Expt. No.

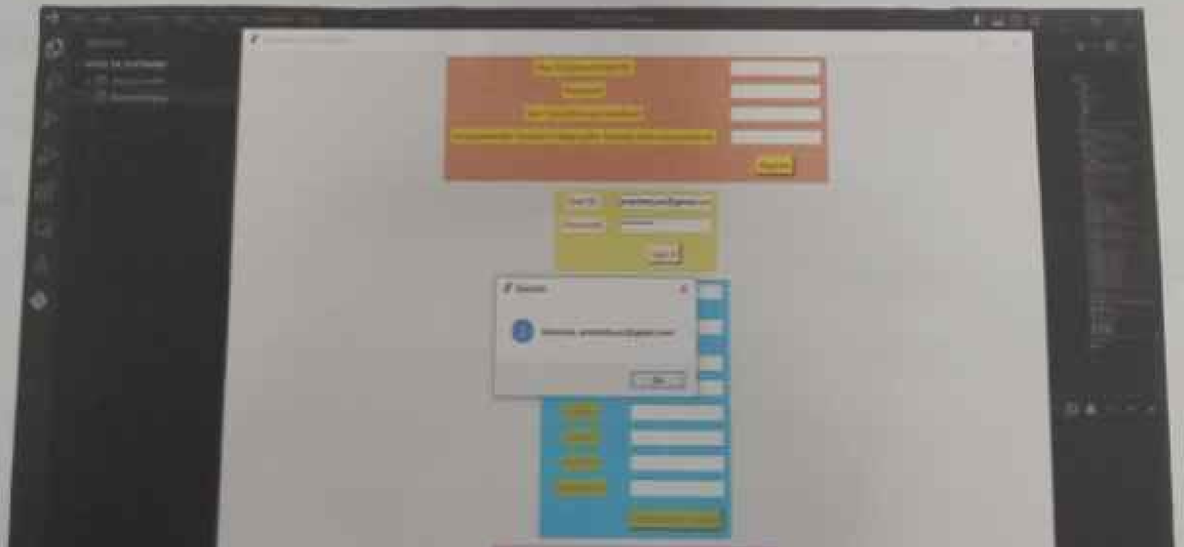
Orion
PAGE:
DATE: / /

INTRODUCTION

Python is a high-level general purpose programming language whose design philosophy emphasizes code readability with the use of significant indentation. It is dynamically typed and garbage collected. It supports multiple programming paradigms like structured (procedural), object-oriented and functional programming. Because of its ease of use, Python consistently ranks as one of the most popular programming languages. It is highly extensive via modules and libraries and one of its modules Tkinter has been used in this program to develop Graphical User Interface (GUI) in the frontend.

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit and is Python's standard GUI. It is implemented as a Python wrapper around a complete Tcl interpreter embedded in the Python interpreter. The main components of a Tkinter application are: Window, which is a rectangular area on the user's display screen, Widget, which refers to the building blocks of the GUI application, and Frames, which is the basic unit of organization and represents a section of the GUI.

Teacher's Signature.......................



Expt. No.

Orion
PAGE:
DATE: / /

LIBRARIES USED:

The libraries imported for the academic unit application are listed as follows:

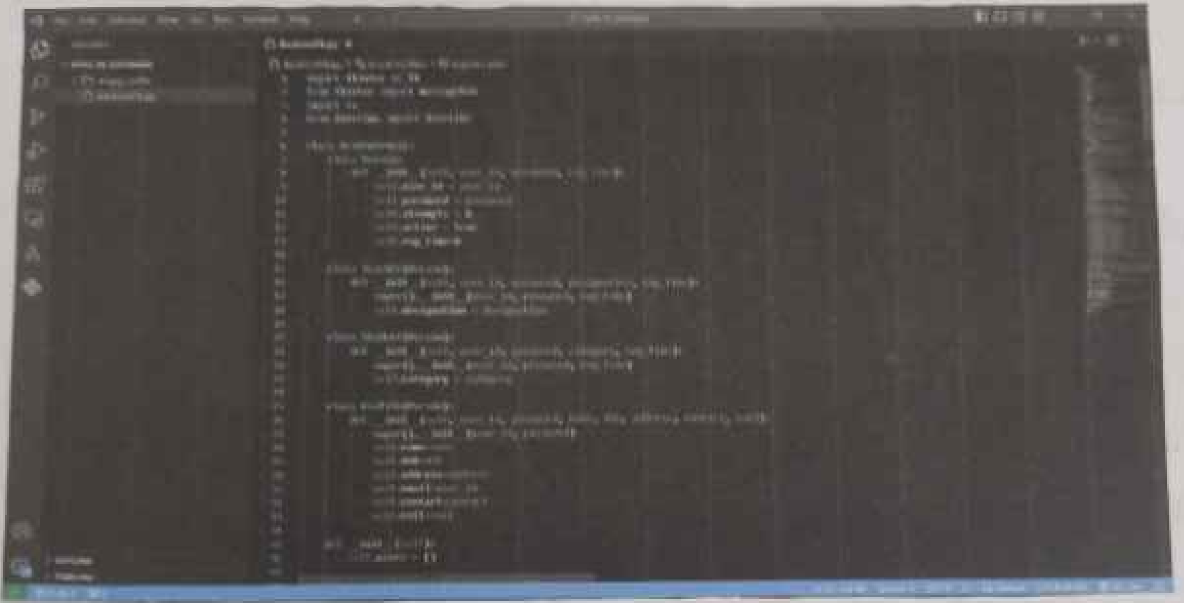
• Tkinter : For the GUI part of the application with which the user interacts.
 - Module messagebox was also imported from the tkinter library.

• Regular Expression (re) library: To carry out string manipulations at the backend. It has been used to check whether the password entered satisfies the password policy, and to check whether the category of a person and a student is valid.

