installation Testing… 33](#_TOC_250005)
  + [6.2 Unit Testing… 34](#_TOC_250004)
  + [6.3 User Testing 37](#_TOC_250003)
  + [6.4 Performance Testing 38](#_TOC_250002)
  + 6.5 Compatibility Testing 39

[Chapter 7 Conclusion 40](#_TOC_250001)

[References *41*](#_TOC_250000)

**LIST OF FIGURES**

###### Existing System 2

###### Use Case Diagram 9

###### Data Flow Diagram 11

###### Sequence Diagram 12

###### Android Kit-Kat… 14

###### Flow Chart for User 21

###### Splash Screen 23

###### Register Page 23

###### Login Page 24

###### Forget Password… 24

###### Navigation Drawer 25

###### Dashboard Fragment… 26

###### Profile Page 27

###### Favourite Page 28

###### Sign-out Page 28

###### About App Page 29

###### FAQ Page 29

###### Search Book Page 30

###### Description Page 30

###### Book List 31

###### Book added to Favourite… 31

###### Favourites 32

###### Book removed from Favourite 32

# LIST OF TABLES

###### Version of Android… 14

###### Unit testing of Bookopedia 34

* 1. **CONTEXT**

CHAPTER-1 INTRODUCTION

This Web application “Handwriting Recognition” has been submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering at GLA University, Mathura supervised by Ms Ruchi Gupta. This project has been completed approximately three months and has been executed in modules, meetings have been organised to check the progress of the work and for instructions and guidelines.

### MOTIVATION

The Role of Artificial Intelligence made it possible for machines to learn from **experience to perform tasks more efficiently**. The Artificial neural network is one of its advancements which is inspired by the structure of the human brain that helps computers and machines more like a human.

Neural networks have a remarkable ability to retrieve meaningful data from imprecise data, that is used in detecting trends and extract patterns that are difficult to understand either by computer or humans. A trained NN can be made an "expert" in the information that has been given to analyze and can be used to provide projections.

In the century we are living, the world is progressing at a really great pace, a large number of technologies come up every single day. Keeping up with the technology is also important to survive in this world of digitalization and learning.

### OBJECTIVE

The main objective of this web application is to create a hand written character recognition system which can identify numbers by using images uploaded by the user. It uses neural networks to identify the digits in the image. After uploading the image drawn using any drawing tool (like MS Paint, Photoshop) the user will be able to procure a prediction made by our system. There will also be a facility to upload images of hand-drawn digits, which will be identified by our applications.

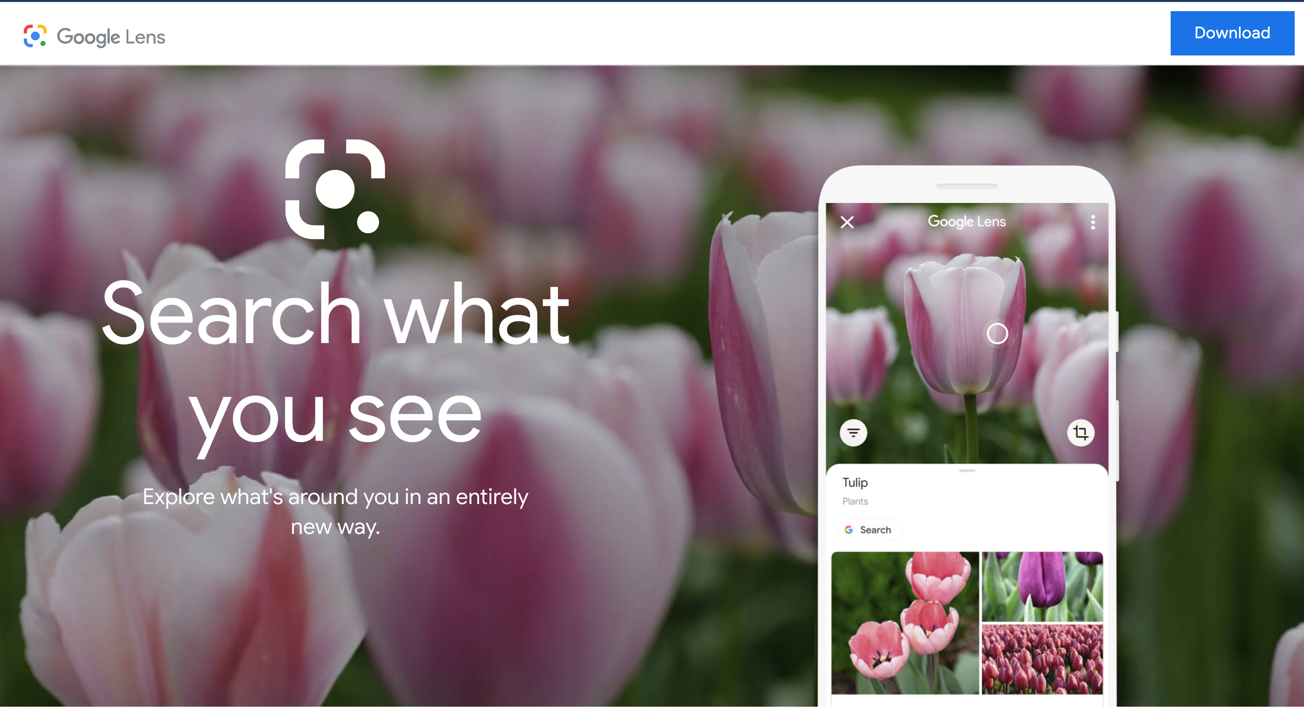
This application developed can be used at a variety of places, for eg, if we want a particular digit in a different font, our system can be used to identify the digit from its handwritten form, so that the user can change its style.  Furthermore, OCR plays an important role for digital libraries, allowing the entry of image textual information into computers by digitization, image restoration, and recognition methods

