TITLE OF THE PROJECT

A

Mini Project Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

A. Abhiram 1602-21-737-066

Mirza Rafiq 1602-21-737-095

Md. Imran 1602-21-737-097

Department of Information Technology

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2020

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

We, **Abhiram ,Mirza, Imran** bearing hall ticket number, **1602-21-73-066**, **1602-21-737-095**, **1602-21-737-97** hereby declare that the project report entitled **"Online Admission System"** Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

A.Abhiram

1602-21-737-066

Mirza Rafiq Ahmed

1602-21-737-095

Mohammed Imran

1602-21-737-097

Dr .K. Ram Mohan Rao,

Professor & HOD

Dept. of IT.

(Faculty In-Charge)

ACKNOWLEDGEMENT

We extend our sincere thanks to Dr. S. V. Ramana, Principal, Vasavi College of Engineering for his encouragement. We express our sincere gratitude to Dr. K. Ram Mohan Rao, Professor & Head, Department of Information Technology, Vasavi College of Engineering, for introducing the Mini-Project module in our curriculum, and also for his suggestions, motivation, and co-operation for the successful completion of our Mini Project. We also want to thank and convey our gratitude towards our mini project coordinators **S.RENUKA**, **L.DIVYA, DAVID RAJU**, for guiding us in understanding the process of project development & giving us timely suggestions at every phase. We would also like to sincerely thank the project reviewers for their valuable inputs and suggestions.

Table of Content:

1. Abstract

The Online Admission System is an application designed to streamline the admission process for colleges. The system uses the Tkinter Python Library and the MySQL database to provide a user-friendly and efficient platform for both students and administrators. The system has two main modules: the administrator module and the user module. The administrator module includes the functionality to log in, modify college details, modify registered student details, and modify cut-off marks. The user module includes the functionality for registration, login, application for admission, viewing cut-off details, and viewing college details. The use case diagram and flowcharts were used to design the system, and the main modules, classes, and components were described in the implementation section. The system was tested to validate the cases, and the results were compared with existing systems. The additional learning gained from implementing this mini project was also discussed, along with future work and references.

2. Introduction

a.Problem Definition

The Online Admission System is an innovative solution aimed at simplifying and streamlining the admission process for colleges. In today's digital era, it is crucial for educational institutions to have an efficient and user-friendly system in place to manage the admission process. The traditional manual admission process can be time-consuming and prone to errors, leading to a poor student experience. The Online Admission System offers a solution to these problems .

b. Objective

The system uses the Tkinter Python Library and the MySQL database to provide a user-friendly interface and efficient data management. The system has two main modules: the administrator module and the user module. The administrator module provides the necessary tools for the college administrators to manage the admission process, including the ability to log in, modify college details, modify registered student details, and modify cut-off marks. The user module provides the students with the necessary tools to apply for admission, including the ability to register, log in, view cut-off details, and view college details.

3. Technology

Technology section in the report would include information about the software and hardware requirements for the implementation of the Online Admission System.

Software Requirements:

- Python programming language
- Tkinter library in Python for creating GUI interfaces
- MySQL database to store and retrieve data

Hardware Requirements:

- A computer or laptop with at least 2 GB of RAM and a processing speed of 2.0 GHz
- A stable internet connection for accessing the database

It is important to mention the technology used in the project as it gives a clear picture of the tools and platforms required for the implementation and also helps in future maintenance and upgradation of the system.