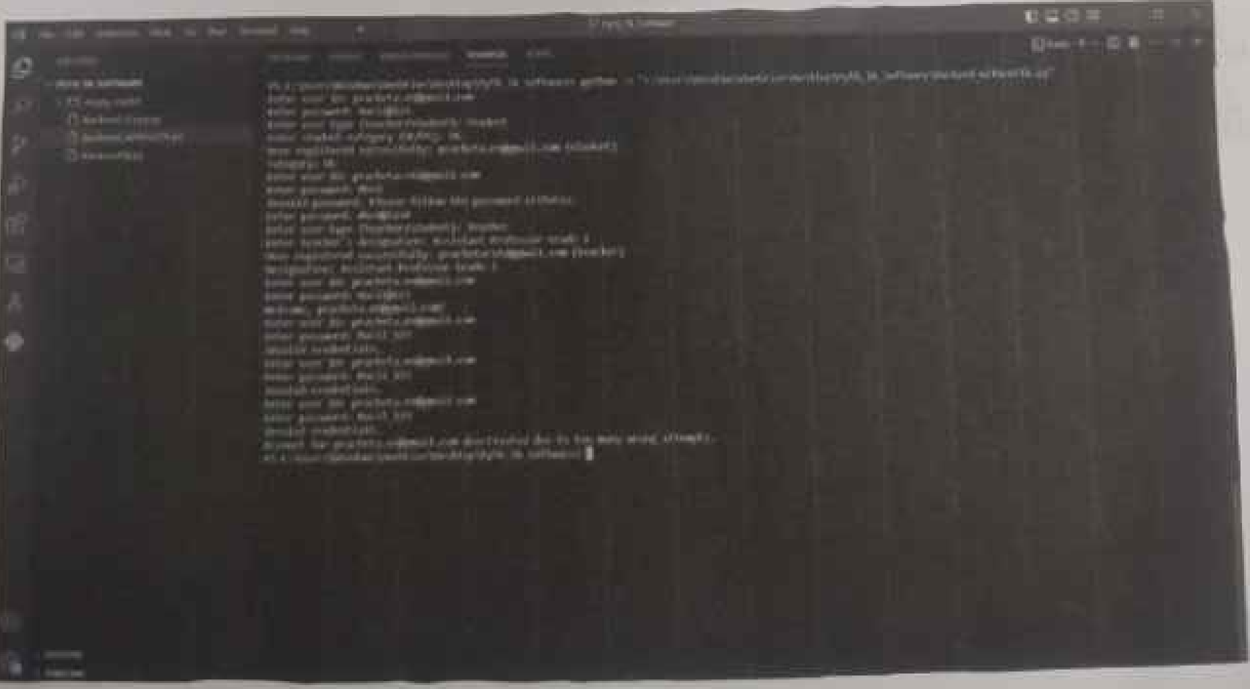
• Datetime module: The datetime module has been used to store the time of registering a user into the academic unit, which is stored into the reg_time field of the user.

• Pickle library: The pickle library has been imported to save and store the user data in "user_data.pkl" file so that it can be retrieved and used even after the program is closed

Teacher's Signature.......................



```
import tkinter as tk
from tkinter import messagebox
import re
from datetime import datetime

class Authenticator:
    class User:
        def __init__(self, user_id, password, [illegible]):
            self.user_id = user_id
            self.password = password
            self.attempts = 0
            self.active = True
            self.[illegible]

    class Teacher(User):
        def __init__(self, user_id, password, designation, [illegible]):
            super().__init__(user_id, password, [illegible])
            self.designation = designation

    class Student(User):
        def __init__(self, user_id, password, category, [illegible]):
            super().__init__(user_id, password, [illegible])
            self.category = category

    class [illegible](User):
        def __init__(self, user_id, password, name, [illegible], address, contact, [illegible]):
            super().__init__(user_id, password)
            self.name = [illegible]
            self.[illegible]
            self.address = address
            self.email = user_id
            self.contact = contact
            self.[illegible]

    def __init__(self):
        self.users = {}
```

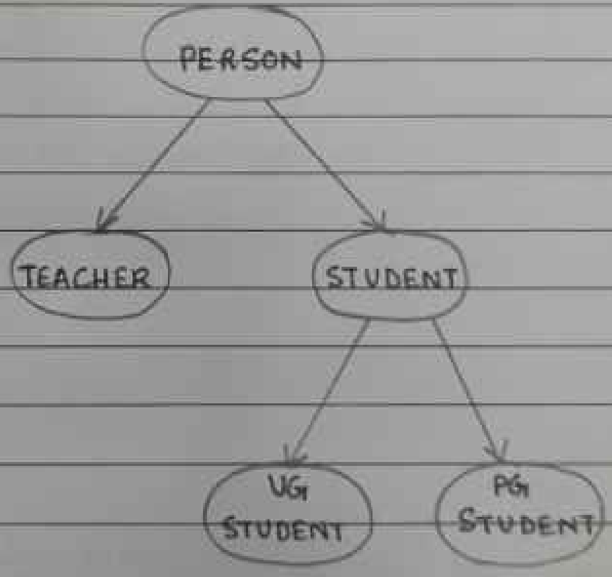


```
PS [illegible] python -u "[illegible].py"
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Enter user type (Teacher/Student): Student
Enter student category (UG/PG): UG
User registered successfully: prachetu.m@gmail.com (Student)
Category: UG
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Invalid password. Please follow the password criteria.
Enter password: [illegible]
Enter user type (Teacher/Student): Teacher
Enter teacher's designation: Assistant Professor Grade 1
User registered successfully: prachetu.m@gmail.com (Teacher)
Designation: Assistant Professor Grade 1
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Welcome, prachetu.m@gmail.com!
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Invalid credentials.
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Invalid credentials.
Enter user ID: prachetu.m@gmail.com
Enter password: [illegible]
Invalid credentials.
Account for prachetu.m@gmail.com deactivated due to too many wrong attempts.
PS [illegible]>
```

Expt. No. PAGE: DATE: / /

IMPLEMENTATION

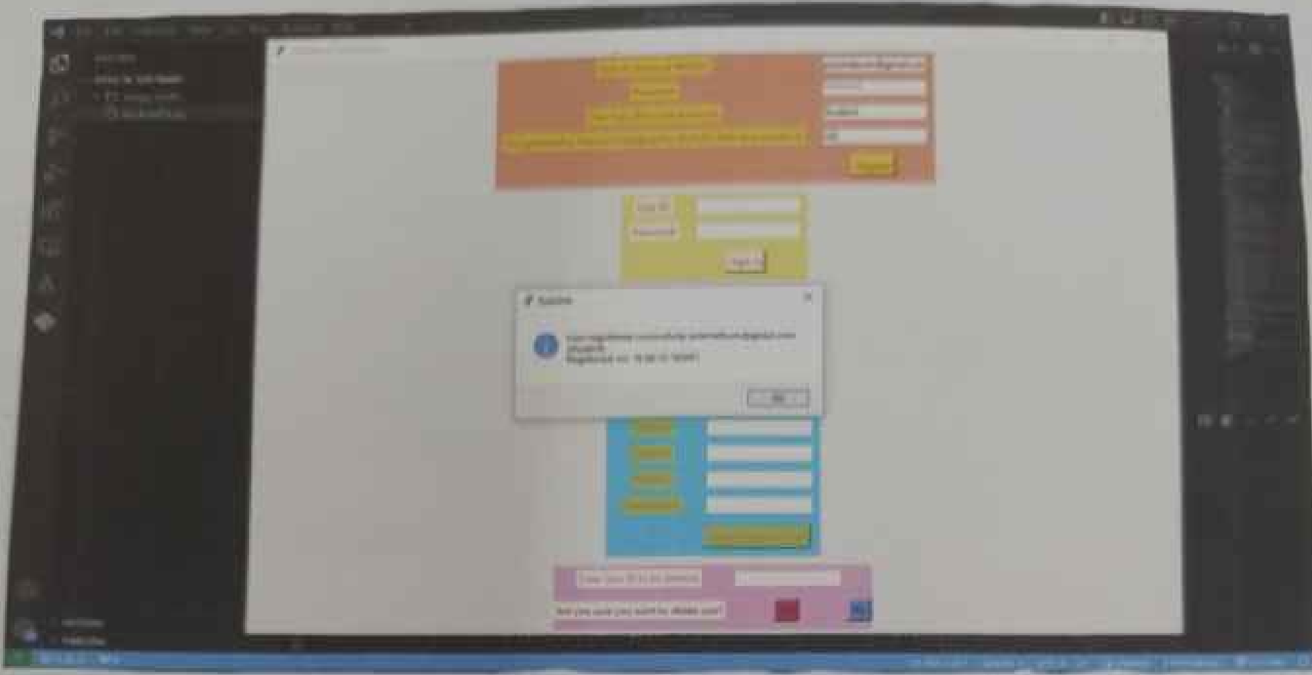
An object-oriented programming approach has been used to develop the application at the backend. We define a class AcademicUnit which contains as individual "sub" classes the basic units and details of the academic unit as its data members and we define member functions to implement the relevant functionality that describes the behavior of the unit. The hierarchial structure of the academic unit is:



