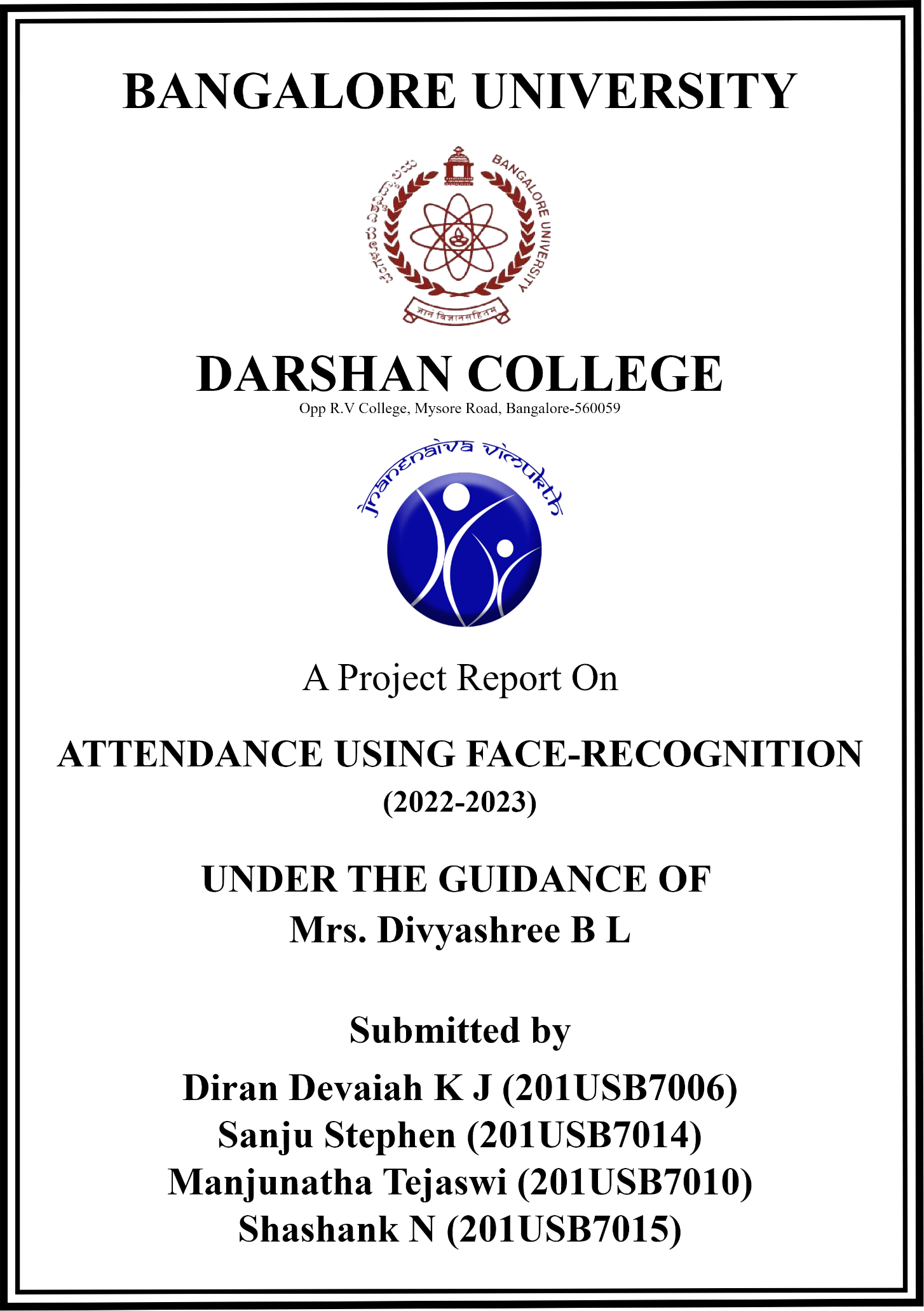
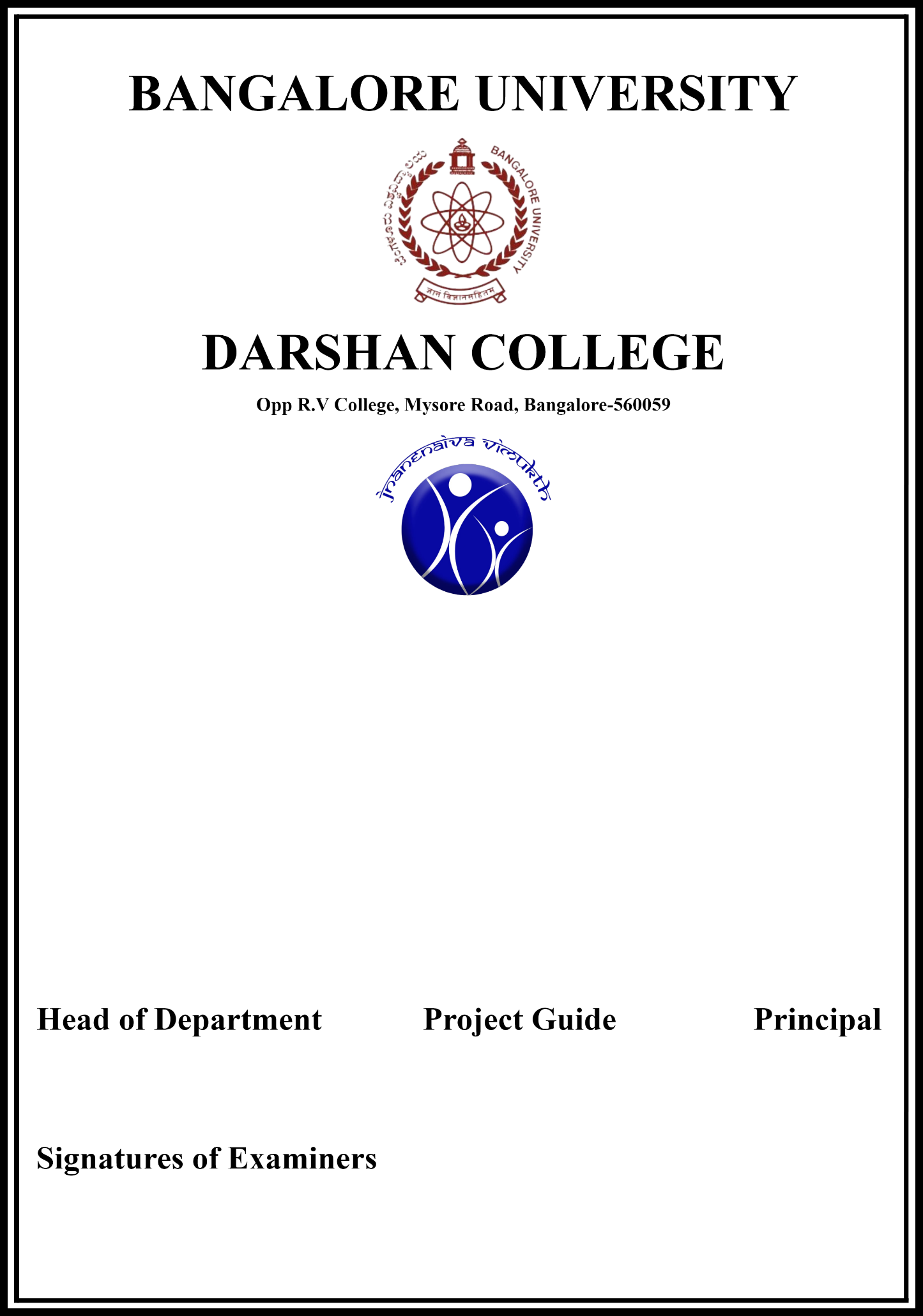
****

**SHASHANK (201USB7015)**

****

**This is to certify that Shashank (201USB7015) has successfully completed the project titled “ATTENDANCE USING FACE-RECOGNITION”, under the guidance of Prof. Divyashree B L, for the practical fulfillment of V Sem course in Mini Project at Darshan College for the academic year 2022-23.**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SL NO** | **TITLE** | **PAGE NO** |
| **1** | **ABSTRACT** | **5** |
| **2** | **INTRODUCTION** | **6** |
| **3** | **LITERATURE SURVEY** | **7** |
| **4** | **SYSTEM REQUIRMENT SPECIFICATION** | **9** |
| **5** | **SYSTEM ANALYSIS** | **10** |
| **6** | **SYSTEM IMPLEMETATION** | **11** |
| **7** | **FEASIBILITY STUDY** | **12** |
| **8** | **SOFTWARE REQUIRMENT SPECIFICATION** | **14** |
| **9** | **SYSTEM DESIGN** | **18** |
| **10** | **CODING** | **20** |
| **11** | **SOFTWARE TESTING** | **31** |
| **12** | **SCREEN –SHOTS** | **35** |
| **13** | **REFERENCES** | **40** |
| **14** | **CONCLUSION** | **41** |
| **15** | **FUTURE ENHANCMENT** | **41** |

**ACKNOWLEDGEMENT**

**We hereby present this project title “FACE-LOGIN” with great pleasure and satisfaction.**

**We would like to take this opportunity to convey our sincere thanks to all those intellectuals concern for their magnanimous vision by the virtue of which we have been guided throughout to accomplish our project.**

**We have great pleasure in expressing our attitude to our beloved principal Rev.Fr. Joy Wilson D’Souza for his continuous encouragement and god blessing.**

**We express our sincere thanks to Prof. Balkhi’s H.O.D BCA Dept., for her support.**

**We would like to express our gratitude to Prof. Divyashree B L, BCA Department, for guiding throughout our project work and showed her great interest and encourage for us.**

**We also express our sincere thanking to all teaching and non-teaching staff of our department.**

**ABSTRACT**

In Modern times, face recognition has become one of the key aspects of computer vision. There are at least two reasons for this trend; the first is the commercial and law enforcement applications, and second is the availability of feasible technologies after years of research. Due to the nature of the problem, computer scientists, neuroscientists and psychologists all share a keen interest in this field. In plain words, it is a computer application for automatically identifying a person from a still image or video frame.