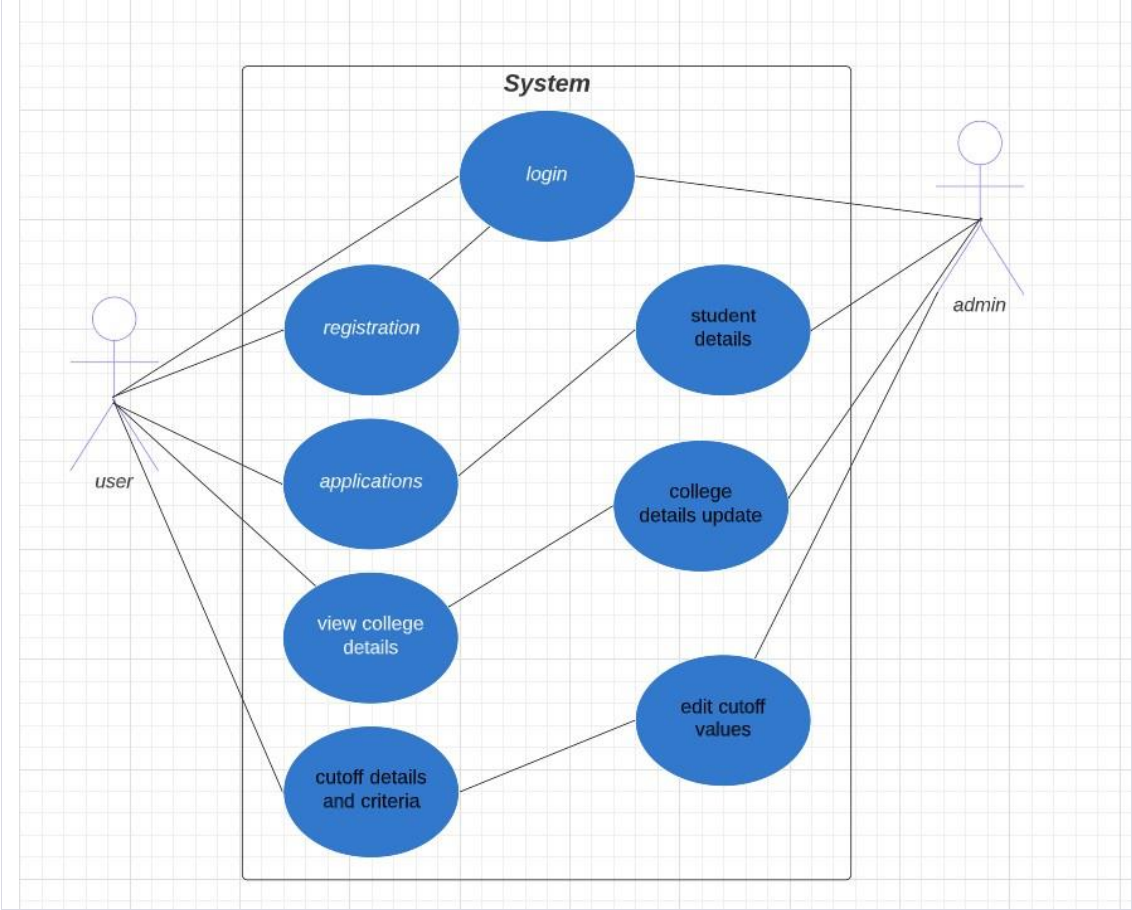
4. Proposed Work

The Proposed Work section of the report would consist of the following sub-sections:

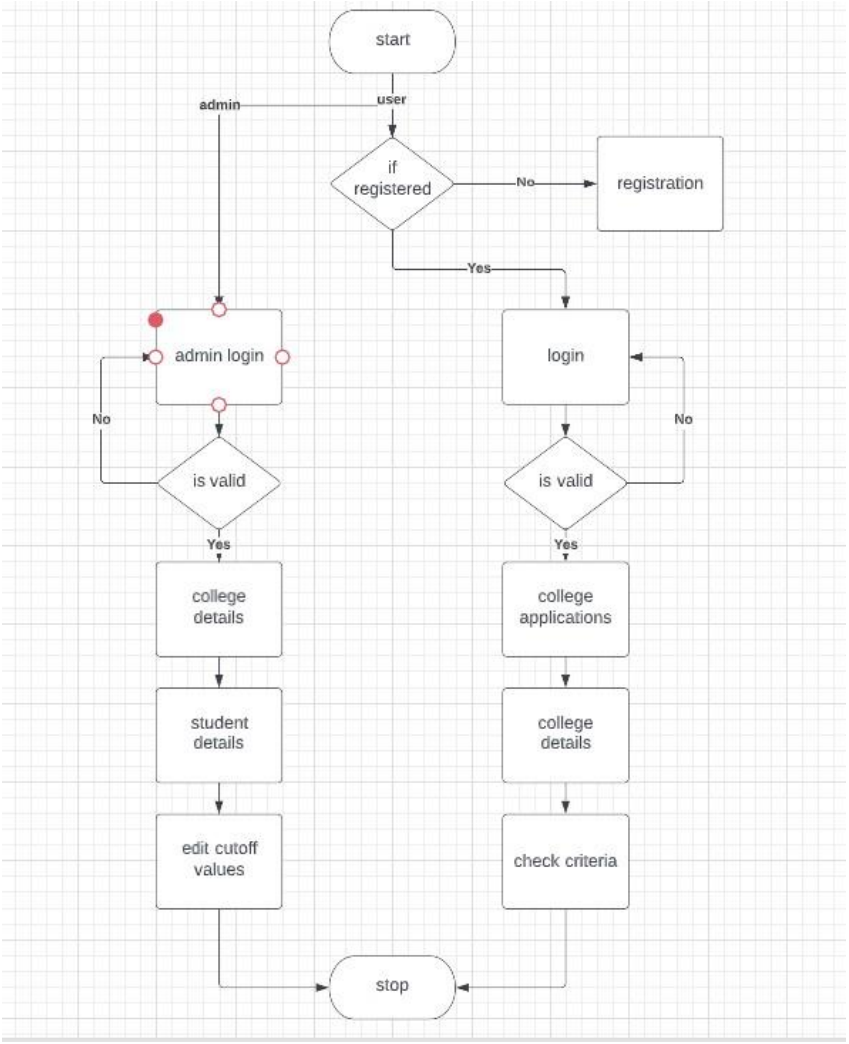
a. Design:

- Use case diagram: A visual representation of the interaction between the system and its users (Admin and User) and the flow of events in the system.
- Flowcharts: Detailed flowcharts explaining the steps involved in each module of the system, such as Registration, Login, Application for admission, etc.

UML Diagram:



Use Case Diagram:



b. Implementation:

- Description of main modules/classes/components: An explanation of the various modules or components used in the system, their functions, and their interactions with each other.
- Any specific algorithms/logic to be highlighted: An explanation of any complex algorithms or logics used in the system, along with their implementation and results.
- GitHub links and folder structure: Details about the source code repository, links to the code and a description of the folder structure used for organizing the code and related files.

```
from tkinter import *
import tkinter.messagebox
import os
import random
import time
import datetime
import login_db

def Login():
    u = Username.get()
    p = Password.get()
    cred = login_db.view()
    len_cred = len(cred)
    for i in range(0, len_cred):
        if u == cred[i][1]:
            if p == cred[i][2] and p == cred[i][3]:
                __menu__()
            elif p!=cred[i][2]:
                tkinter.messagebox.showerror("Login", "Error : Password Not Matching")
                Username.set("")
                Password.set("")
                text_Username.focus()
                break
        else:
            tkinter.messagebox.showerror("Error", "Error! Please try again")
            Username.set("")
            Password.set("")
            text_Username.focus()
            break

def Reset():
    Username.set("")
    Password.set("")
    text_Username.focus()
```

```
def Exit():
    Exit = tkinter.messagebox.askokcancel("Login System", "Confirm if you want to Exit")
    if Exit > 0:
        master.destroy()
        return

def __menu__():
    filename = 'Menu.py'
    os.system("python "+filename)
    #os.system('notepad'+filename)

def __Adminlogin__():
    filename = 'admin_login.py'
    os.system(filename)
    os.system('python '+filename)

def __Register__():
    filename = 'Register.py'
    #exec(open(filename).read())
    os.system('python '+filename)

master = Tk()


master.title("Student Login")
master.geometry('1400x1200')
master.config(bg = 'orange')
Frame = Frame(master, bg = 'orange')
Frame.pack()


Username = StringVar()
Password = StringVar()

Lbl_title = Label(Frame, text = 'PSG College of Technology', font = ('arial',40,'bold'), bg = 'orange', fg =
'Black')
Lbl_title.grid(row = 0, column = 0, columnspan =5, pady = 20)
Lbl_name = Label(Frame, text = 'College Admission System', font = ('arial',40,'bold'), bg = 'orange', fg =
'Black')
Lbl_name.grid(row = 1, column = 0, columnspan =5, pady = 20)

Login_Frame_1 = LabelFrame(Frame, width = 1350, height = 600, relief = 'ridge', bg = 'orange', bd = 15,
font = ('arial',20,'bold'))
Login_Frame_1.grid(row = 3, column =0, columnspan =5)
Login_Frame_2 = LabelFrame(Frame, width = 1000, height = 600, relief = 'ridge',bg = 'orange', bd = 15,
font = ('arial',20,'bold'))
Login_Frame_2.grid(row = 4, column = 0, columnspan =5)
```

```
Label_Username = Label(Login_Frame_1, text = 'Username', font = ('arial',20,'bold'), bg = 'orange', bd = 20)
Label_Username.grid(row = 0, column = 0)
text_Username = Entry(Login_Frame_1, font = ('arial',20,'bold'), textvariable = Username)
text_Username.grid(row = 0, column = 1, padx = 50)

Label_Password = Label(Login_Frame_1, text = 'Password', font = ('arial',20,'bold'), bg = 'orange', bd = 20)
Label_Password.grid(row = 1, column = 0)
text_Password = Entry(Login_Frame_1, font = ('arial',20,'bold'), show = '*', textvariable = Password)
text_Password.grid(row = 1, column = 1)


btnLogin = Button(Login_Frame_2, text = 'Login', width = 10, font = ('airia',15,'bold'), command = Login)
btnLogin.grid(row = 3, column = 0, padx = 8, pady = 20)

btnReset = Button(Login_Frame_2, text = 'Reset', width = 10, font = ('airia',15,'bold'), command = Reset)
btnReset.grid(row = 3, column = 1, padx = 8, pady = 20)

btnExit = Button(Login_Frame_2, text = 'Exit', width = 10, font = ('airia',15,'bold'), command = Exit)
btnExit.grid(row = 3, column = 2, padx = 8, pady = 20)

btnAdmLogin = Button(Login_Frame_2, text = 'Admin Login', width = 10, font = ('airia',15,'bold'), command = __Adminlogin__)
btnAdmLogin.grid(row = 4, column = 1, padx = 8, pady = 20)

btnRgstr= Button(Login_Frame_2, text = 'Register', width = 10, font = ('airia',15,'bold'), command = __Register__)
btnRgstr.grid(row = 4, column = 2, padx = 8, pady = 20)

master.mainloop()
```