To implement this hierarchy, we define the following child classes of class AcademicUnit:

• Person : It defines as a member method function __init__ which sets and initializes the essential attributes associated with each person in the unit: it sets the user id, password, time of registration (these three are passed to

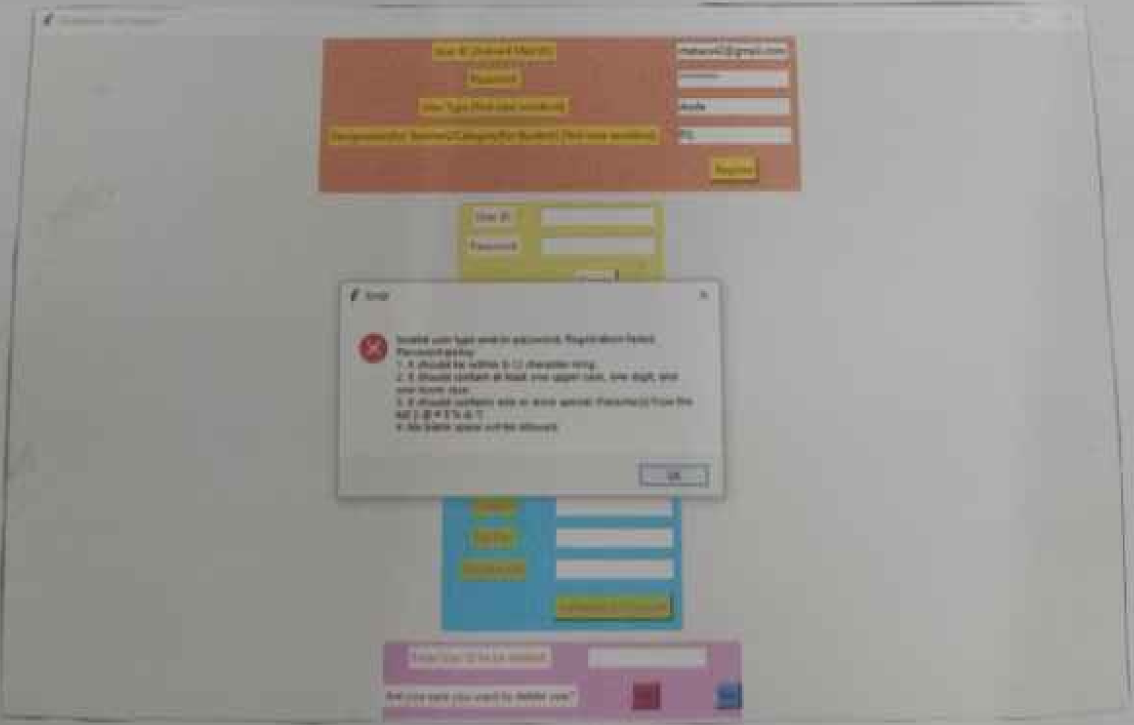
Teacher's Signature..



Success

User registered successfully: [illegible]@gmail.com
[illegible]
Registered on: [illegible]