In this paper we proposed automated attendance management system. This system based on face detection and recognition algorithms, which automatically detects the student when they enter into the classroom and marks the attendance by recognizing them.

We used OpenCV which is a great tool for image processing and performing computer vision tasks and Dlib is used to estimate the coordinates (x, y) that map the facial points on a person’s face.

In face recognition there are 2 commonly used open-source libraries namely Dlib and OpenCV. Analysis of facial recognition algorithms is needed as reference for software developers who want to implement facial recognition features into an application program.

When compared to traditional attendance marking this system and using biometrics, iris, saves the time and also helps to monitor the students.

**INTRODUCTION**

Face is one of the most important biometric features of a human. A human can recognize different faces without difficulty. Yet it is a challenging task to design a robust computer system for face identification. The inadequacy of automated face recognition systems is especially apparent when compared to our own innate face recognition ability. Humans perform face recognition, an extremely complex visual task, almost instantaneously and our own recognition ability is far more robust than any computer can hope to be.

Human can recognize a familiar individual under very adverse lighting conditions, from varying angles or viewpoints. While research into this area dates back to the 1960's, it is only very recently that acceptable results have been obtained.

Biometric techniques play a bigger and bigger role in nowadays research and development, because more and more applications find their place in people’s life: Fingerprints to login on your OS, to get in your workout center or to start your car engine aren’t rare anymore. On higher (security) levels scans of the eye are used. Cameras are present on many public places to improve security, and this works fine with motion detectors if it is not necessary to identify the person on the picture automatically.

Overlooking the fact that not everybody agrees with the presence of the cameras it is a difficult problem to detect and recognize faces on given camera images. There are a lot of approaches to this problem, which have all advantages and disadvantages.

The aim of this project is to get an idea of some (simple) methods and algorithms, how faces can be detected in images and how they can be identified or matched with a given face database.

LITERATURE SURVEY

PyCharm

PyCharm is an integrated development environment (IDE) that help professional python be more productive be more confident and write better code. It comes in two version, PyCharm pro and PyCharm community, and supports the full python workflow out of the box in the latter, including web frameworks frontend technologies, databases, and scientific tooling.

PyCharm is developed by the Czech company JetBrains it is cross platform working on Microsoft windows macOS and Linux. PyCharm has a professional edition released under a proprietary license and a community edition released under Apache license PyCharm community edition is less extensive than the professional edition

**PyCharm community edition**

The community edition of PyCharm is free and built on open source. JetBrains developed and published this freeware for python code developers. It has full support for python, including code insight, debugging capabilities, testing and more. you are free to use it whenever, and wherever, you like including at work.

Its source code is part of the IntelliJ platform and can be found on GITHUB where you can also fork and modify it as you wish. We are allowed to use PyCharm community for commercial use

When PyCharm community with free edition people only receive the code inspector graphical debugger and test runner intuitive python editor navigation with re-factoring and VCS support. Additionally, the professional edition includes HTML, JS and SQL assistance while the community option only permits core python support.



**MySQL**

The MySQL server provides a database management system with querying and connectivity capabilities as well the ability to have excellent data structure and integration with many different platforms it can handle large databases reliably and quickly in high demanding and libraries administrative tools and many applications programming interfaces (API)s.

The most comprehensive set of advanced features management tools and technical support to achieve the highest levels of MySQL scalability security reliability and uptime.it reduces risk, cost, and complexity in developing deploying and managing business critical MySQL applications

MySQL server is relational data base management system, it is a software product with the primary function of storing and retrieving data as requested by other software application—which may run either on the same computer or on another computer across a network (including the Internet). Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.

**SYSTEM REQUIREMENT SPECIFICATION**

The purpose of System requirement specification is to produce the specification analysis of the task and also to establish complete information about the requirement, behavior and other constraints such as functional performance and so on.

**HARDWARE REQUIREMENTS**

The following are the minimum Hardware requirement for the development system

|  |  |
| --- | --- |
| Processor | INTEL i5 |
| RAM | 4 GB |
| Hard Disk | 250 HDD |

***Table 1: Hardware Requirements***

**SOFTWARE REQUIREMENT**

The following are the software requirement for the development system

|  |  |
| --- | --- |
| ***Operating System*** | ***Windows 8, Linux, Mac OS*** |
| ***Programming Language*** | ***Python*** |
| ***Integrated Development Environment (IDE)*** | ***PyCharm*** |
| ***Database*** | ***MySQL*** |

*Table 2: Software Requirements*

SYSTEM ANALYSIS

**PROBLEMS IN THE EXISTING SYSTEM:**

1. Face recognition system can be impacted by poor lighting or low image quality.

2. This system can be manipulated and hence threatens privacy.

3. Misuses can cause fraud and other crimes.

4. It has data vulnerabilities and can cause data/security piracy.

**PROPOSED SYSTEM:**

1. Suitable image quality to avoid any negative impact on the face recognition system.
2. Cannot be misused as only details such as gender, college register number and combination are provided rather than any other personal details.
3. No data vulnerabilities as there are no hidden data but the details of the user displayed at the time of login or register.
4. No complex operations: the user can login or register easily as it only for the purpose of attendance and doesn’t require many details about the user.

**ADVANTAGES**

1. The software can be used for security purpose in organizations and in secured zones.
2. The system is convenient and secure for the users.
3. It saves their time and efforts.

**SYSTEM IMPLEMENTATION**

There is one main modules in this project

1.Register New User

**Functionalities of Register New User:**

1.The new users must use the ‘register new user’ module first.

2.Once they register their details will be stored in the database.

3.To register the user must provide with a face id, which will be recognized by the system each time the user logs in.

4.After face id the user must provide a username that will be displayed during the login session.

5.Users who have already registered can use login module.

6.The login module will recognize the user once their face is detected and display their username.

7.The login time and username will be stored in the database.

8.If the detected face is not recognized by the system the login will not be successful.

**FEASIBILITY ANALYSIS**

All projects are feasible – given unlimited resources and infinite time! Unfortunately, the development of computer-based system or product is more likely plagued by a scarcity of resources and difficult delivery dates. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. Months or years of effort, thousands or millions of dollars, and untold professional embarrassment can be averted if an ill-conceived system is recognized early in the definition phase.

Feasibility and risk analysis are related in many ways. If project risk is great the feasibility of producing quality software is reduced. During product engineering, however, we concentrate our attention on four primary areas of interest.

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

**TECHNICAL FEASIBILITY:**

Technical feasibility is carried out to determine whether the project is feasible in terms of software hardware personnel and expertise to handle the completion of the project. it considers determining resources for the proposed system

As the system is developed using python it is platform independent therefore the users of this system can have average processing capabilities running on any platform the technology is one the latest hence the system is also technically feasible

**ECONOMIC FEASIBILITY:**

Economic feasibility defines whether the expected benefit equals or exceeds analysis the excepted costs it also commonly referred to as cost benefit analysis. the procedure is to determine the benefits and the saving expected from the system and compare them with costs a proposed system is expected to out weight the cost

This is small project with no cost for development. The system is easy to understand and use therefore there is no need to spend on training to use the system. This system has the potential to grow by adding functionalities for students. This can hence the project could have economic benefits in the future

**OPERATIONAL FEASIBILITY:**

Operational feasibility is the measure of how well a proposed system solves the problems with the users. Operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented the projects is operationally feasible for the users as nowadays almost all the students are familiar

**REQUIREMENT SPECIFICATION**

One of the most important activities in software development is preparation of Software Requirement Specification (SRS). Since the problems in the modern world are becoming more and more complex, it is becoming increasingly difficult for the developers to comprehend the problems fully and work exactly according to the predicted goal all through the work, hence the need for a more rigorous requirement analysis. In the present time the analysis phase is considered to be most critical and difficult.

**Requirement Analysis**

A software requirements specification (SRS) is a complete description of the behavior of the system to be developed. SRS should describe the general factors that affect the product and its requirements. The purpose of this Software Requirement Specification (SRS) Document is to specify the user goals and tasks that need to be achieved. It must also include detailed description of the context and requirements – both functional and non-functional, which are vital to the successful completion of the project. Apart from these, the SRS Document incorporates in itself the constraints and assumptions made during the course of the project. Requirements must be measurable, testable, related to identified needs or opportunities, and defined to a level of detail sufficient for system design

**Purpose**

The aim of the software requirement specification document is to list out the user requirement in an organized manner. It defines all the constraint and software requirements needed to understand this application and documentation. It also has to give the overall design plan.

The use should be able to understand the proposed system after going through the SRS document and should be in a position to incorporate some changes if required. Further it should be used as the theme of software design and development.

**Scope**

Software requirement specification is the only written document that describes the requirements of the system. It is meant for use by developers and will be the basis for validating the final delivered system. Any changes made to the requirements in the future will have to go through a formal change approval process. The developer is responsible for clarification, whatever necessary, and will not make any alternatives without the permission of the client. It allows the user to analyze the future enhancements in the system.

**Functional Requirements**

**Functionalities of login**

1.Users who have already registered can use login module.

2.The login module will recognize the user once their face is detected and display their username.

3.The login time and username will be stored in the database.

4.If the detected face is not recognized by the system the login will not be successful

**Functionalities of register new user**

1.The new users must use the ‘register new user’ module first.

2.Once they register their details will be stored in the database.

3.To register the user must provide with a face id, which will be recognized by the system each time the user logs in.

4.After face id the user must provide a username that will be displayed during the login session.

**Non –Functional Requirement:**

### 

### Availability

The Availability of a system is defined as any-time access to the system. Enabling the application with proper session fail over and eliminating Single point of failure.