### EXISTING SYSTEM

Our system has many existing projects similar to it but we would be implementing it using web applications(HTML, CSS, Javascript).

As soon as the user enters opens the website, there will be landing page containing the name. Then, you will click on the upload image button. This will lead to a page where the users can upload an image. Then, with the help of neural networks, our application will test the given image and fetch the results based on prior results.

The pre-existing system similar to this web application is Google Lens.



**Figure-1**

* 1. **SOURCES**

The source of our project (including all the project work, documentations and presentations) will is available at the following link

https://github.com/shishank2311/Handwritten-Digit-And-Character-Recognition

**CHAPTER -2**

**SOFTWARE REQUIREMENT ANALYSIS**

### IMPACT OF MACHINE LEARNING AND ITS SCOPE

Machine learning is a method of data analysis that automates analytical model building.

It is a branch of [artificial intelligence](https://www.sas.com/en_in/insights/analytics/what-is-artificial-intelligence.html) based on the idea that systems can learn from data,

identify patterns and make decisions with minimal human intervention.

The scope of Machine Learning is not limited to the investment sector. Rather, it is expanding

across all fields such as banking and finance, information technology, media & entertainment,

gaming, and the automotive industry. As the Machine Learning scope is very high, there are

some of the areas where researchers are working toward revolutionizing the world for the future.

Some of them are :

a) Automative Industry

b) Robotics

c) Quantum Computing

d) Computer Vision

### PROBLEM STATEMENT

The Handwriting Recognition is a Web Application which will allow the users to upload an image of hand drawn digits, then our system will divide the image into a matrix and will scale down the image to 28\*28 pixels then our system will predict the digit based on previous results.

As a help to the users, there will be FAQ Section containing the basic details and guidelines and the problems that can be faced by the users while using our application.

There will be a contact us page where users can share their thoughts with us about our web application and even contribute to our site. There will also be an about us page.

### 

### HARDWARE AND SOFTWARE REQUIREMENTS

**Hardware Requirement**

* + - Processor :intel i5
    - Operating System :Any Operating System
    - RAM : 8 GB (or higher)
    - Hard disk : 256GB

**Software Requirement**

* + - Software used: VS CODE, JUPITER NOTEBOOK
    - Language used : Python, HTML, CSS, Javascript.
    - Database: Jupiter Notebook
    - User Interface Design : WebApplication and Anvil

**MODULES AND FUNCTIONALITIES**

* **Home Screen**: The first screen with which the user interacts will be this screen containing the logo. It also contains quick links for other pages.
* **FAQ Pages:** This page contains some of the questions that might arise in the mind of the users while using the web app and to answer those, these answers are pre-written.
* **Contact Us :** This page will contain our social media links and a form to contact our members.
* **About Us:** This page contains information about all the team members.
* **Navigation Bar** : It contains a logo that will redirect the user to functionality page, the about us page ,the Contact us page, the FAQ page.

CHAPTER- 3

**SOFTWARE DESIGN**

### USE-CASE DIAGRAM:

### 

**Figure-2: Use–Case Diagram**

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

Due to their simplistic nature, use case diagrams can be a good communication tool for stakeholders. The drawings attempt to mimic the real world and provide a view for the stakeholder to understand how the system is going to be designed. Siau and Lee conducted research to determine if there was a valid situation for use case diagrams at all or if they were unnecessary. What was found was that the use case diagrams conveyed the intent of the system in a more simplified manner to stakeholders and that they were "interpreted more completely than class diagrams".