OK



User ID (Student Mail ID) mahara42@gmail.com

Password

User Type (first one uppercase) Asofe

Designation(for Teacher)/Category(for Student) (first one uppercase) PG

Register

User ID

Password

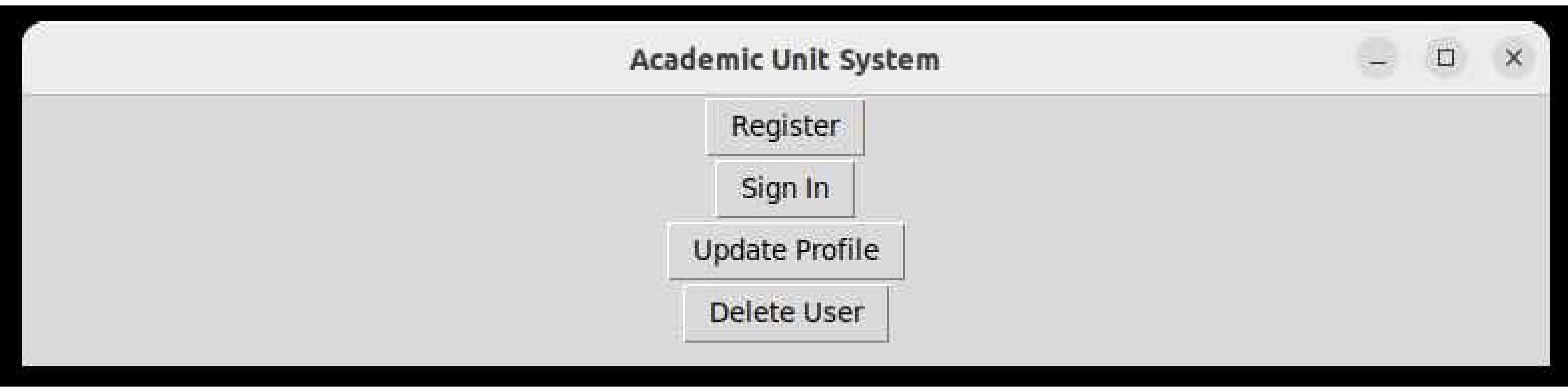
Error

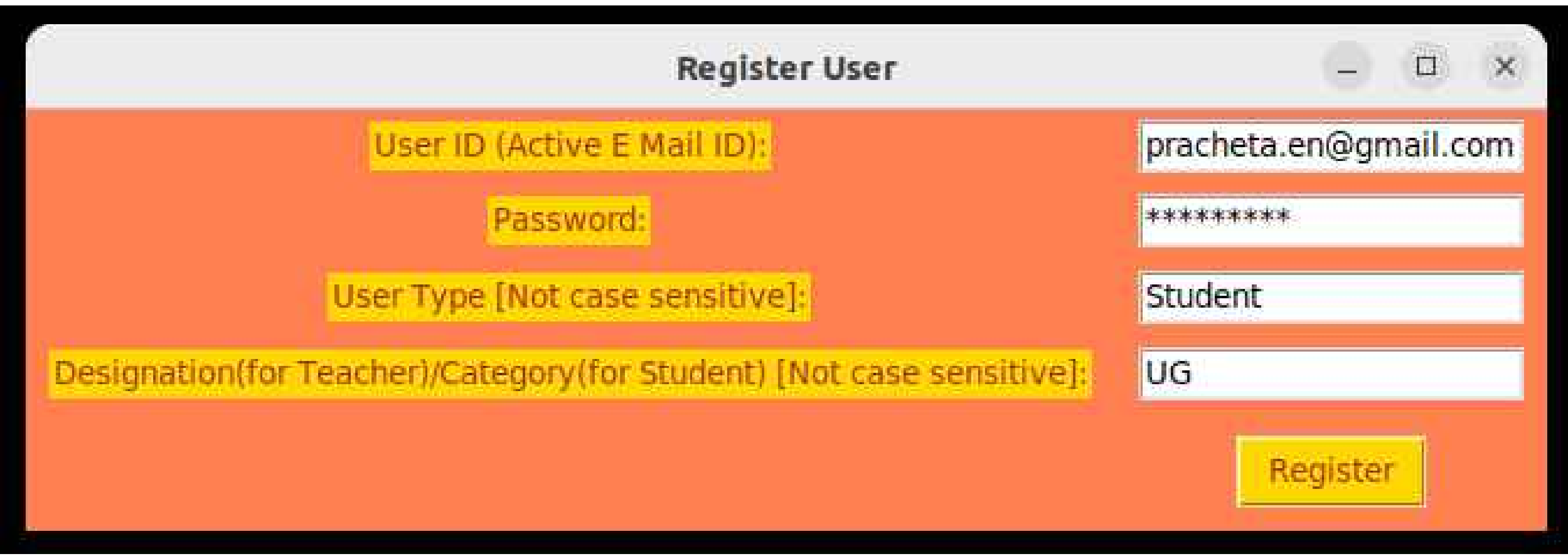
Invalid user type and/or password. Registration failed.
Password policy:
1. It should be within 6-12 character long.
2. It should contain at least one upper case, one digit, and one lower case.
3. It should contain one or more special character(s) from the list [@ # $ % & ?].
4. No blank space will be allowed.

OK

Enter your ID to be deleted

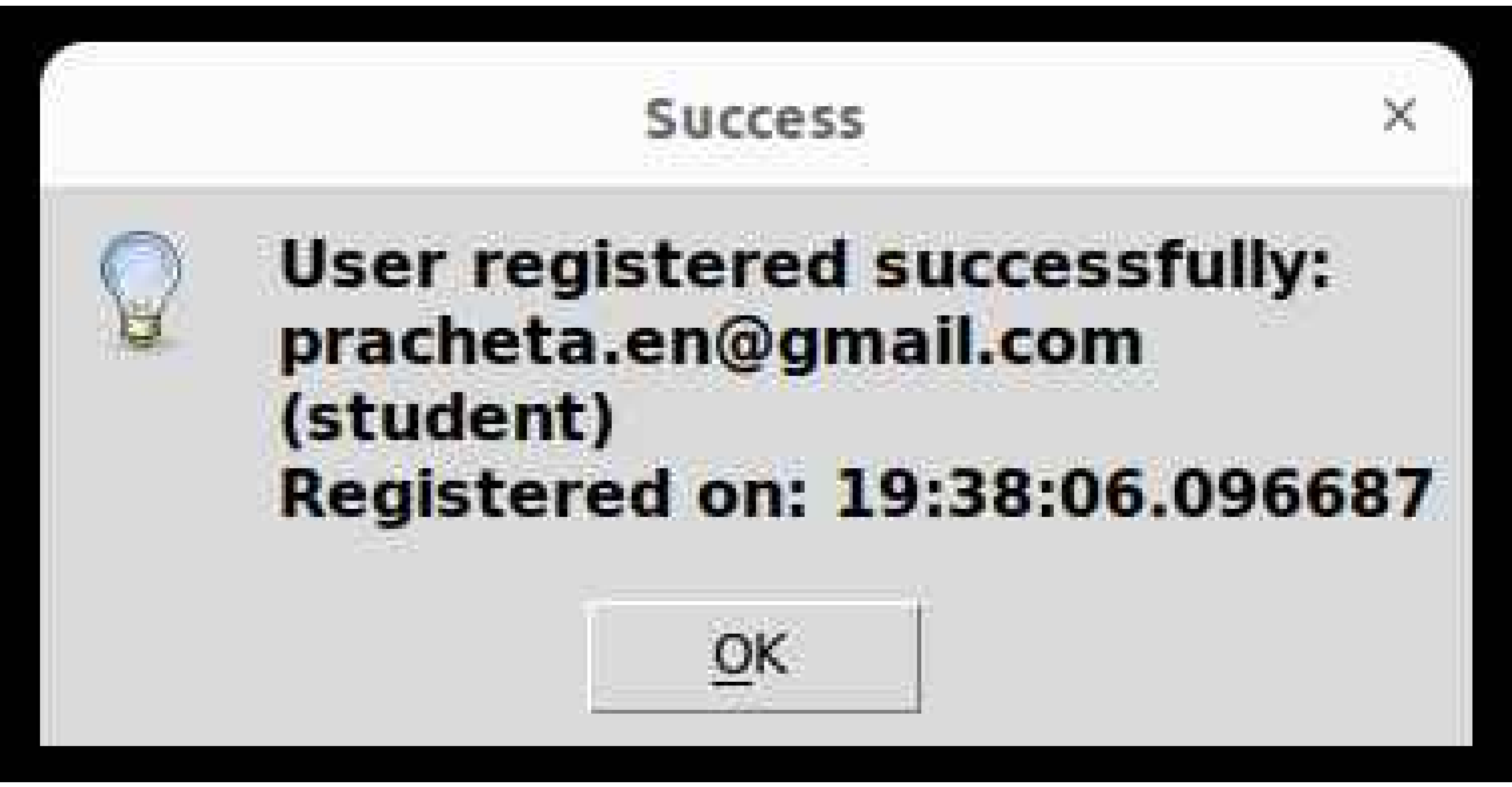
Are you sure you want to delete user?


Academic Unit System
Register
Sign In
Update Profile
Delete User



Register User

User ID (Active E Mail ID):	pracheta.en@gmail.com
Password:	*********
User Type [Not case sensitive]:	Student
Designation(for Teacher)/Category(for Student) [Not case sensitive]:	UG



User registered successfully:
pracheta.en@gmail.com
(student)
Registered on: 19:38:06.096687

Expt. No.

Orion
PAGE:
DATE: / /

the function) and initialize the number of attempts to sign in
to 0, and the user to be in an active state.