### 

### Security

Security is a major concern in the Web application. This Project will provide the authorization mechanism to ensure the access to the business services is provided to authorize personnel.

* **Flexibility:**

As it is reliable, efficient, it’s also flexible for the customer to use it.

* **Reliability:**

As the application is built in Visual Studio 2012 technology, it is more reliable and robust. Reliability of a system is defined as time extent to which the system can be operational without introducing performance degradation. Though asp.net itself ensures the proper garbage collection, this project would ensure clean up the objects in session as soon as the usage is over which can be implemented through a unified mechanism of session data handling to ensure the reliability aspect of the system.

* **Usability:**

The interface is user friendly and easy to access and easy to understand for new user as well.

* **Efficiency:**

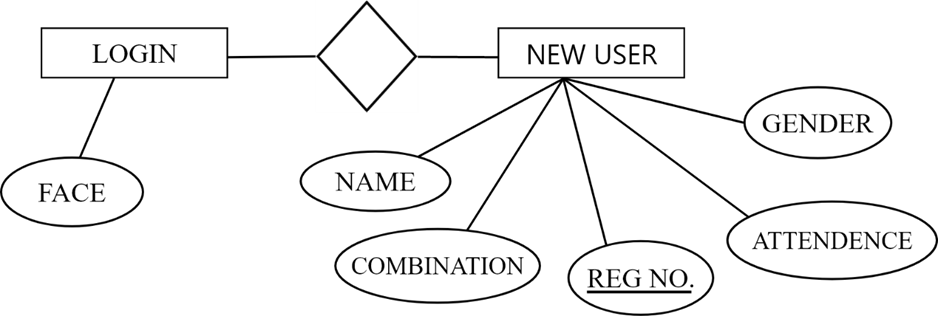
As the application doesn’t require so much of memory it is quite efficient and can be stored easily in minimal amount of disk space.

* **Maintainability:**

Maintaining the application is not tedious as the whole application is based on technologies readily available and very potent. Maintainability of the system is defined as the ease with which a software system or component can be modified to improve performance, or other attributes, or adapt to a changed environment.

**SYSTEM DESIGN**

**ER Diagrams**

****

LOGIN

**Data Flow Diagrams**

New

User

Register

Login/Face

Detect

Attendance

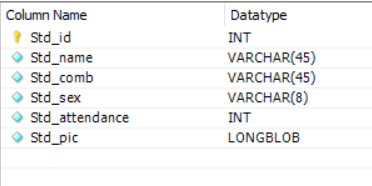
Count

Generate

Report

User Details

**Tables**



**CODING**

*# REQUIRED LIBRARIES*import io  
import os.path  
import datetime  
import subprocess  
import tkinter as pk  
from tkinter import ttk  
from tkinter import \*  
import util  
import cv2  
from PIL import Image, ImageTk  
import mysql.connector  
import base64  
  
  
*# MAIN CLASS*class App:  
 *# MAIN WINDOW IS THE START-UP INTERFACE* def \_\_init\_\_(self):  
  
 self.main\_window = pk.Tk()  
 self.main\_window.geometry("1920x1080+0+0")  
 self.main\_window.title("main app")  
 self.l\_btn = util.bttn(self.main\_window, 600, 200, './Images/img1.png', "./Images/img2.png", self.login1)  
 self.r\_btn = util.bttn(self.main\_window, 580, 380, './Images/img3.png', "./Images/img4.png", self.register\_new\_user)  
  
 *# FUNCTION OF LOGIN BUTTON IN MAIN WINDOW* def login1(self):  
  
 self.login\_capture\_window = pk.Toplevel(self.main\_window)  
 self.login\_capture\_window.geometry("1920x1080+0+0")  
 self.l1\_btn = util.bttn(self.login\_capture\_window, 900, 320, './Images/img5.png', "./Images/img6.png", self.login)  
 self.pic\_label = util.get\_img\_label(self.login\_capture\_window)  
 self.pic\_label.place(x=100, y=100, width=700, height=500)  
 self.add\_webcam(self.pic\_label)  
  
 *# THIS MODULE IS USED TO START WEBCAM* def add\_webcam(self, label):  
 if 'cap' not in self.\_\_dict\_\_:  
 self.cap = cv2.VideoCapture(0)  
 self.\_label = label  
 self.process\_webcam()  
  
 *# THIS MODULE IS USED TO PROCESS IMAGES CAPTURED BY THE WEBCAM AND DISPLAY IT* def process\_webcam(self):  
 ret, frame = self.cap.read()  
 self.most\_recent\_capture\_arr = frame  
 img\_ = cv2.cvtColor(self.most\_recent\_capture\_arr, cv2.COLOR\_BGR2RGB)  
 self.most\_recent\_capture\_pil = Image.fromarray(img\_)  
  
 imgtk = ImageTk.PhotoImage(image=self.most\_recent\_capture\_pil)  
 self.\_label.imgtk = imgtk  
 self.\_label.configure(image=imgtk)  
  
 self.\_label.after(20, self.process\_webcam)  
  
 *# THIS MODULE IS TO CONNECT DATABASE TO THIS PROJECT* global my\_db  
 global mycursor  
 my\_db = mysql.connector.connect(  
 host='Localhost',  
 user='root',  
 password='9448292290@moni.com',  
 port='3306',  
 database='dev\_db'  
 )  
  
 mycursor = my\_db.cursor()  
  
 *# LOGIN MODULE* def login(self):  
 *# THIS SET OF CODE IS TO CREATE FILE DIRECTORY TO MAINTAIN PHOTOS OF USERS* self.db\_dir = './db'  
 if not os.path.exists(self.db\_dir):  
 os.mkdir(self.db\_dir)  
 self.log\_path = 'log.xls' *# THIS APPENDS THE DETAILS OF LOGGED-IN USERS* unknown\_img\_path = './.tmp.jpg' *# THIS SAVES THE RECENT FRAME OF PICTURES CAPTURED BY THE WEBCAM* cv2.imwrite(unknown\_img\_path, self.most\_recent\_capture\_arr)  
 output = str(subprocess.check\_output(['face\_recognition', './db', unknown\_img\_path]))  
 name = output.split(',')[1][:-5] *# THIS REMOVES UNWANTED CHARACTERS LIKE '\n' AND '/d' FROM VARIABLE OUTPUT* global name1  
 name1 = name  
  
 if name in ['unknown\_person', 'no\_persons\_found']:  
 util.msg\_box('Oops...', 'User not found')  
 self.login\_capture\_window.destroy()  
 else:  
 util.msg\_box('welcome', 'Welcome ,{}'.format(name))  
 self.login\_capture\_window.destroy()  
 self.retrive()  
 self.time = datetime.datetime.now()  
 with open(self.log\_path, 'a') as f:  
 f.write('NAME:{}, DATE: {}, TIME: {}\n'.format(name, self.time.date(), self.time.time()))  
 f.close()  
 self.login\_capture\_window.destroy()  
  
 def destroy(self):  
 self.display.destroy()  
  
 *# THIS RETRIEVES DATA FROM DATABASE BY NAME RECOGNISED BY THE FACE-DETECTOR* def retrive(self):  
  
 self.display = pk.Toplevel(self.main\_window)  
 self.display.geometry("1920x1080+0+0")  
 my\_name = (name1,)  
 att\_inc = "UPDATE attendance SET Std\_attendance=Std\_attendance+1 where Std\_name=%s"  
 mycursor.execute(att\_inc, my\_name)  
 my\_db.commit()  
 my\_data = (name1,)  
 my\_row = "SELECT \* FROM attendance WHERE Std\_name=%s"  
 mycursor.execute(my\_row, my\_data)  
 student = mycursor.fetchmany(1)  
 for user in student:  
 global a, b, c, d, e  
 a = user[0]  
 b = user[1]  
 c = user[2]  
 d = user[3]  
 e = user[4]  
 my\_data = (name1,)  
 query = "SELECT Std\_pic FROM attendance WHERE Std\_name=%s"  
 mycursor.execute(query, my\_data)  
 data = mycursor.fetchall()  
 with open('image.jpg', 'wb') as fout:  
 fout.write(data[0][0])  
 self.image = data[0][0]  
 self.binary\_data = base64.b64decode(self.image)  
  
 self.my\_image = ImageTk.PhotoImage(Image.open(io.BytesIO(self.binary\_data)))  
 self.display\_pic = pk.Label(self.display, image=self.my\_image)  
 self.display\_pic.place(x=200, y=100)  
 if os.path.exists('image.jpg'):  
 os.remove('image.jpg')  
  
 *# THIS SET OF CODE IS TO OUTPUT THE DETAILS FROM DATABASE OF RECOGNISED PERSON* self.id\_lab\_db = util.get\_text\_label(self.display, 'REGISTER ID :')  
 self.id\_lab\_db.place(x=938, y=150)  
 self.disp\_a = util.db\_label(self.display, a)  
 self.disp\_a.place(x=1170, y=150)  
 self.name\_lab\_db = util.get\_text\_label(self.display, 'STUDENT NAME :')  
 self.name\_lab\_db.place(x=900, y=200)  
 self.disp\_b = util.db\_label(self.display, b)  
 self.disp\_b.place(x=1170, y=200)  
 self.comb\_lab\_db = util.get\_text\_label(self.display, 'COMBINATION :')  
 self.comb\_lab\_db.place(x=920, y=250)  
 self.disp\_c = util.db\_label(self.display, c)  
 self.disp\_c.place(x=1170, y=250)  
 self.sex\_lab\_db = util.get\_text\_label(self.display, 'GENDER :')  
 self.sex\_lab\_db.place(x=980, y=300)  
 self.disp\_d = util.db\_label(self.display, d)  
 self.disp\_d.place(x=1170, y=300)  
 self.att\_lab\_db = util.get\_text\_label(self.display, 'ATTENDANCE :')  
 self.att\_lab\_db.place(x=920, y=350)  
 self.disp\_e = util.db\_label(self.display, e)  
 self.disp\_e.place(x=1170, y=350)  
  
 self.destroy = pk.Button(self.display, text="BACK", command=self.destroy, fg="white", bg="#141414", height=1,  
 width=10, font=('Times New Roman', 15))  
 self.destroy.place(x=1170, y=450)  
  
 self.db\_dir = './db'  
 if not os.path.exists(self.db\_dir):  
 os.mkdir(self.db\_dir)  
  