* 1. **DATA FOW DIAGRAM**

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

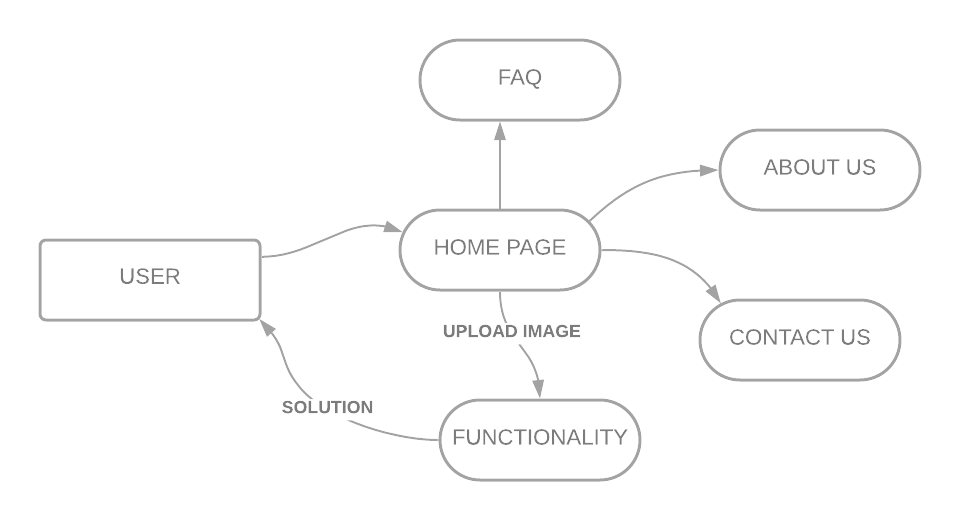
There are several notations for displaying data-flow diagrams. The notation presented above was described in 1979 by Tom DeMarco as part of structured analysis.

For each data flow, at least one of the endpoints (source and / or destination) must exist in a process. The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes.

The data-flow diagram is a tool that is part of structured analysis and data modeling. When using UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.

Data-flow diagrams can be regarded as inverted Petri nets, because places in such networks correspond to the semantics of data memories. Analogously, the semantics of transitions from Petri nets and data flows and functions from data-flow diagrams should be considered equivalent.

The DFD for the Handwritten Character Recognition is shown below:



**Figure-3: Data Flow Diagram**

**CHAPTER-4 TECHNOLOGY USED**

**CHAPTER-4**

**TECHNOLOGY USED**

### MACHINE LEARNING

Machine learning is a method of data analysis that automates analytical model building. It is a branch of [artificial intelligence](https://www.sas.com/en_in/insights/analytics/what-is-artificial-intelligence.html) based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

Because of new computing technologies, machine learning today is not like machine learning of the past. It was born from pattern recognition and the theory that computers can learn without being programmed to perform specific tasks; researchers interested in artificial intelligence wanted to see if computers could learn from data. The iterative aspect of machine learning is important because as models are exposed to new data, they are able to independently adapt. They learn from previous computations to produce reliable, repeatable decisions and results. It’s a science that’s not new – but one that has gained fresh momentum.

While many machine learning algorithms have been around for a long time, the ability to automatically apply complex mathematical calculations to big data – over and over, faster and faster – is a recent development. Here are a few widely publicized examples of machine learning applications you may be familiar with:

• The heavily hyped, self-driving Google car? The essence of machine learning.

• Online recommendation offers such as those from Amazon and Netflix? Machine learning applications for everyday life.

• Knowing what customers are saying about you on Twitter? Machine learning combined with linguistic rule creation.

• Fraud detection? One of the more obvious, important uses in our world today.

**Why is machine learning important?**

Resurging interest in machine learning is due to the same factors that have made data mining and Bayesian analysis more popular than ever. Things like growing volumes and varieties of available data, computational processing that is cheaper and more powerful, and affordable data storage.

All of these things mean it's possible to quickly and automatically produce models that can analyze bigger, more complex data and deliver faster, more accurate results – even on a very large scale. And by building precise models, an organization has a better chance of identifying profitable opportunities – or avoiding unknown risks.

### PYTHON

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0 Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting). Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible. Python 2 was discontinued with version 2.7.18 in 2020.

### 

### TOOLS AND LANGUAGES

Tools used to build the model for our machine learning project :-

* + 1. **Jupyter Notebook :** The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.
    2. **VS Code :**  Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

Languages used in building our Machine Learning Application are classified as Front End and Backend and are :

* + 1. **PYTHON:** Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

* + 1. **HTML :** The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

* + 1. **CSS :** Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.[1] CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.[2]

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

* + 1. **JAVASCRIPT :** JavaScript (/ˈdʒɑːvəˌskrɪpt/),[10] often abbreviated as JS, is a programming language that conforms to the ECMAScript specification.[11] JavaScript is high-level, often just-in-time compiled and multi-paradigm. It has dynamic typing, prototype-based object-orientation and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web.[12] Over 97% of websites use it client-side for web page behavior,[13] often incorporating third-party libraries.[14] All major web browsers have a dedicated JavaScript engine to execute the code on the user's device.

* + 1. **ANVIL :** Anvil is a web-based web-app builder which allows you to write your app, both client- and server-side, in Python (the client-side code gets transpiled to Javascript).