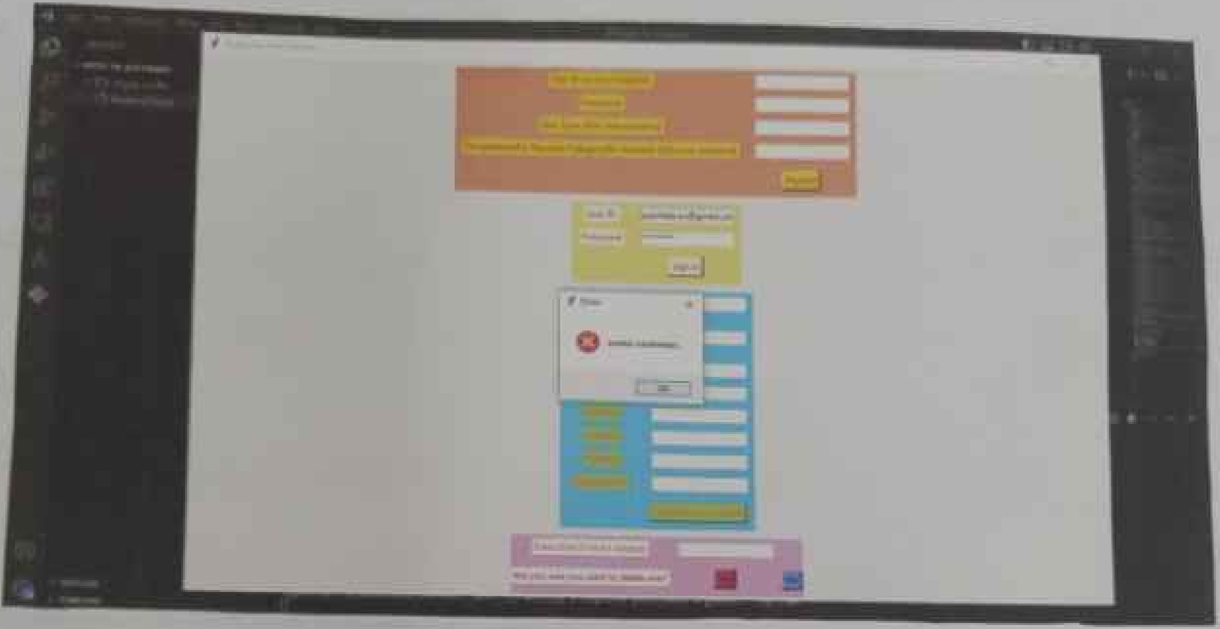
• Teacher : Makes use of Person as its sibling class to inherit
the function __init__ defined in class Person, which creates
and initializes all the basic attributes of the teacher and
also sets the designation of the user as entered by
the user.

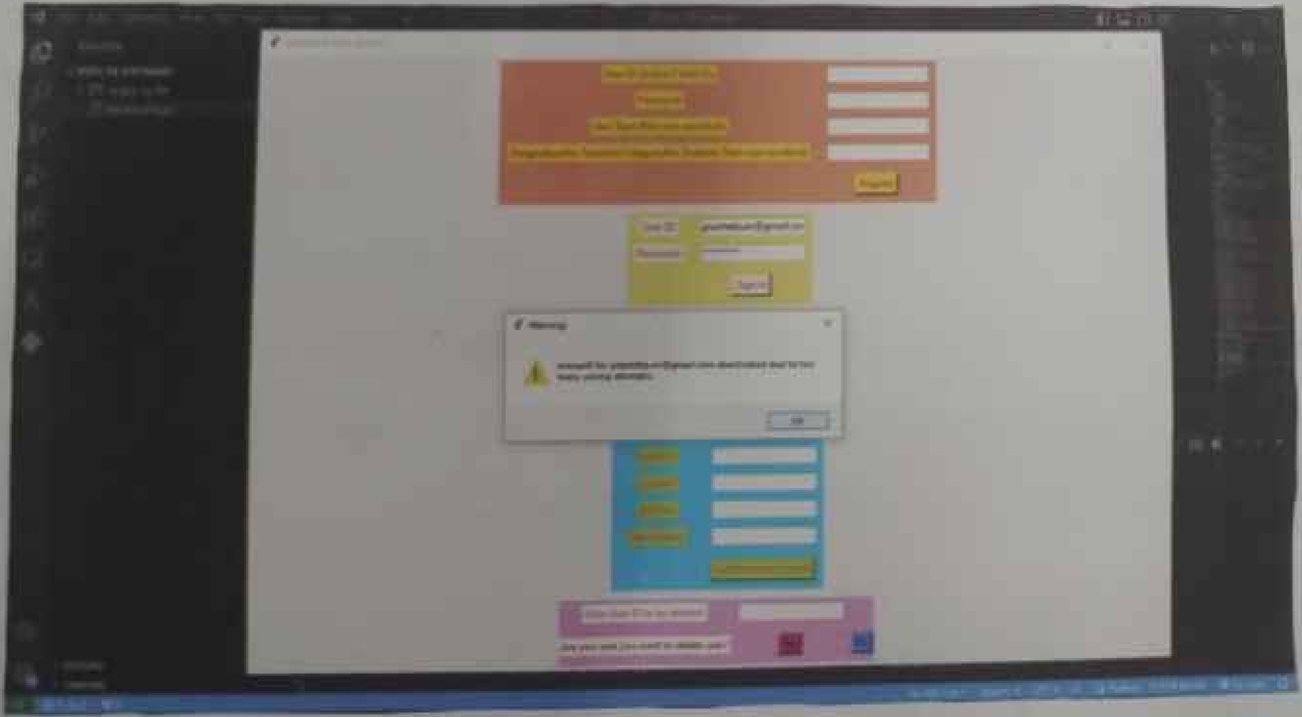
• Student : Makes use of Person as its sibling class to inherit
and invoke the function __init__. and initialises the basic
attributes of a student and sets the category of a student
as either a UG or a PG student.