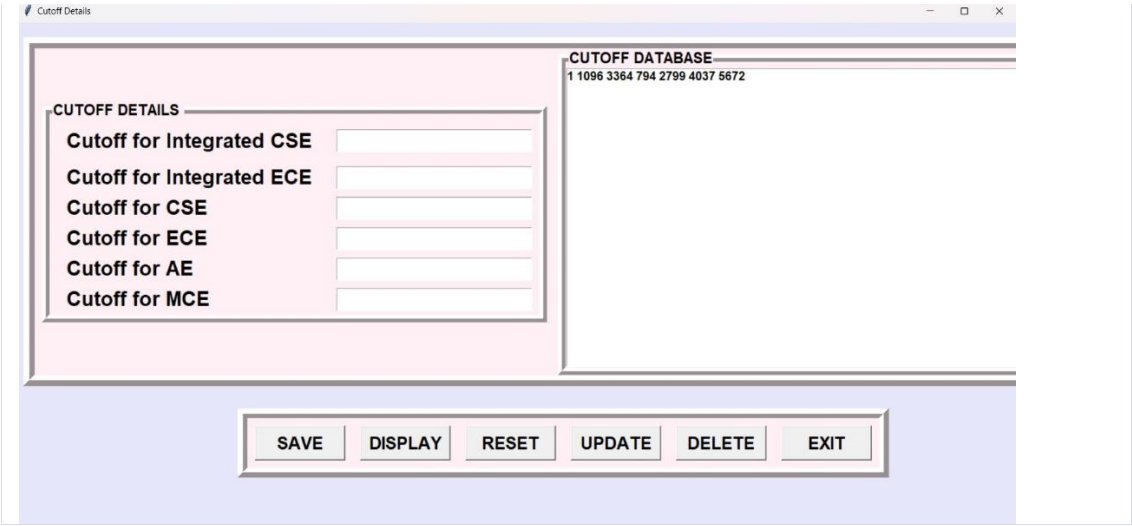
```
from tkinter import *
import tkinter.messagebox
import os
import random
import time
import datetime
```