### BASIC TERMINOLOGY

* + 1. **Layout:** Layout is the parent of view. It arranges all the views in a proper manner on the screen.
    2. **Activity**: An activity can be referred as your device’s screen which you see. User can place UI elements in any order in the created window of user’s choice.
    3. **View**: A view is an UI which occupies rectangular area on the screen to draw and handle user events.
    4. **Emulator**: An emulator is an Android virtual device through which you can select the target Android version or platform to run and test your developed application.
    5. **Manifest file**: Manifest file acts as a metadata for every application. This file contains all the essential information about the application like app icon, app name, launcher activity, and required permissions etc.
    6. **API:** Short for Application Programming Interface. APIs are functions that developers can call on to access specific features by calling upon programs, code, and services that others have written. For example, if a developer wants to draw a button on the screen, she can insert a small bit of code that says “draw this kind of button, with this color and size and style, at this location” instead of dozens of lines of code that tells the graphics processor, in detail, exactly how to draw a button. If the application wants your location, it can use the location API to “get the device’s location” and let Google’s code handle the rest, instead of requiring the developer to build an entire location service from scratch just for her own app. There are thousands of APIs in Android, covering everything from drawing interface elements, to the cameras, to location access, to accessing storage, to 3D graphics (see: OpenGL ES) and much more.
    7. **Intent:** Intents are an essential part of the Android ecosystem. They are used to express an action to be performed. Intents allow you to interact with components from the same applications as well as with components contributed by other applications. It can be classified into implicit and explicit intents.
    8. **Implicit intent:** It does not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it.
    9. **Explicit Intent:** It specifies the component to start by name. You’ll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start.
    10. **Navigation bar**: Android Navigation Drawer is a sliding left menu that is used to display the important links in the application. Navigation drawer makes it easy to navigate to and fro between those links. It’s not visible by default and it needs to opened either by sliding from left or clicking its icon in the Action Bar.
    11. **Fragment**: A Fragment represents a behavior or a portion of user interface in a Fragment Activity. You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities.

**CHAPTER -5 IMPLEMENTATION AND USER INTERFACE**

Creating an web app concept design with screen sketches and functional flow diagrams is the best way to communicate your vision to the full stack developer. Making the concept clear to the developer is probably the most important factor in successful web app development. Yet it is one of the most common problems or obstacles in a web app development outsourcing project.

No matter what the marketing and profit goals are or if you are outsourcing an web app for your personal use, you need to fully design and document the web app concept if you expect a programmer to make your vision a reality. Developers are not mind readers and even descriptions given during conversations can be very fleeting or interpreted differently. Fully documenting your concept, therefore, leaves little to chance. The two most important things to do are: A) make a comprehensive description of how the web application works and what it does (functionality) and B) create a comprehensive description of what the user sees and does (look and feel).

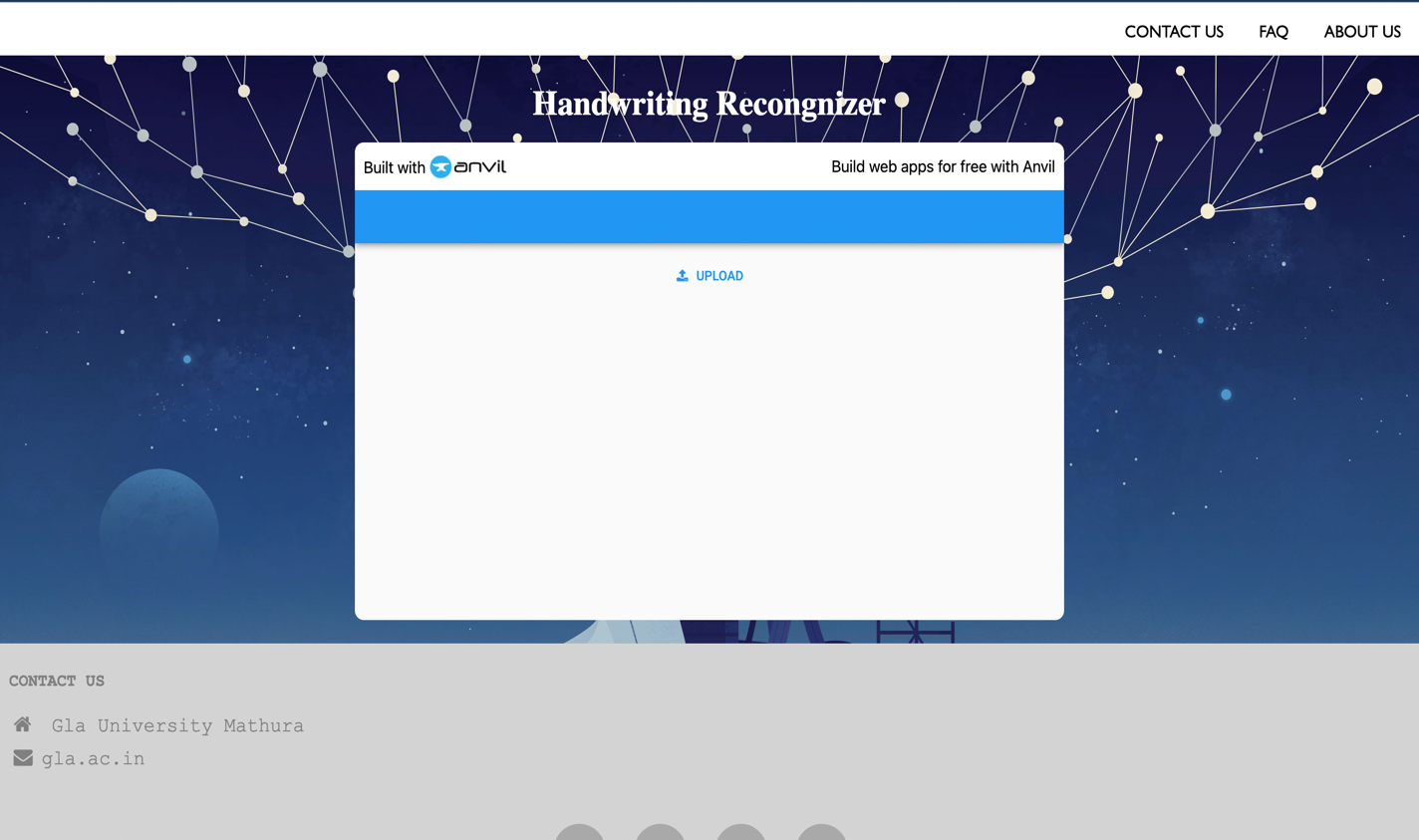
* 1. **Implementation of the HandWriting Recognition :**