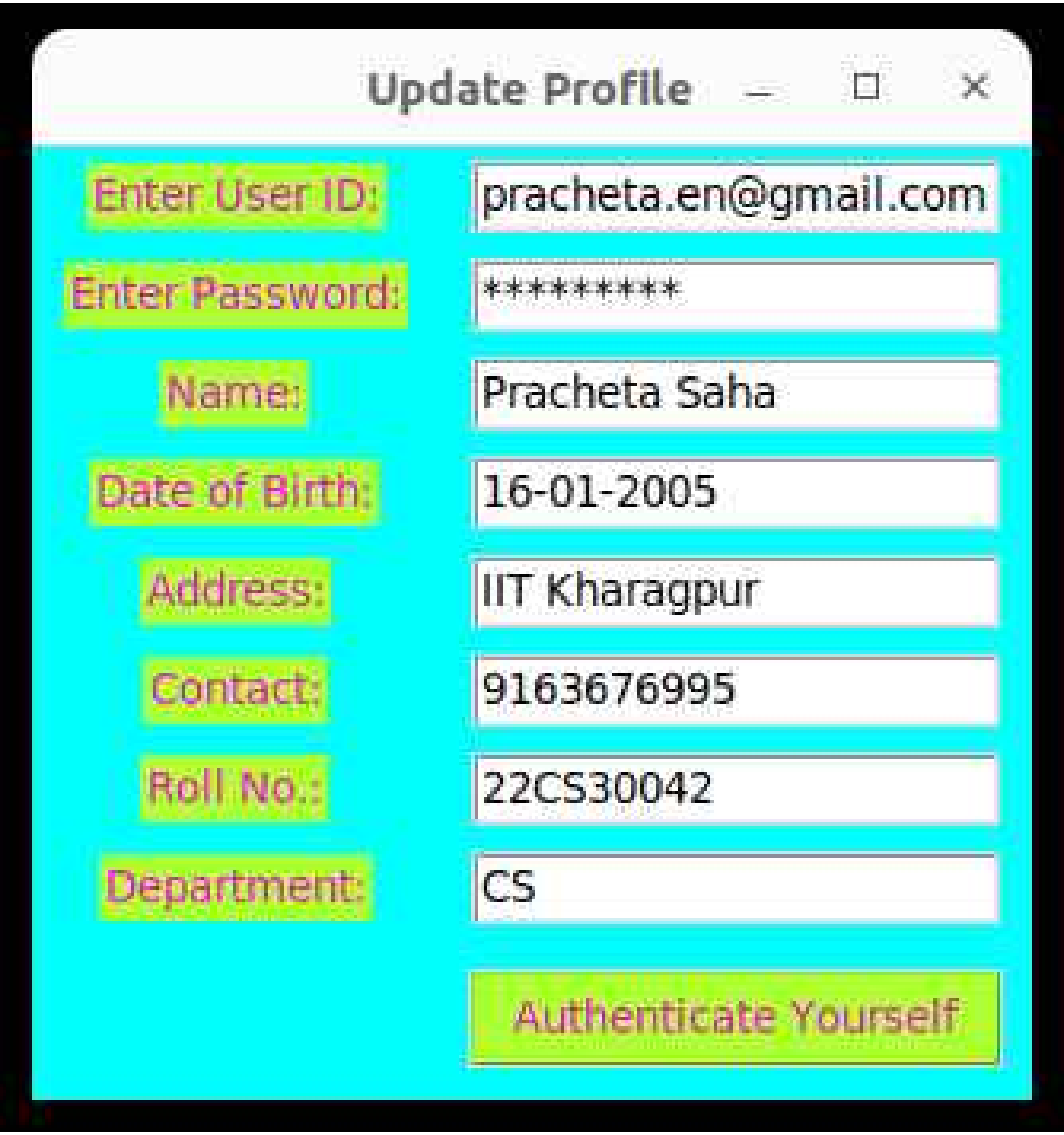
• Profile : It too uses super() to inherit features of __init__
function which is defined in sibling class Person. It is used
to store the profile fields of a person: the name, the
date of birth, address, email id, (which happens to be the
user id and therefore needs not be entered explicitly) contact
number, roll number and the department.

Within the class AcademicUnit, member functions have been
defined to implement the register, sign-in, update profile
and delete user features of the GUI.

Teacher's Signature..........................

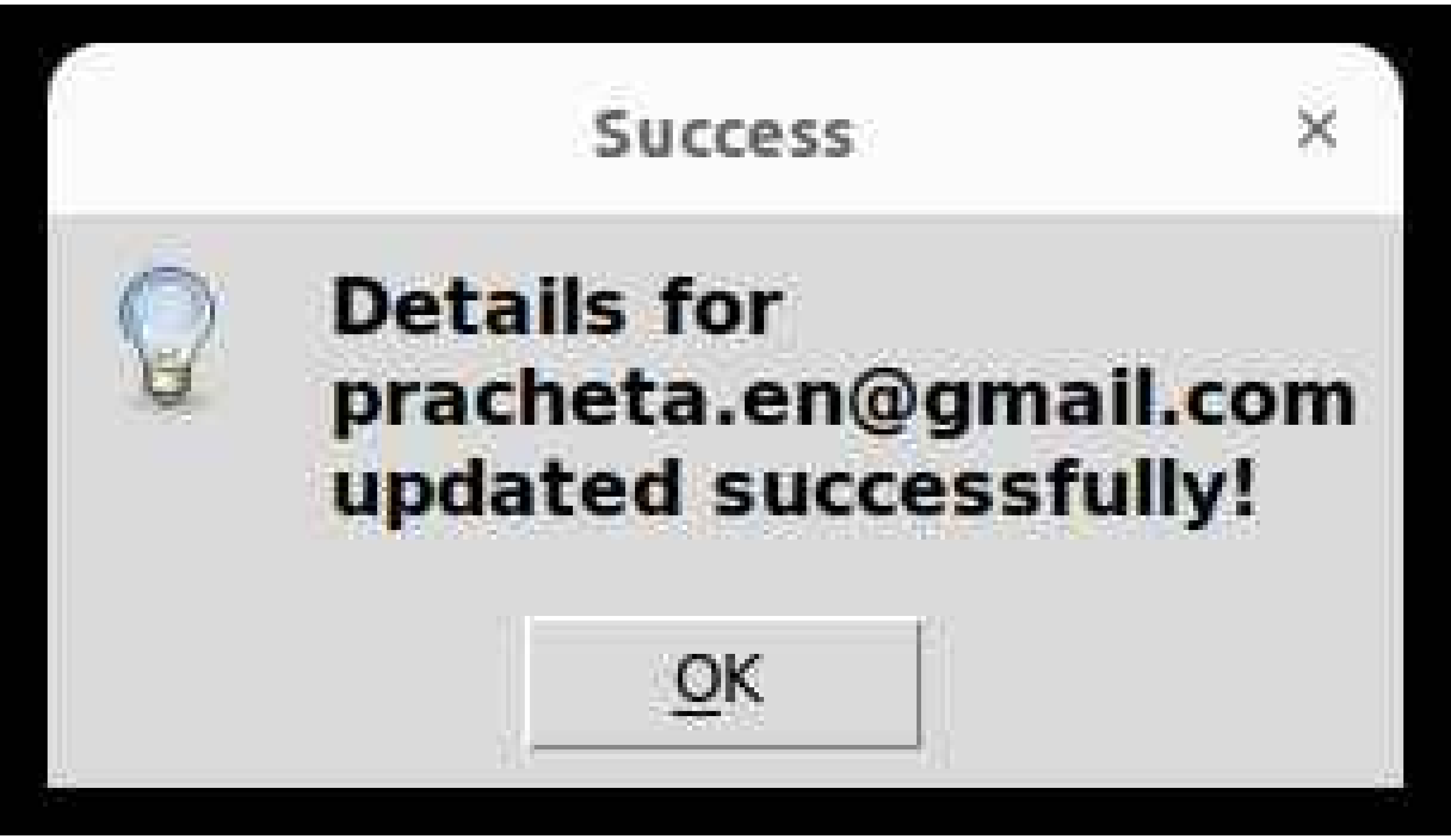






Update Profile

Enter User ID:	pracheta.en@gmail.com
Enter Password:	*********
Name:	Pracheta Saha
Date of Birth:	16-01-2005
Address:	IIT Kharagpur
Contact:	9163676995
Roll No.:	22CS30042
Department:	CS


Success

Details for
pracheta.en@gmail.com
updated successfully!