 *# CAPTURE WINDOW* def register\_new\_user(self):  
 self.register\_new\_user\_window = pk.Toplevel(self.main\_window)  
 self.register\_new\_user\_window.geometry("1920x1080+0+0")  
 self.capt\_btn = util.bttn(self.register\_new\_user\_window, 920, 320, './Images/img7.png', "./Images/img8.png", self.snap\_pic)  
 *# CAPTURE* self.capture\_label = util.get\_img\_label(self.register\_new\_user\_window)  
 self.capture\_label.place(x=100, y=100, width=700, height=500)  
 self.add\_webcam(self.capture\_label)  
  
 *# THIS CLOSES THE PHOTO WINDOW WHEN TRY-AGAIN BUTTON IS CLICKED* def try\_again\_register\_new\_user(self):  
 self.photo\_window.destroy()  
  
 *# THIS IS TO GET DATA FROM RADIO BUTTON* def on\_select(self, value):  
 global gender  
 gender = str(value)  
  
 def unselect\_all(self):  
 self.var = pk.StringVar()  
 self.var.set(None)  
  
 *# HELPS TO PRINT IMAGE ON TO THE INTERFACE* def add\_img\_to\_label(self, label):  
 imgtk = ImageTk.PhotoImage(image=self.most\_recent\_capture\_pil)  
 label.imgtk = imgtk  
 label.configure(image=imgtk)  
 self.register\_new\_user\_capture = self.most\_recent\_capture\_arr.copy()  
  
 *# THIS USED TO GET DETAILS FROM THE USER* def snap\_pic(self):  
 self.photo\_window = pk.Toplevel(self.main\_window)  
 self.photo\_window.geometry("1920x1080+0+0")  
  
 self.snap\_label = util.get\_img\_label(self.photo\_window)  
 self.accept\_photo\_window = util.get\_button(self.photo\_window, 'ACCEPT', self.accept\_register\_new\_user, 'white', "#141414")  
 self.accept\_photo\_window.place(x=920, y=500)  
  
 self.try\_again\_photo\_window = util.get\_button(self.photo\_window, 'TRY AGAIN', self.try\_again\_register\_new\_user, 'white', "#141414")  
 self.try\_again\_photo\_window.place(x=920, y=600)  
  
 self.snap\_label.place(x=100, y=100, width=700, height=500)  
 self.add\_img\_to\_label(self.snap\_label)  
  
 self.name\_lab = util.get\_text\_label(self.photo\_window, 'STUDENT NAME :')  
 self.name\_lab.place(x=810, y=250)  
  
 self.std\_name = util.get\_entry\_text(self.photo\_window)  
 self.std\_name.place(x=1050, y=250)  
  
 self.id\_lab = util.get\_text\_label(self.photo\_window, 'REGISTER ID :')  
 self.id\_lab.place(x=845, y=300)  
  
 self.std\_id = util.get\_entry\_text(self.photo\_window)  
 self.std\_id.place(x=1050, y=300)  
  
 self.comb\_lab = util.get\_text\_label(self.photo\_window, 'COMBINATION :')  
 self.comb\_lab.place(x=830, y=350)  
  
 self.std\_comb = util.get\_entry\_text(self.photo\_window)  
 self.std\_comb.place(x=1050, y=350)  
  
 self.sex\_lab = util.get\_text\_label(self.photo\_window, 'GENDER :')  
 self.sex\_lab.place(x=895, y=415)  
 font = ('Times New Roman', 15)  
 self.var = pk.StringVar()  
 self.unselect\_all()  
 self.std\_sex = pk.Radiobutton(self.photo\_window, text="MALE", variable=self.var, value="MALE",  
 command=lambda: self.on\_select(self.var.get()), font=font)  
 self.std\_sex.place(x=1050, y=400)  
  
 self.std\_sex2 = pk.Radiobutton(self.photo\_window, text="FEMALE", variable=self.var, value="FEMALE",  
 command=lambda: self.on\_select(self.var.get()), font=font)  
 self.std\_sex2.place(x=1050, y=430)  
  
 self.text\_label = util.get\_text\_label(self.photo\_window, 'PLEASE ENTER YOUR DETAILS')  
 self.text\_label.place(x=920, y=170)  
 self.register\_new\_user\_window.destroy()  
  
 self.db\_dir = './db'  
 if not os.path.exists(self.db\_dir):  
 os.mkdir(self.db\_dir)  
  
 *# MAIN MODULE OF TKINTER* def start(self):  
 self.main\_window.mainloop()  
  
 *# THIS MODULE IS TO INSERT DETAILS ENTERED BY USER INTO DATABASE* def accept\_register\_new\_user(self):  
 self.db\_dir = './db'  
 if not os.path.exists(self.db\_dir):  
 os.mkdir(self.db\_dir)  
 name = self.std\_name.get(1.0, "end-1c")  
  
 cv2.imwrite(os.path.join(self.db\_dir, '{}.jpg'.format(name)), self.register\_new\_user\_capture)  
 *# TO NOTIFY USER THAT THE TEXTFIELD IS EMPTY* self.text\_fields = [self.std\_name, self.std\_id, self.std\_comb]  
 for i in range(len(self.text\_fields)):  
 self.content = self.text\_fields[i].get("1.0", "end").strip()  
  
 if len(self.content) == 0:  
 self.label\_notify = util.label\_colour(self.photo\_window, 'ENTER EVERY FIELDS', 'red')  
 self.label\_notify.place(x=270, y=600)

pic\_db1 = './.temp1.jpg'  
 cv2.imwrite(pic\_db1, self.most\_recent\_capture\_arr)  
 file = open('./.temp1.jpg', 'rb').read()  
  
 *# We must encode the file to get base64 string* file1 = base64.b64encode(file)s\_id = str(self.std\_id.get(1.0, "end-1c"))  
 comb = str(self.std\_comb.get(1.0, "end-1c"))  
  
 data = "INSERT INTO attendance (Std\_id, Std\_name, Std\_comb, Std\_sex, Std\_pic) VALUES (%s, %s, %s, %s, %s)"  
 val = (s\_id, name, comb, gender, file1)  
 mycursor.execute(data, val)  
 my\_db.commit()  
 util.msg\_box('Welcome', 'Successfully inserted into Database!')  
 self.photo\_window.destroy()  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app = App()  
 app.start()

**UTIL PACKAGE:**

import tkinter  
import tkinter as tk  
from tkinter import messagebox  
from PIL import Image, ImageTk  
from tkinter import ttk  
  
  
def get\_button(window, text, command, bcolor, fcolor):  
 def on\_enter(e):  
 button['background'] = '#ed2d2d'  
 button['foreground'] = bcolor  
  
 def on\_leave(e):  
 button['background'] = fcolor  
 button['foreground'] = bcolor  
  
 button = tk.Button(window,  
 text=text,  
 activebackground=fcolor,  
 activeforeground=bcolor,  
 fg=bcolor,  
 bg=fcolor,  
 command=command,  
 height=1,  
 width=20,  
 font=('Times', 25))  
 button.bind("<Enter>", on\_enter)  
 button.bind("<Leave>", on\_leave)  
  
 return button  
  
  
def bttn(window, x, y, image1, image2, command):  
 img\_a = ImageTk.PhotoImage(Image.open(image1))  
 img\_b = ImageTk.PhotoImage(Image.open(image2))  
  
 def on\_enter(e):  
 mybtn['image'] = img\_b  
  
 def on\_leave(e):  
 mybtn['image'] = img\_a  
  
 mybtn = tkinter.Button(window, image=img\_a, border=0, borderwidth=0, cursor='hand2', command=command, relief="flat")  
 mybtn.bind("<Enter>", on\_enter)  
 mybtn.bind("<Leave>", on\_leave)  
  
 mybtn.place(x=x, y=y)  
  
  
def get\_img\_label(window):  
 label = tk.Label(window)  
 label.grid(row=0, column=0)  
 return label  
  
  
def get\_text\_label(window, text):  
 label = tk.Label(window, text=text)  
 label.config(font=("Arial", 17), justify="left", borderwidth=0, relief="flat")  
 return label  
  
  
def label\_colour(window, text, fg):  
 label = tk.Label(window, text=text, fg=fg)  
 label.config(font=("Arial", 17), justify="left", borderwidth=0, relief="flat")  
 return label  
  
  
def get\_entry\_text(window):  
 inputtxt = tk.Text(window,  
 height=1,  
 width=14, font=("Arial", 24))  
 return inputtxt  
  
  
def msg\_box(title, description):  
 messagebox.showinfo(title, description)

**SOFTWARE TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement. The testing process focuses on the logical intervals of the software ensuring that all statements have been tested and on functional interval is conducting tests to uncover errors and ensure that defined input will produce actual results that agree with the required results. Program level testing, modules level testing integrated and carried out.

Different types of testing are:

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually, and functional tests will be written in detail.