Implementation of HandWriting Recognition is taken place in various phases. Firstly we build the backend. Backend is developed using Keras, Matlplotlib, Tensorflow, Python and OpenCV. Front End is developed using HTML,CSS, Javascript, and Anvil.

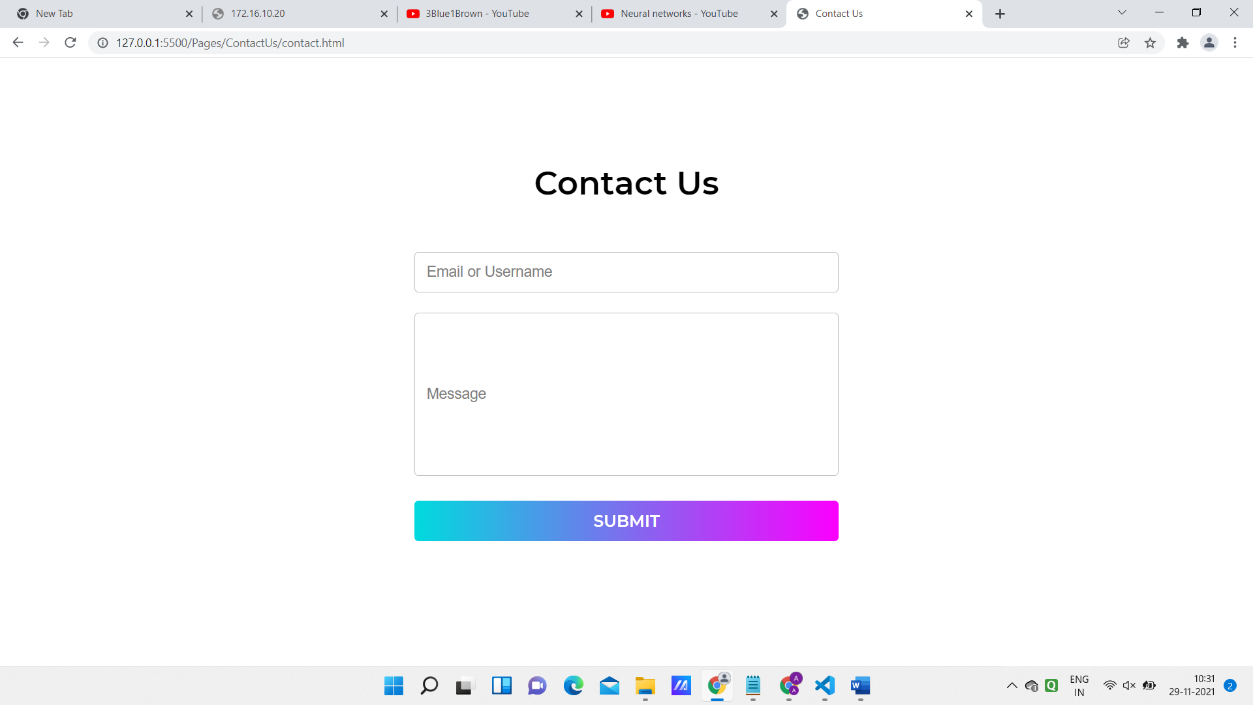
**Step to be followed to develop the app:**