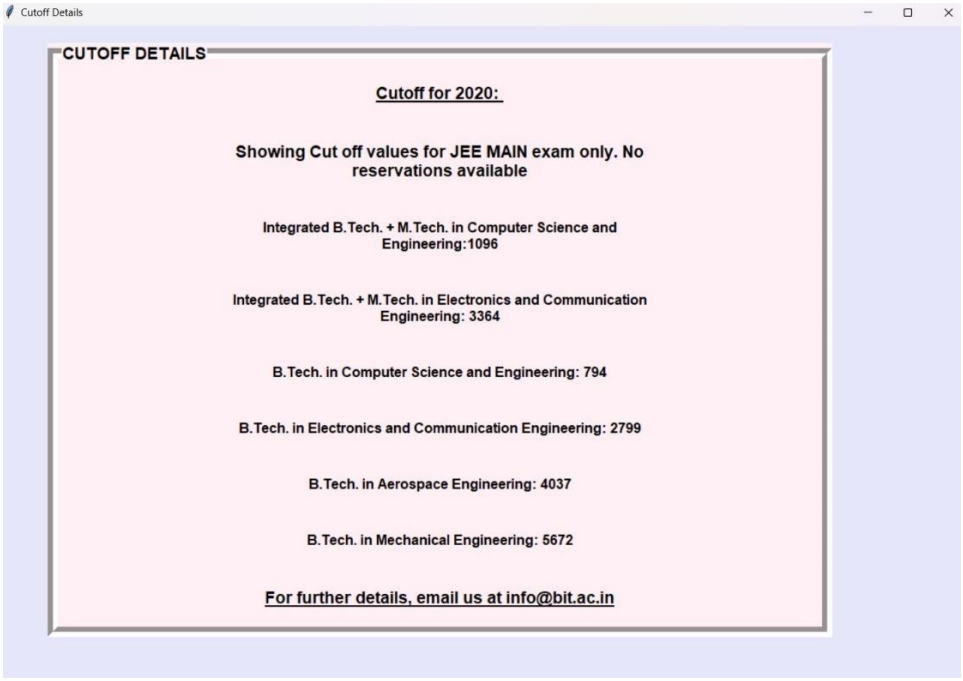
```
import sqlite3

def insert(id = 'NULL', username = ' ', password = ' ', confirm = ' '):
      con = sqlite3.connect('collegeadmn.db')
      cur = con.cursor()

      cur.execute('INSERT INTO register VALUES (NULL,?,?,?)',(username, password, confirm))

      con.commit()
      con.close()

def Add():
   if(len(Username.get())!= 0 and Password.get()==Confirm.get()):
      insert(username = Username.get(), password = Password.get(), confirm = Confirm.get())
      tkinter.messagebox.showinfo("Login", "Account Successfully Created")
      Exit()

   elif(Password.get()!=Confirm.get()):
      tkinter.messagebox.showerror("Login", "Error : Password Not Matching")
      Username.set("")
      Password.set("")
      Confirm.set("")
      text_Username.focus()
   else:
      tkinter.messagebox.showerror("Error", "Error! Please try again")
      Username.set("")
      Password.set("")
      Confirm.set("")
      text_Username.focus()

def Exit():
   Exit = tkinter.messagebox.askokcancel("Login System", "Confirm if you want to Exit")
   if Exit > 0:
      master.destroy()
      return

master = Tk()


master.title("Student Registration")
master.geometry('1400x1200')
master.config(bg = 'orange')
Frame = Frame(master, bg = 'orange')
Frame.pack()


Username = StringVar()
Password = StringVar()
Confirm = StringVar()
```

```
Lbl_title = Label(Frame, text = 'PSG College of Technology', font = ('arial',40,'bold'), bg = 'orange', fg = 'Black')
Lbl_title.grid(row = 0, column = 0, columnspan =5, pady = 20)
Lbl_name = Label(Frame, text = 'Student Registration', font = ('arial',40,'bold'), bg = 'orange', fg = 'Black')
Lbl_name.grid(row = 1, column = 0, columnspan =5, pady = 20)

Reg_Frame_1 = LabelFrame(Frame, width = 1350, height = 600, relief = 'ridge', bg = 'orange', bd = 15, font = ('arial',20,'bold'))
Reg_Frame_1.grid(row = 3, column =0, columnspan =5)
Reg_Frame_2 = LabelFrame(Frame, width = 1000, height = 600, relief = 'ridge',bg = 'orange', bd = 15, font = ('arial',20,'bold'))
Reg_Frame_2.grid(row = 4, column = 0, columnspan =5)

#USERNAME AND PASSWORD
Label_Username = Label(Reg_Frame_1, text = 'Username', font = ('arial',20,'bold'), bg = 'orange', bd = 20)
Label_Username.grid(row = 0, column = 0)
text_Username = Entry(Reg_Frame_1, font = ('arial',20,'bold'), textvariable = Username)
text_Username.grid(row = 0, column = 1, padx = 50)

Label_Password = Label(Reg_Frame_1, text = 'Password', font = ('arial',20,'bold'), bg = 'orange', bd = 20)
Label_Password.grid(row = 1, column = 0)
text_Password = Entry(Reg_Frame_1, font = ('arial',20,'bold'), show = '*', textvariable = Password)
text_Password.grid(row = 1, column = 1)

Label_Confirm = Label(Reg_Frame_1, text = 'Confirm Password', font = ('arial',20,'bold'), bg = 'orange', bd = 20)
Label_Confirm.grid(row = 2, column = 0)
text_Confirm = Entry(Reg_Frame_1, font = ('arial',20,'bold'), show = '*', textvariable = Confirm)
text_Confirm.grid(row = 2, column = 1)


btnReg = Button(Reg_Frame_2, text = 'Register', width = 10, font = ('airia',15,'bold'), command = Add)
btnReg.grid(row = 3, column = 0, padx = 8, pady = 20)

btnExit = Button(Reg_Frame_2, text = 'Exit', width = 10, font = ('airia',15,'bold'), command = Exit)
btnExit.grid(row = 3, column = 1, padx = 8, pady = 20)

master.mainloop()
```