Test objectives

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

Features to be tested

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

* Valid Input: identified classes of valid input must be accepted.
* Invalid Input: identified classes of invalid input must be rejected.
* Functions:identified functions must be exercised.
* Output: identified classes of application outputs must be exercised.
* Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

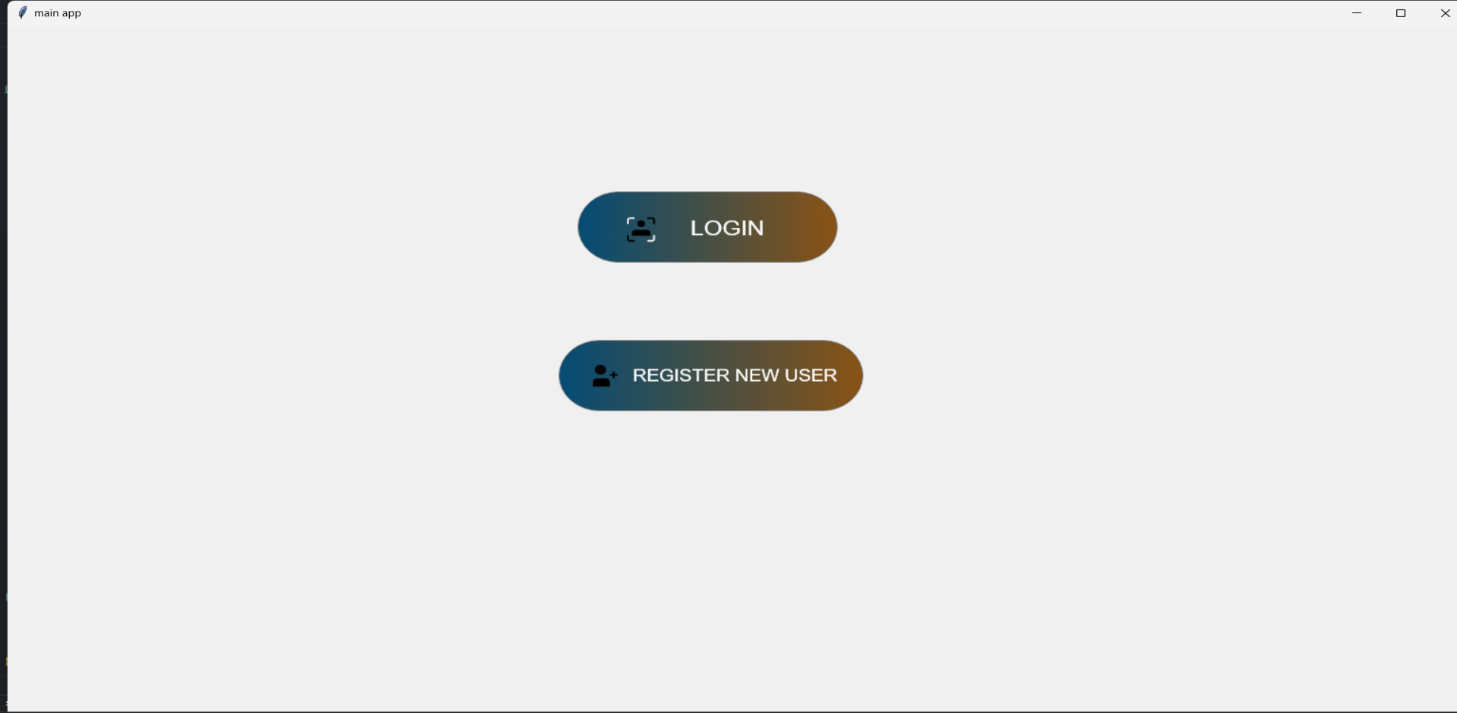
Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. You cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Validation test**

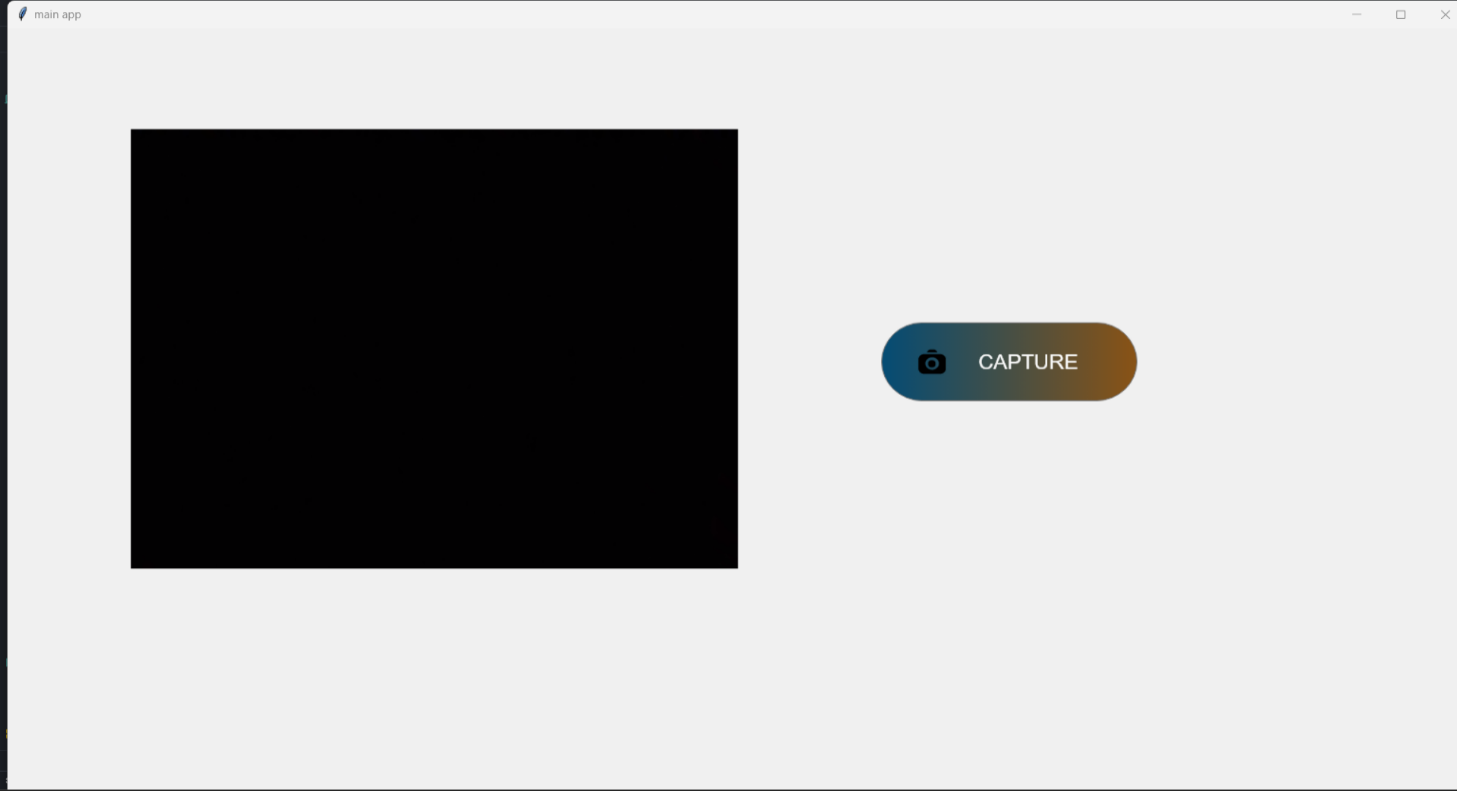
After the culmination of black box testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected and final series of software validation tests begin. Validation testing can be defined as many, but a single definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the customer. Validation refers to the process of using the software in a live environment to find errors. During the course of validation system may occur and the software will be changed.