1. Firstly we create the splash screen with animated text and linked it with the main Activity through HTML.
2. Now, we are going to create Navigation bar for that purpose we have used following functionality of web page :
   * About US
   * Contact US
   * Upload Button
   * FAQ
     1. **Step to be followed by the user**
3. Firstly, the user has to upload the photo by clicking on the upload button on the home page.
4. The user will than wait for the result to appear on the output screen.
5. We also have About Us, Contact Us and FAQ.
6. In the ‘About Us’ page the user can see our information.
7. In the ‘Contact Us’ the user can send us some queries.
8. In the ‘FAQ’ page we have answered some of the common questions the user might have.
   1. **User Interface**

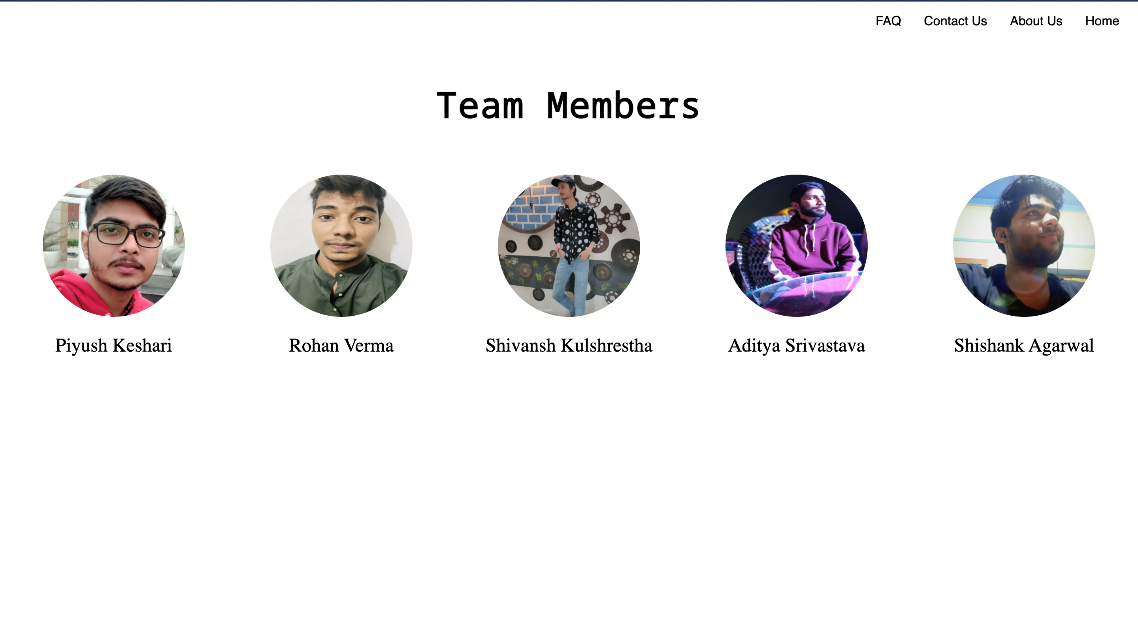
* **Index Page**

****

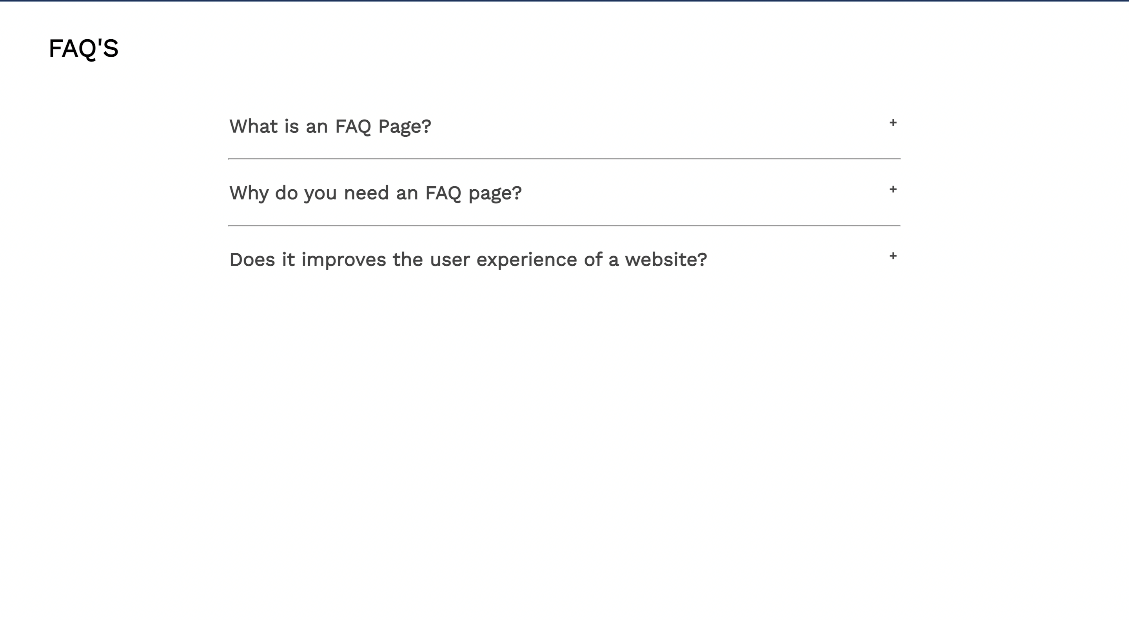
* **Contact Us**

****

* **About us**