Expt. No.

Orion
PAGE:
DATE: / /

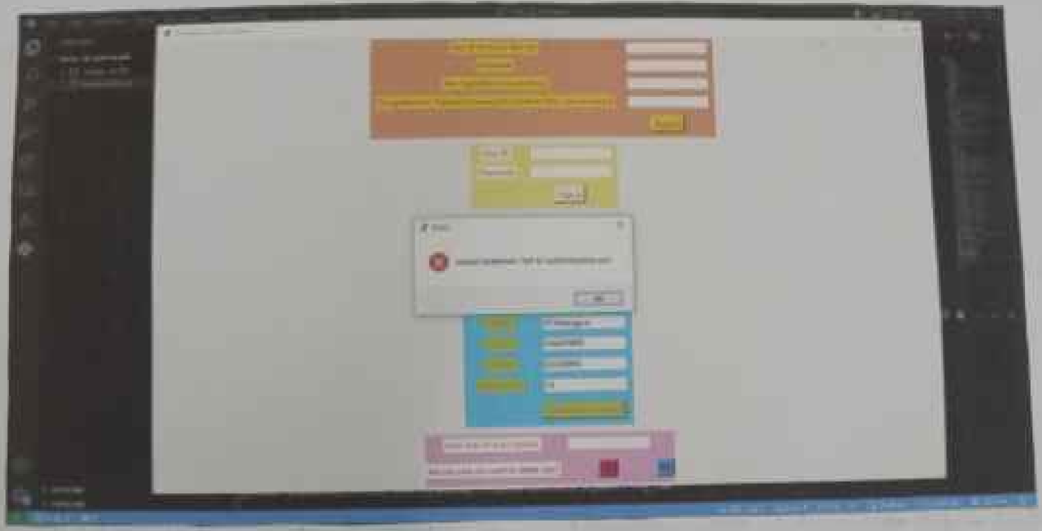
__init__: Member function that takes in object self and is an instance of the class and initialize, corresponding to self, two lists, users[] and profiles[].

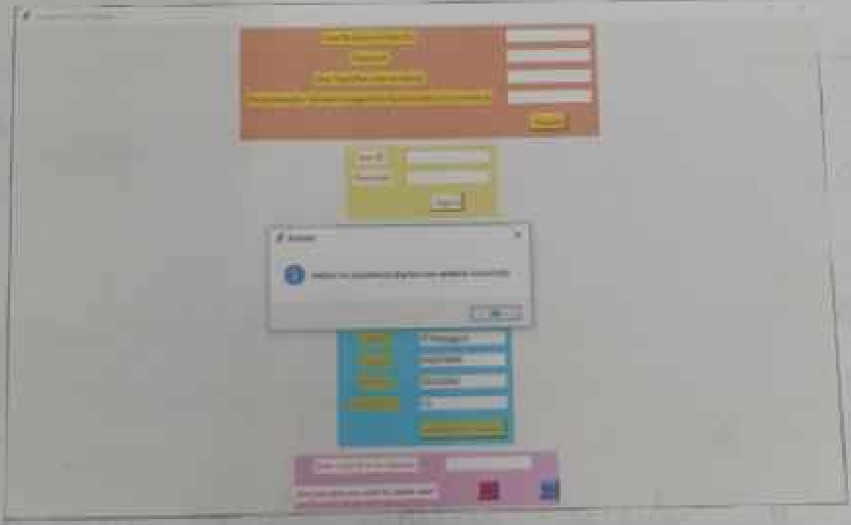
is_valid_password: It checks whether the password entered at the time of registration is valid or not.

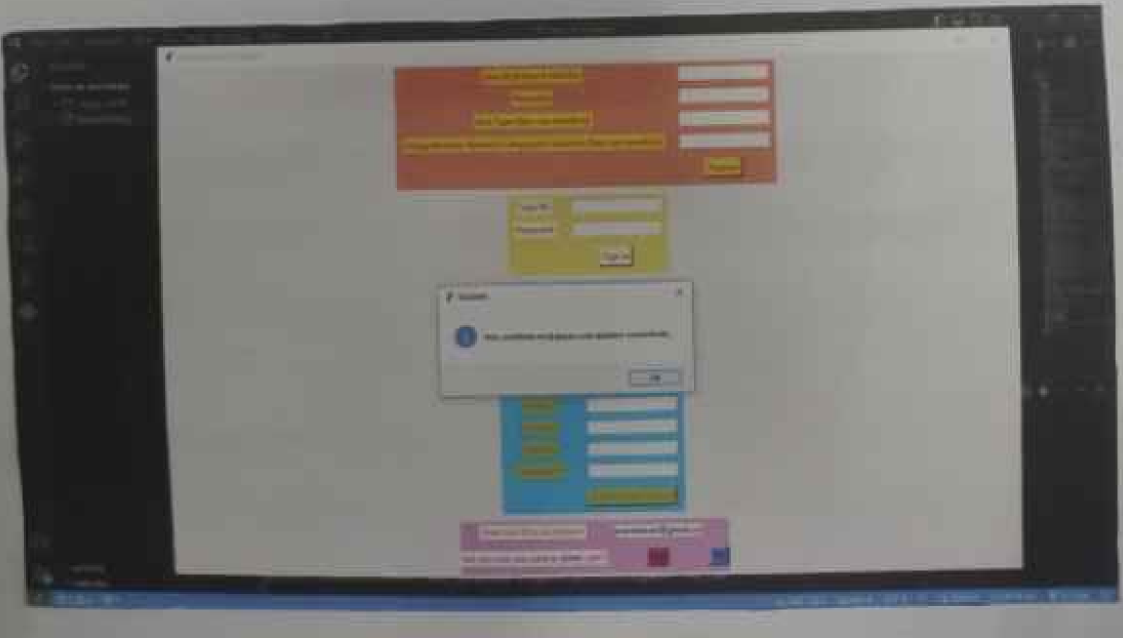
register_user: It registers a user after checking whether the entered details are valid. It checks whether the password is valid by invoking method is_valid_password and also verifies whether the user type entered is valid. If any of the conditions are not met, it flashes an error message mentioning the password policy and asking the user to register again. Else it successfully registers the user and displays a success message and appends to the list of registered users and profiles.

sign_in: It caters to a user's request of sign in after verifying entered credentials. If the user id or password is not valid, it displays an error message. If it has been found that a particular user id has 3 consecutive sign in attempts that have failed, the account is deactivated. Else, the user is signed in and the attempt counter is set to 0.

Teacher's Signature..........


User ID
Password
OK


User ID
Password
OK


User ID
Password
Sign in
OK

Expt. No.