**SNAPSHOT**

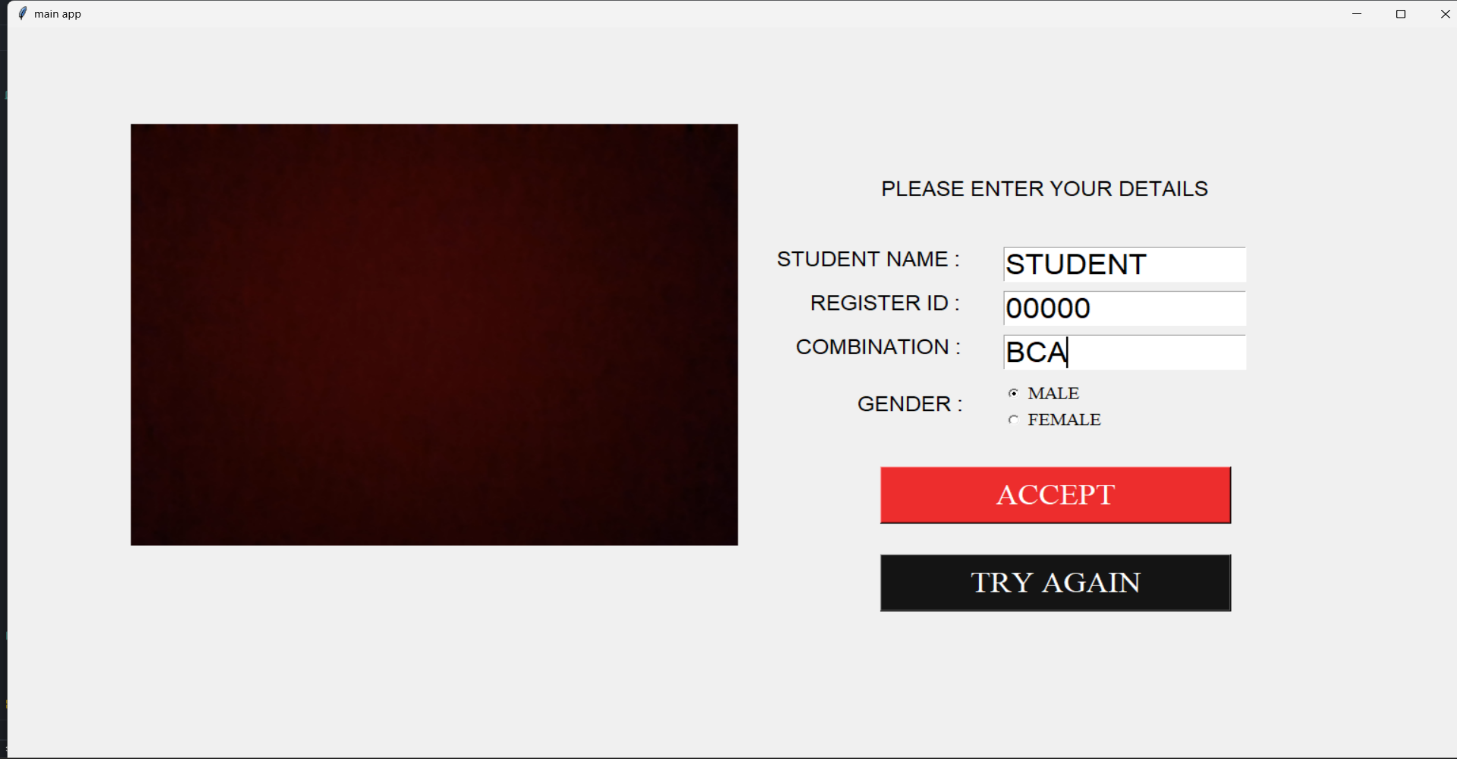
**MENU:**

****

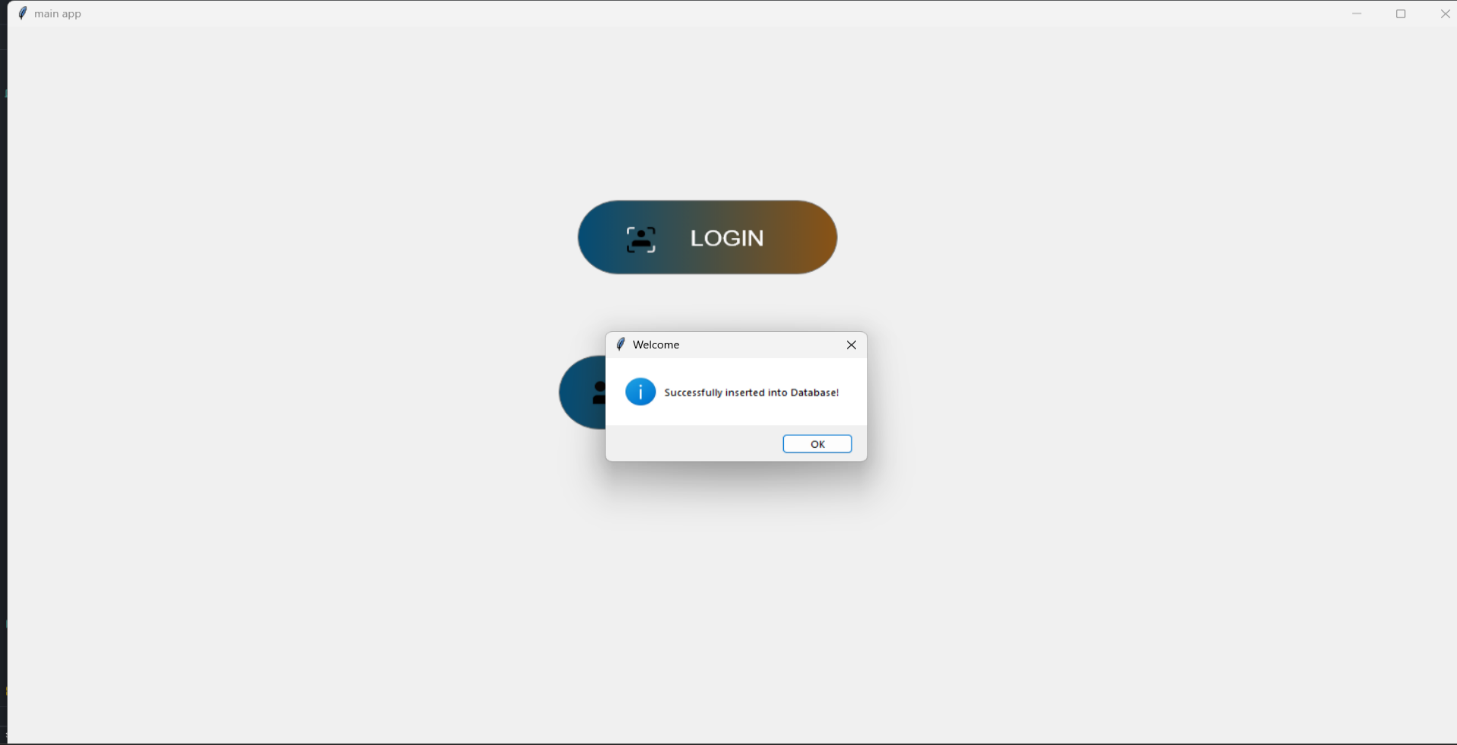
**REGISTER NEW USER:**

****

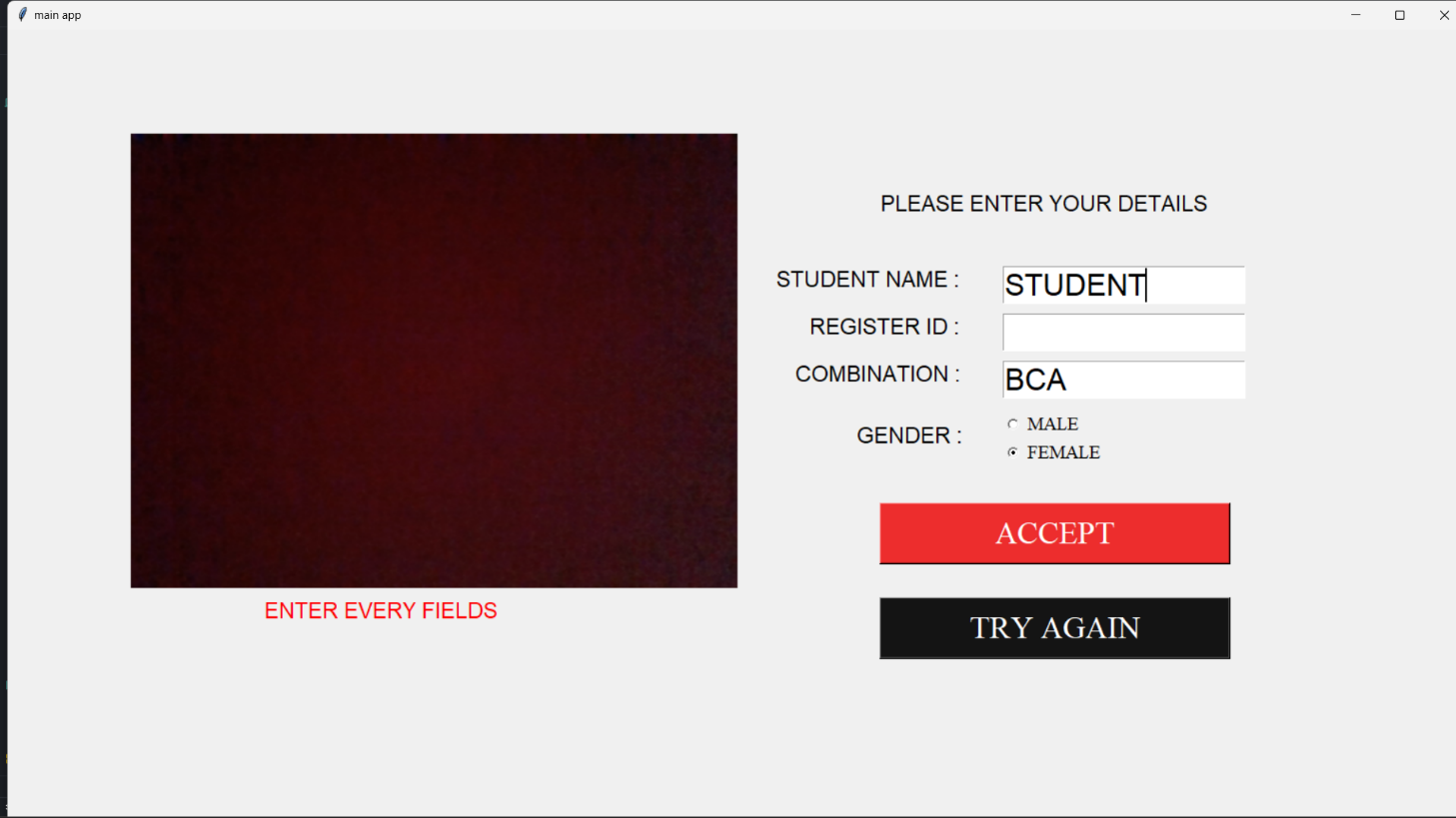
**USER DETAILS:**

****

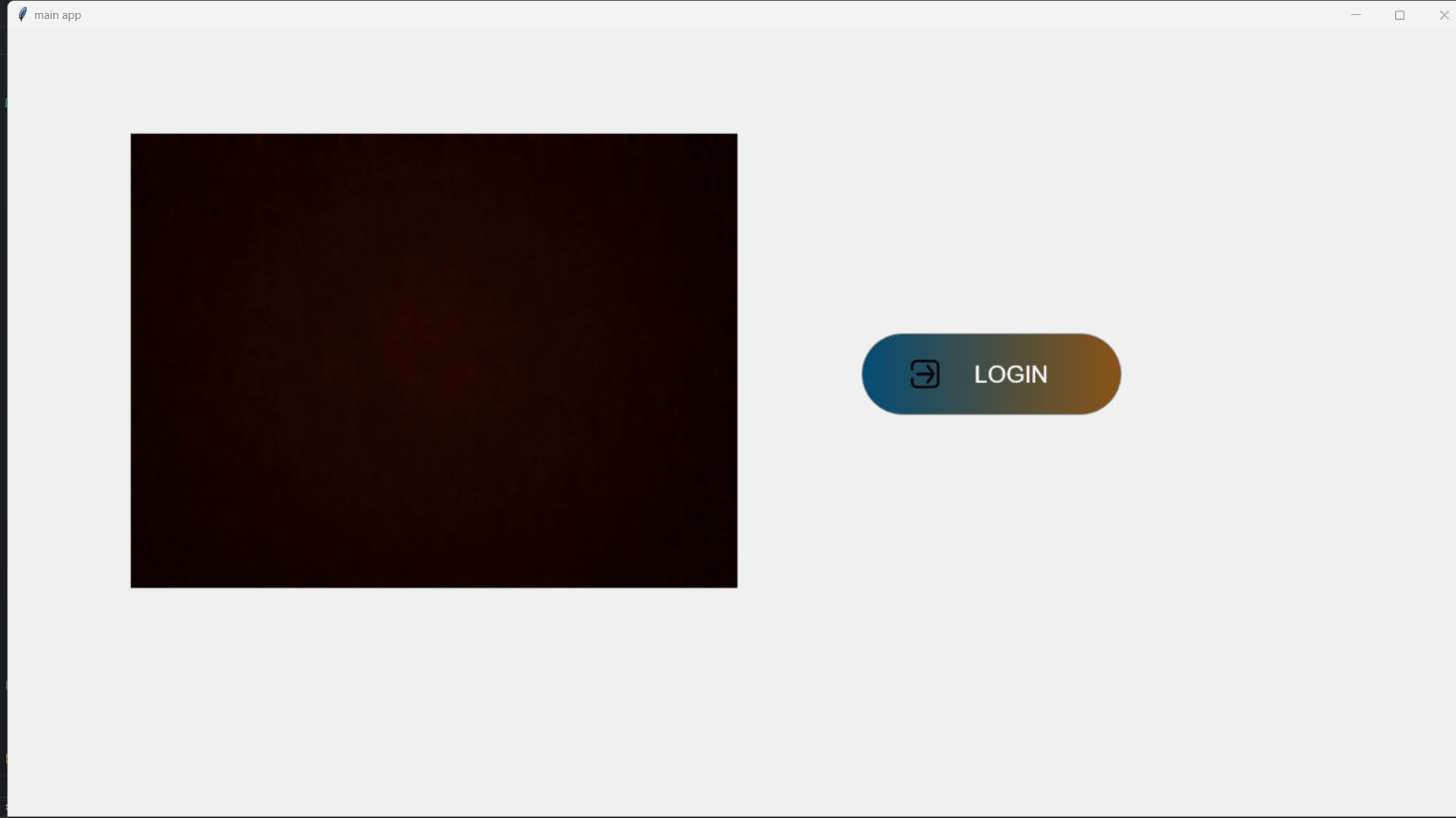
**REGISTER VALIDATION:**

****

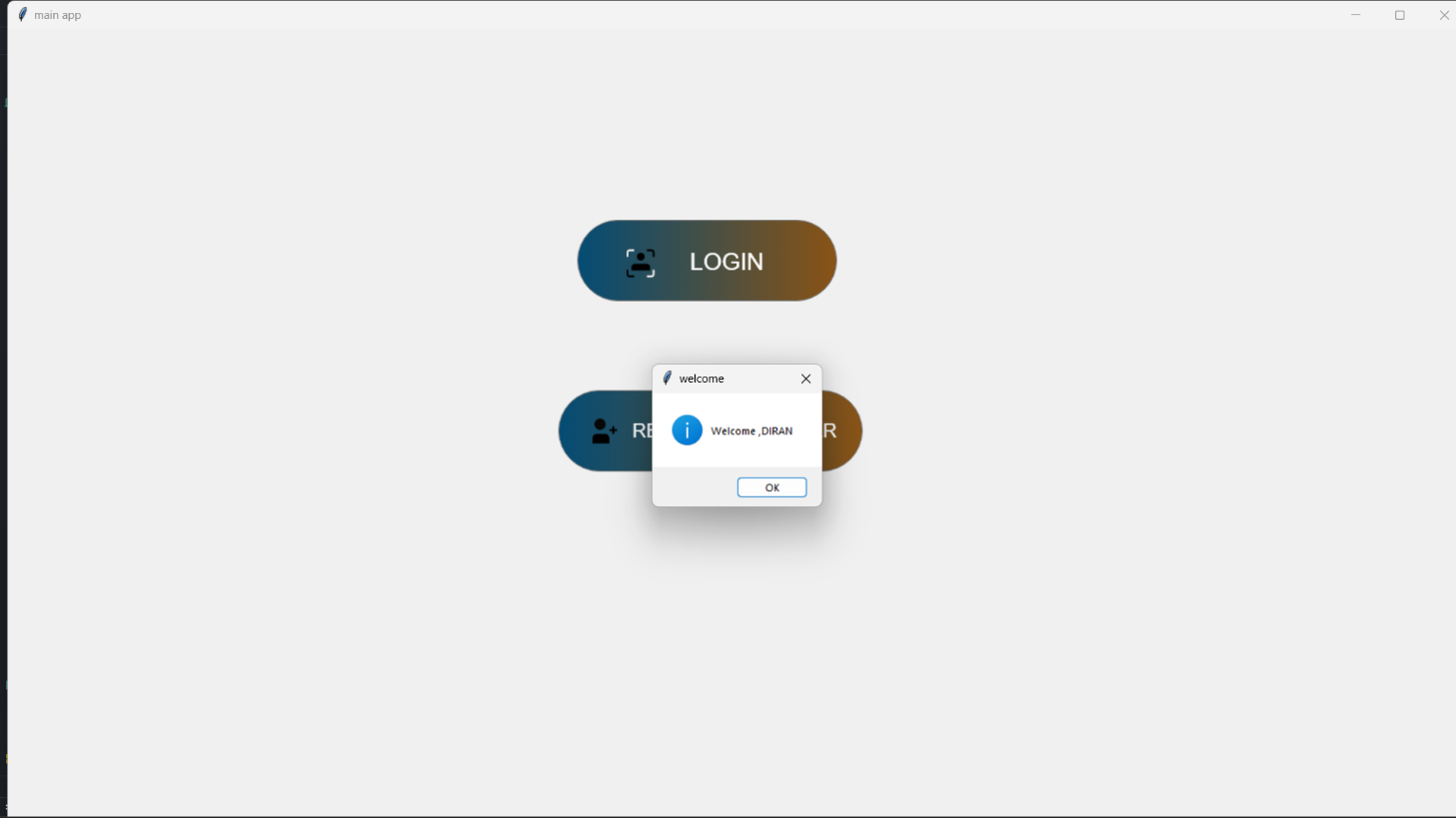
**INVALID REGISTER:**

****

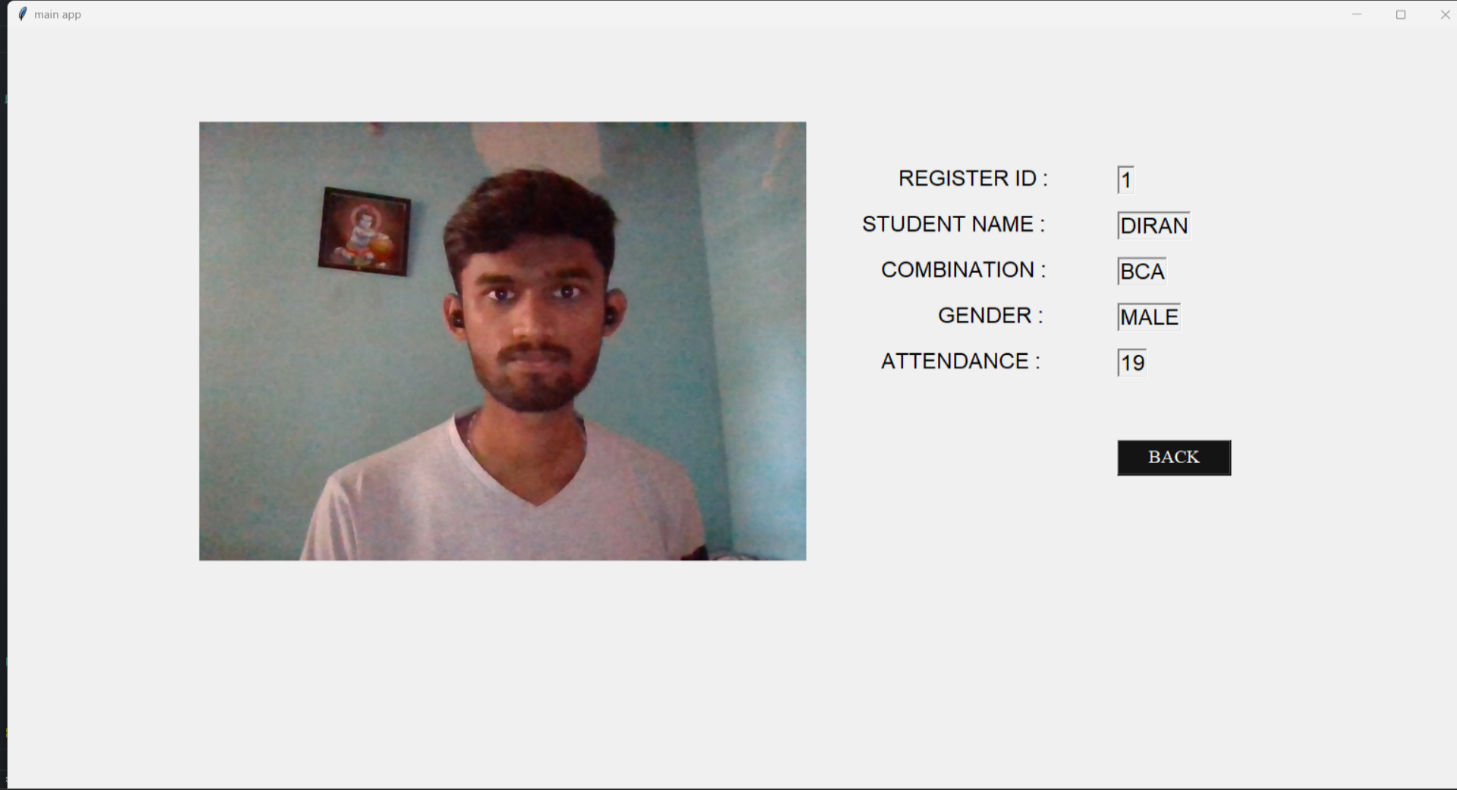
**LOGIN MENU:**

****

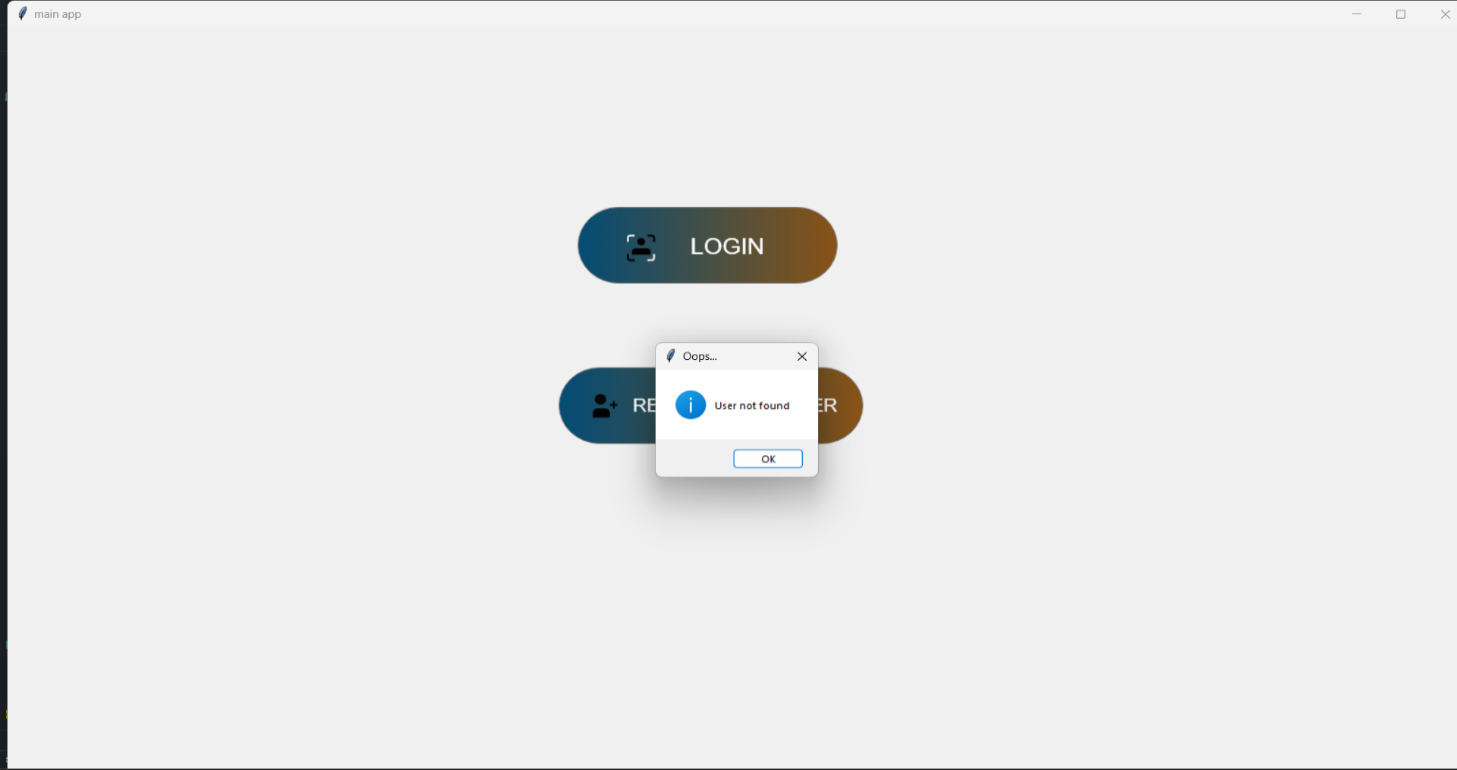
**LOGIN SUCCESSFUL:**

****

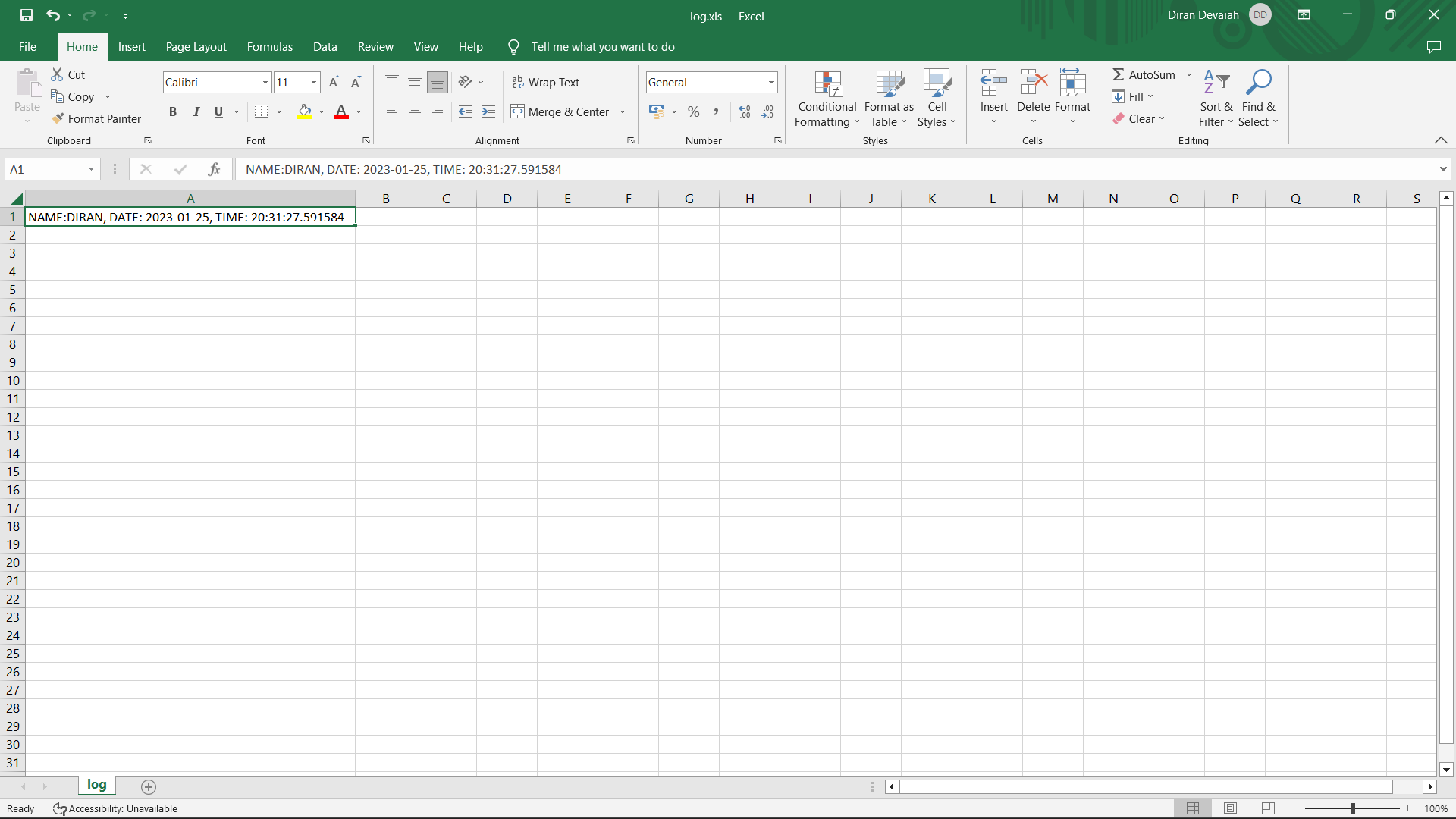
**USER DETAILS:**

****

**INVALID LOGIN:**

****

**ATTENDANCE IN EXCEL SHEET:**

****

**REFERENCE**

**Websites:**

[www.github.com/ageitgey/face\_recognition](http://www.github.com/ageitgey/face_recognition)

[www.geeksforgeeks.org/pythn-multiple-face-recognition-using-dlib/?ref=gcse](http://www.geeksforgeeks.org/pythn-multiple-face-recognition-using-dlib/?ref=gcse)

