Assignment - 2
Software Application Development Using Python
Report : Academic Unit Registration System

Name : Tuhin Mondal   **Roll Number : 22CS10087**

The project aims to design and implement an Academic Registration System tailored for small academic unit. This system provides a comprehensive solution for managing the academic records and user information within the unit. Key Features of this project are:

User Authentication: The system ensures secure user authentication, managing login credentials and login attempts for individuals within the academic unit.

Teacher and Student Profiles: Teachers and students are categorized and provided with dedicated profiles, capturing essential information such as names, departments, dates of birth, and more.

Research and Specialization: For teachers, the system accommodates research areas, acknowledging the diverse academic interests within the unit. Postgraduate students have the additional attribute of specialization.

Academic Progress Tracking: Undergraduate students have features for tracking academic progress, with Cumulative Grade Point Averages (CGPA) recorded for each student.

Objects Defined :

1) **Person:**
   ID: Unique identifier for the person.
   Password: Password for authentication.
   Attempts: Number of login attempts left (default set to 3).

2) **Teacher (inherits from Person):**
   Name: Name of the teacher.
   Department: Department to which the teacher belongs.
   Date of Birth: Date of birth of the teacher.
   Research areas: Research area of the teacher.

3) **Student (inherits from Person):**
   Name: Name of the student.
   Department: Department to which the student belongs.
   Date of Birth: Date of birth of the student.
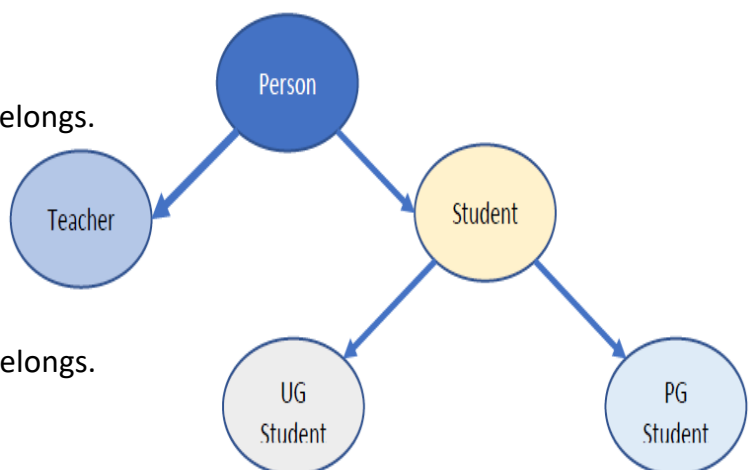   Roll Number: Roll number of the student.
   Year Enrolled: Academic year of the student.

4) **UG Student (inherits from Student):**
   CGPA: Cumulative Grade Point Average of the undergraduate student.

5) **PG Student (inherits from Student):**
   Specialisation: Specialization area of the post graduate student.

GUI Frames Used:

1) **Home Page:** It serves as the graphical entry point to an academic unit application developed using the Tkinter library in Python. When invoked, this function clears any existing frames, creating a new frame styled with a specific background colour. The frame comprises a welcome message and a title. It prompts new users to sign up provides a corresponding "Sign Up" button, which, when clicked, opens the registration page. Additionally, the home page acknowledges existing users with a separate label and offers a "Sign In" button, activating the Sign In page.
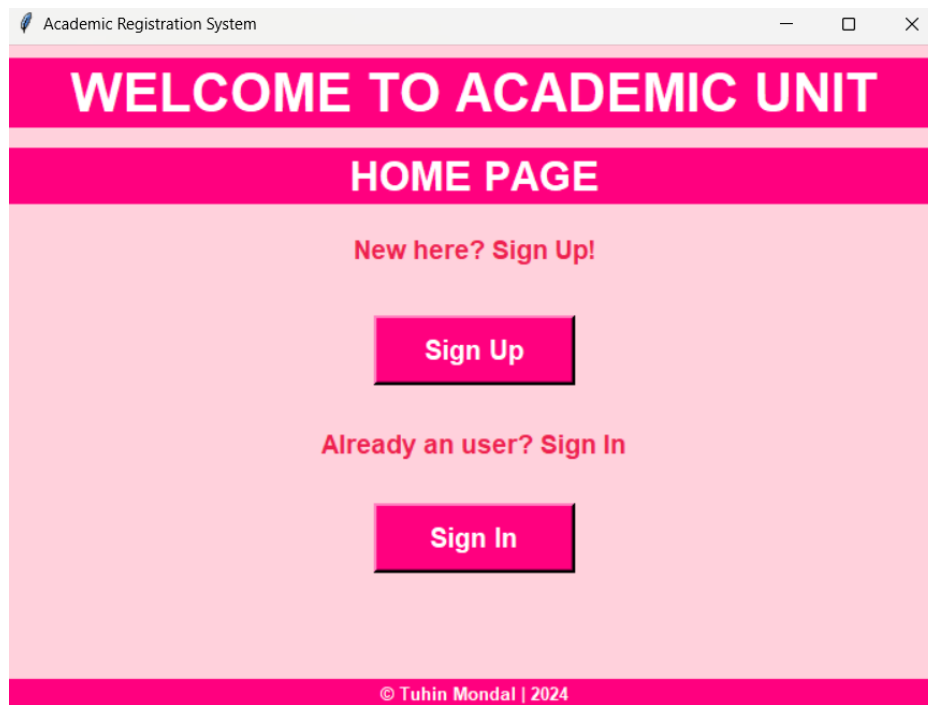
2) **Sign Up Page:** The registration page is structured with a prominent title, "REGISTRATION PAGE," and prompts users to select their registration type. Options are provided through distinct buttons for registering as an Undergraduate (UG) Student, a Postgraduate (PG) Student, or a Teacher. These buttons are linked to the Student + Teacher Register page, which handles the specific registration process for each user type. Additionally, a "Back to Home Page" button allows users to easily navigate back to the home page.
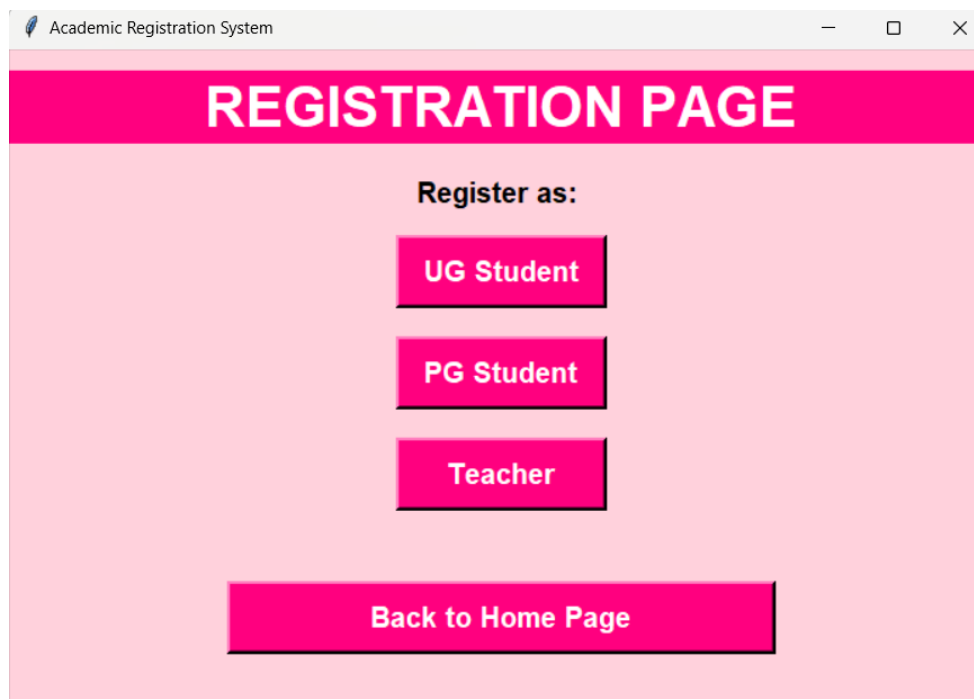
3) Student + Teacher Register page: The title dynamically adjusts based on the user type specified as a parameter (UG Student, PG Student, or Teacher). The page includes input fields for User ID and Password, along with corresponding entry widgets. There are two hidden labels which shows up if the username is invalid or already exists, and the second one checks if the password meet all the criteria. A registration button is provided to initiate the registration process, and a label is allocated to display validation messages. If username and password are valid it redirects to the Update Info page. Notably, a "Show Password" button allows users to toggle password visibility for enhanced user experience. Additionally, buttons for redirection to the home page and toggling password visibility are included.

4) Edit Pages: These GUIs are used for creating graphical user interface pages to edit and update user information for Teacher, Undergraduate Student (UG Student), and Postgraduate Student (PG Student) profiles, respectively. Each edit page is structured with a consistent layout, featuring labels and entry fields for various attributes such as name, department, date of birth, and more, depending on the user type. Entries already have the pre-defined content to edit. Users can input their updated information, and a "Save Changes" button triggers the corresponding submission. The pages have labels indicating the user ID, profile type, and specific fields for user details. These edit pages contribute to a user-friendly interface for updating and managing user profiles within the academic unit system.