Orion
PAGE:
DATE: / /

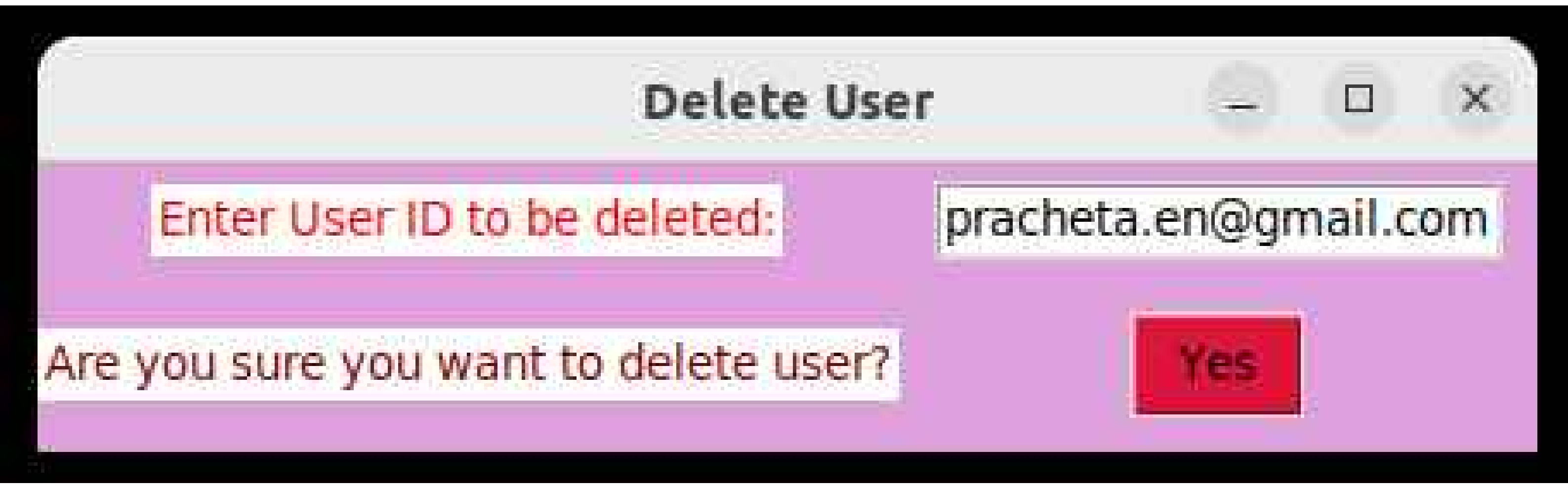
update_profile: It lets only an authenticated user update the profile for a particular user id. It first verifies the user id and password entered and if the credentials do not match, the user is denied permission to modify profile. else the profile details entered are updated into the list of profiles.

delete_user: It is used to remove a user from the list of users and profiles provided that the given user id exists.

nil: It is a function used to display a message box when a deletion action is requested for cancellation by a user.

Next, we define the class AcademicUnitGUI which defines the GUI functions for the application. The main window of the interface is titled "Academic Unit System" and object self calls the constructor AcademicUnit through self.academic_unit. Four different frames are defined for user registration, sign in, profile update and user delete. Each frame has a set of labels into which the user enters data and buttons which on clicking, triggers the functions specified under command. These functions in turn call the main function defined in the backend class AcademicUnit.

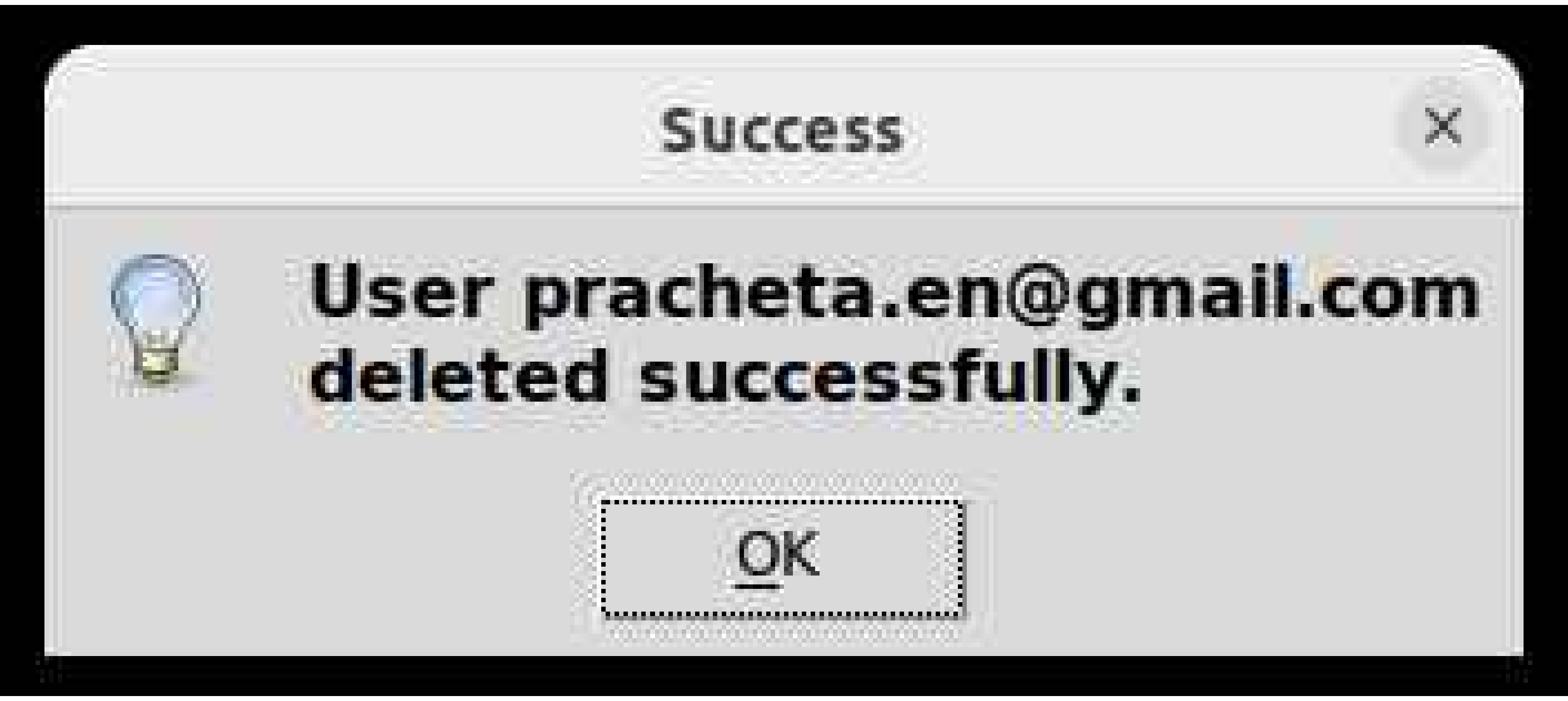
Teacher's Signature...



Delete User

Enter User ID to be deleted: pracheta.en@gmail.com

Are you sure you want to delete user?



User pracheta.en@gmail.com deleted successfully.