c. Testing:

- Description of testing: An explanation of the testing process used to validate the system, including the various test cases and their results.


Cutoff Details
CUTOFF DETAILS
Cutoff for Integrated CSE
Cutoff for Integrated ECE
Cutoff for CSE
Cutoff for ECE
Cutoff for AE
Cutoff for MCE
CUTOFF DATABASE
1 1096 3364 794 2799 4037 5672
SAVE
DISPLAY
RESET
UPDATE
DELETE
EXIT


Cutoff Details
CUTOFF DETAILS
Cutoff for 2020:
Showing Cut off values for JEE MAIN exam only. No reservations available
Integrated B.Tech. + M.Tech. in Computer Science and Engineering:1096
Integrated B.Tech. + M.Tech. in Electronics and Communication Engineering: 3364
B.Tech. in Computer Science and Engineering: 794
B.Tech. in Electronics and Communication Engineering: 2799
B.Tech. in Aerospace Engineering: 4037
B.Tech. in Mechanical Engineering: 5672
For further details, email us at info@bit.ac.in

5.Result

The results of implementing the online admission system using the Tkinter python library and MySQL database were successful. The system was able to fulfill all the requirements as specified in the problem statement, which included modules for both the administrator and the user. The administrator was able to perform various tasks such as logging in, modifying college details, modifying registered student details, and modifying cut-off marks. The user was able to register, log in, apply for admission, view cut-off details, and view college details.

The system was tested for various scenarios to ensure that it was functioning as expected. The results of the testing showed that the system was able to perform all the tasks efficiently and effectively. The system was able to handle a large number of users and was able to provide the desired results in a timely manner.

The implementation of this system provided a valuable learning experience, as it involved the use of various technologies such as the Tkinter python library, MySQL database, and algorithms. The implementation of the system allowed the team to gain a deeper understanding of how these technologies can be used to build a complete and functional system.

6.Discussion and Future Work

The problem-solving aspect of the online admission system project lies in the design and implementation of the system to meet the specified requirements. This involved identifying the various components and modules required to build the system and how they would interact with each other. Additionally, the team had to consider various constraints such as performance, scalability, and security when designing and implementing the system.

The development of an online admission system using the Tkinter Python library and a MySQL database presents several opportunities for additional learning and problem-solving. Some of the key areas include:

1. GUI development using Tkinter: This project provides hands-on experience in developing graphical user interfaces using the Tkinter library.
2. Database integration: Working with a database (MySQL) helps to understand the concepts of database management, SQL queries, and data storage.
3. User authentication and authorization: Implementing user authentication and authorization ensures that sensitive information is protected and accessible only by authorized users.
4. Algorithms and logic: Designing algorithms and logic to handle various functionalities of the system, such as validating user input and processing admission applications, helps to develop problem-solving skills.Overall, the development of an online admission system provides a platform to apply and enhance programming, database, and web development skills while solving real-world problems.

7.Modules

admin_login : this module provides the functionality of admin login page.
admin_menu : frontend page of admin menu.
Clg_details : the working of frontend and backend of details page.
Collegeadmn: the database class for storing information related to login,cutoff,detail.
Login: the login page common to both user and admin.
Reg: the frontend and backend for the registration process.
Cutoff: for displaying the cutoff information and stroing it.

8.References:

- https://nevonprojects.com/online-college-admission-management-system-project/
- https://www.geeksforgeeks.org/python-tkinter-tutorial/
- https://www.w3schools.com/python/python_mysql_getstarted.asp
- https://pythongeeks.org/gui-programming-in-python/