****

* **FAQ**

****

* **GITHUB**

****

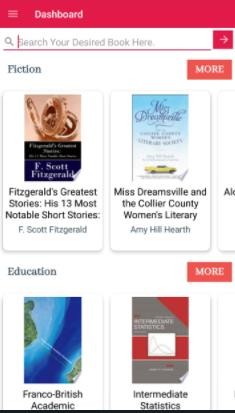
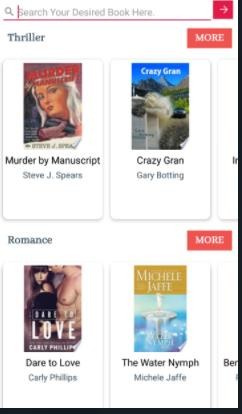
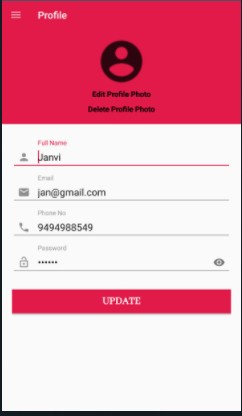
* **Dashboard Fragment**

Figure-12: Dashboard Fragment (A)



* **Profile**

**Figure-13: Dashboard Fragment (B)**



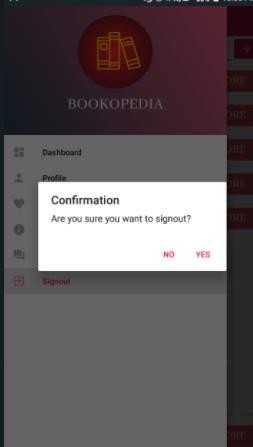
**Figure-14: Profile page**

* **Favourites Page**



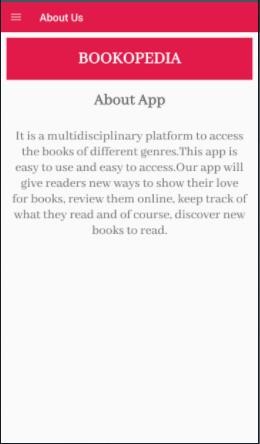
* **Sign-Out Page:**

**Figure-15: Favourite Page**



**Figure-16: Sign-out page**

* **About App**

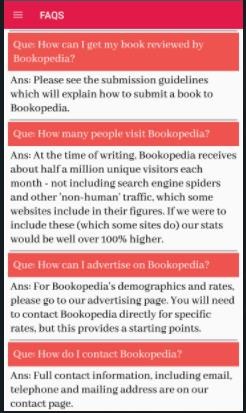


* FAQ’s

Figure-17: About App



Figure-18: FAQ’s (a)



* **Search Book**

Figure-19: FAQ’s (b)

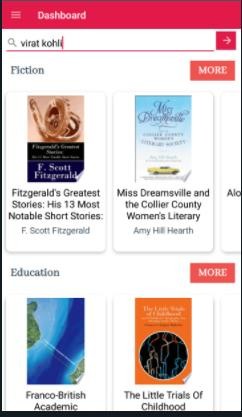


Figure-20: Search book

* **Description Page**



Figure-21: Description Page(a)

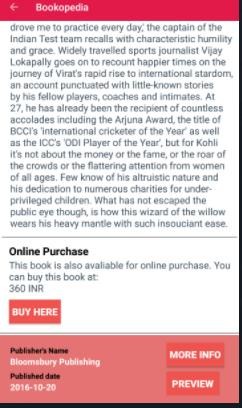
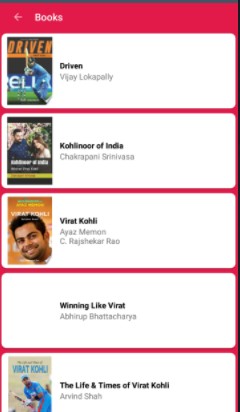


Figure-22: Description Page (b)