[www.medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78](http://www.medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78)

[www.medium.com/gitconnected/face-recognition-with-opencv-e93350876c84](http://www.medium.com/gitconnected/face-recognition-with-opencv-e93350876c84)

[www.geeksforgeeks.org/python-face-recognition-using-gui/?ref=gcse](http://www.geeksforgeeks.org/python-face-recognition-using-gui/?ref=gcse)

**Videos:**

[www.youtube.com/watch?v=z\_dbnYHAQYg&list=WL&index=7](http://www.youtube.com/watch?v=z_dbnYHAQYg&list=WL&index=7)

[www.youtube.com/watch?v=iBomaK2ARyI](http://www.youtube.com/watch?v=iBomaK2ARyI)

**Conclusion**

The purpose of this project is to ease the process of attendance system in educational institutions. This is due to the fact that the traditional attendance system consumes a lot of time and has to be performed on an hourly basis which can be an inconvenience. Moreover, it is a documentation on paper which is far off behind in the age of computers.

Therefore, our project enables the advancement of attendance system and has complete appropriate record of each and every user. This is convenient because it saves a lot of time compared to the traditional attendance system.

**Future enhancements:**

1. The facial recognition feature should not activate when displaying images of the user.
2. To allocate login time sessions according to the time schedule of the educational institution.
3. To implement a new “Log Out” feature.
4. To add modules like displaying results, student remarks and so on.