UPDATE YOUR INFO

User ID:	person1@gmail.com
Profile Type:	UG Student
Name:	
Department:	
Date of Birth:	
Roll Number:	
CGPA:	
Year of Graduation:	

Save Changes

UPDATE YOUR INFO

User ID:	person1@gmail.com
Profile Type:	UG Student
Name:	Tuhin Mondal
Department:	CSE
Date of Birth:	22-11-2004
Roll Number:	22CS10087
CGPA:	9.70
Year of Graduation:	2026

Save Changes

UPDATE YOUR INFO

User ID:	person2@gmail.com
Profile Type:	PG Student
Name:	Alice
Department:	ECE
Date of Birth:	19-02-1998
Roll Number:	22EC3F033
Specialisation:	VLSI
Year of Graduation:	2029

Save Changes

UPDATE YOUR INFO

User ID:	person3@gmail.com
Profile Type:	Teacher
Name:	XYZ
Department:	CSE
Date of Birth:	17-11-1982
Reasearch Areas:	Algorithms, Cryptography

Save Changes

5) **Sign In Page:** The page includes entry fields for User ID and Password, with a "Sign In" button triggering the Check function There's also an option to show/hide the password and a button to navigate back to the home page. It provides visual feedback with a label to display error messages or information. The check function is responsible for validating the user credentials during the sign-in process. It checks if the entered User ID matches any existing user. If a match is found, it verifies the password and redirects the user to the appropriate view (Teacher, UG Student, PG Student). It also handles scenarios such as incorrect passwords and limited login attempts. Error messages are displayed on a label for user feedback. If no matching user is found, it prompts the user to enter a valid username.

SIGN-IN PAGE

User ID

person@gmail.com

Password

••••

Sign In

Show Password

Back to Home Page

SIGN-IN PAGE

User ID

Password

Sign In

Enter Valid Username!

Show Password

Back to Home Page

SIGN-IN PAGE (left)

User ID
person3@gmail.com

Password

Sign In

Wrong Password! You have 2 attempts left!

Show Password

Back to Home Page

SIGN-IN PAGE (right)

User ID
person3@gmail.com

Password

Sign In

Your account is deactivated!

Show Password

Back to Home Page

6) **View Page:** The view pages display profile information for Teachers, Undergraduate (UG) and Postgraduate (PG) students, respectively, within a Tkinter-based academic unit application. These pages showcase details such as User ID, Profile Type, Name, Department, Date of Birth, Roll Number, CGPA (for UG students), Specialization (for PG students), and Year of Graduation. The information is presented using labels and entries, with a provision for users to initiate the editing of their details or perform actions like signing out or submitting a deregistration request.

PROFILE INFO (left)

User ID:	person1@gmail.com
Profile Type:	UG Student
Name:	Tuhin Mondal
Department:	CSE
Date of Birth:	22-11-2004
Roll Number:	22CS10087
CGPA:	9.70
Year of Graduation:	2026

Edit   Sign Out

Deregistration request

PROFILE INFO (right)

User ID:	person2@gmail.com
Profile Type:	PG Student
Name:	Alice
Department:	ECE
Date of Birth:	19-02-1998
Roll Number:	22EC3F033
Specalisation:	VLSI
Year of Graduation:	2029

Edit   Sign Out

Deregistration request

Buttons and their Roles:

Sign Up Button (Sign Up): Takes the user to the registration page where they can choose to register as an Undergraduate (UG) student, Postgraduate (PG) student, or a Teacher.

Sign In Button (Sign In): Takes the user to the sign-in page where they can enter their credentials to sign in to their respective profiles.

Register Type Buttons:
UG Student Button: Takes the user to the registration page for Undergraduate (UG) students.
PG Student Button: Takes the user to the registration page for Postgraduate (PG) students.
Teacher Button: Takes the user to the registration page for Teachers.

Back to Home Page Button (on various pages): Takes the user back to the home page.

Sign In Check Button (on sign-in page):
Checks the entered user ID and password for sign-in validation.

Show Password Button (on sign-in and registration pages):
Toggles the visibility of the password, either showing or hiding it.

Edit Button (on profile view page):
Allows the Teacher to edit their profile information.

Sign Out Button (on profile view page):
Logs the profile out and returns to the home page.

Deregistration request Button (on profile view page)
 Initiates a deregistration request for the profile.

Save Changes Button (on edit pages):
Saves the changes made to the profile.

Conclusion:

In summary, the project employs Tkinter in Python to create a user-friendly graphical interface for user registration, sign-in, and profile management. Users, categorized as Undergraduate (UG) students, Postgraduate (PG) students, or Teachers, can register, sign in, and edit their profiles. The application features robust password validation, limited sign-in attempts, and a deregistration request option for Teachers. The Tkinter GUI design incorporates various elements such as buttons, labels, and entry fields, providing an intuitive navigation experience. We use Python's pickle module to serialize and deserialize the LIST object, storing and retrieving it from a binary file named "file.pkl." This enables persistent data storage, maintaining user information between different runs of the application.