* **Book List**



* **Adding book to the favourites**

**Figure-23: Book List**

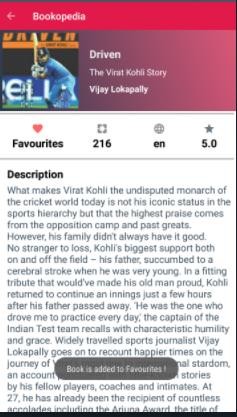
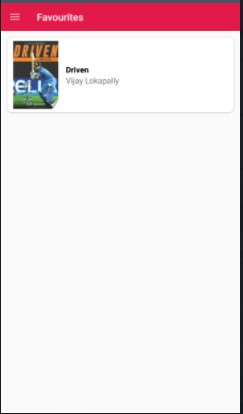


Figure-24: Adding book to the Favourites

* Favourites



* Book Removed from Favourites

Figure-25: Showing Favourites



Figure-26: Book Removed

**CHAPTER - 6 TESTING**

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery. It is very important to work the system successfully and achieve high quality of software. Testing include designing a series of test cases that have a high likelihood of finding errors by applying software-testing techniques.

System testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. The system should be checked logically. Validations and cross checks should be there. Avoid duplications of record that cause redundancy of data.

In other Words, Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

The Android framework includes an integrated testing framework that helps you test all aspects of your application and the SDK tools include tools for setting up and running test applications. Whether you are working in Eclipse with ADT or working from the command line, the SDK tools help you set up and run your tests within an emulator or the device you are targeting.

There are different types of testing some of them are listed below:

* 1. **Installation Testing:**

There are two types of apps on an Android device i.e., Pre-installed applications and the applications which are installed later by the user.

For both of the above, installation testing is carried out by our teammates. It is ensuring smooth installation of the application without ending up in errors, partial installation etc.

* 1. **Unit Testing**

It focuses on smallest unit of software design. In this we test an individual unit or groups of inter related units. It is often done by programmer by using sample input and observing its corresponding outputs. In this testing technique we are primarily focuses on

* + - Loop methods and function is working fine or not.
    - Misunderstood or incorrect Arithmetic precedence
    - Incorrect Initialization

Unit Testing of the app:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test cases** | **Description** | **Expected Outcome** | **Result** |
| 1 | Home Page | Should display a Upload button | Pass |
| 2 | Contact Us | Should display a box where the user can enter their queries. | Pass |

|  |  |  |  |
| --- | --- | --- | --- |
| 3 | About us | Should display login screen And ask for your credentials. | Pass |
| 4 | FAQ | Should display frequently questions | Pass |
| 5 | Result | Should the result after processiong | Pass |

Table 1: Unit Testing of Handwritten Character Recognition

* 1. **User Testing**

User testing is the process through which the interface and functions of a website, app, product, or service are tested by real users who perform specific tasks in realistic conditions. The purpose of this process is to evaluate the usability of that website or app and to decide whether the product is ready to be launched for real users.

This app was tested by our team mates and friends who are using different mobile phones (and having different android version) also tested on different emulator to check its performance and it seems to be working fine and users of this app are satisfied with the facilities and performance of the app and like the way how the app is worked.

* 1. **Performance Testing**

In this type of testing we have checked the performances of our application under some peculiar conditions are checked. Those condition include:

* + - Poor/Bad network reception.

Performance is basically tested from 2 ends, application end, and the application server end. Our app is also performing well in this phase of testing as well. And we are getting positive feedback from user of our app.

* 1. **Compatibility Testing**

This application was tested and used on different web browsers like Microsoft Edge, Google chrome, Mozilla Firefox. The application worked fine and is stable. The web application worked fine in portrait mode and there isn't any problem with compatibility.

On all types of testing (that we have performed above) our performing well on our website i.e. Handwritten Character Recognition6 .

**CHAPTER -7 CONCLUSION**

Proposed HandWriting Recognition App is a Web Application that will allow users to upload images of handwritten text. This application takes in a user image as an input and then predicts which digit it is. It uses Neural Network searches the Google Books API with the user input and gets a list of published books based on the users search query. Search result screen will contain a list of book with following details: Author of the Book, title, average, rating Price of the Book. To get the information of the particular book user can click upon the book from the list and then will be taken to the new tab where description and other information related to the book will be available. Users can also add the book to the favourites.

This application has wide range of scope in the upcoming era. It is impossible to arrange the hard copies of every book so this type of application can reduce the barrier to get knowledge at any place in a cost effective, productive way. For students who are interested in learning online can use this application and keep all the books they want to learn from at one place (in favourites section) and can create their own personal E- library. Even individual book stores can have this system of book apps promoting their brand name as Digital Marketing and can gain number of customers.

# REFERENCES

1. **Internshala:** [**https://trainings.internshala.com/**](https://trainings.internshala.com/)
2. **Developer Guide:**

**https://w3school.com**

1. **For rectifying the error:** [**https://stackoverflow.com/**](https://stackoverflow.com/)
2. **For checking our concept:**

**https://www.youtube.com/playlist?list=PLZHQObOWTQDNU6R1\_67000Dx\_ZCJB-3pi**