

Software Engineering Laboratory | Lab Spring 2024

Lab-4:Software Application Development Using Python

Name-Sumit Kumar

Roll No-22CS30056

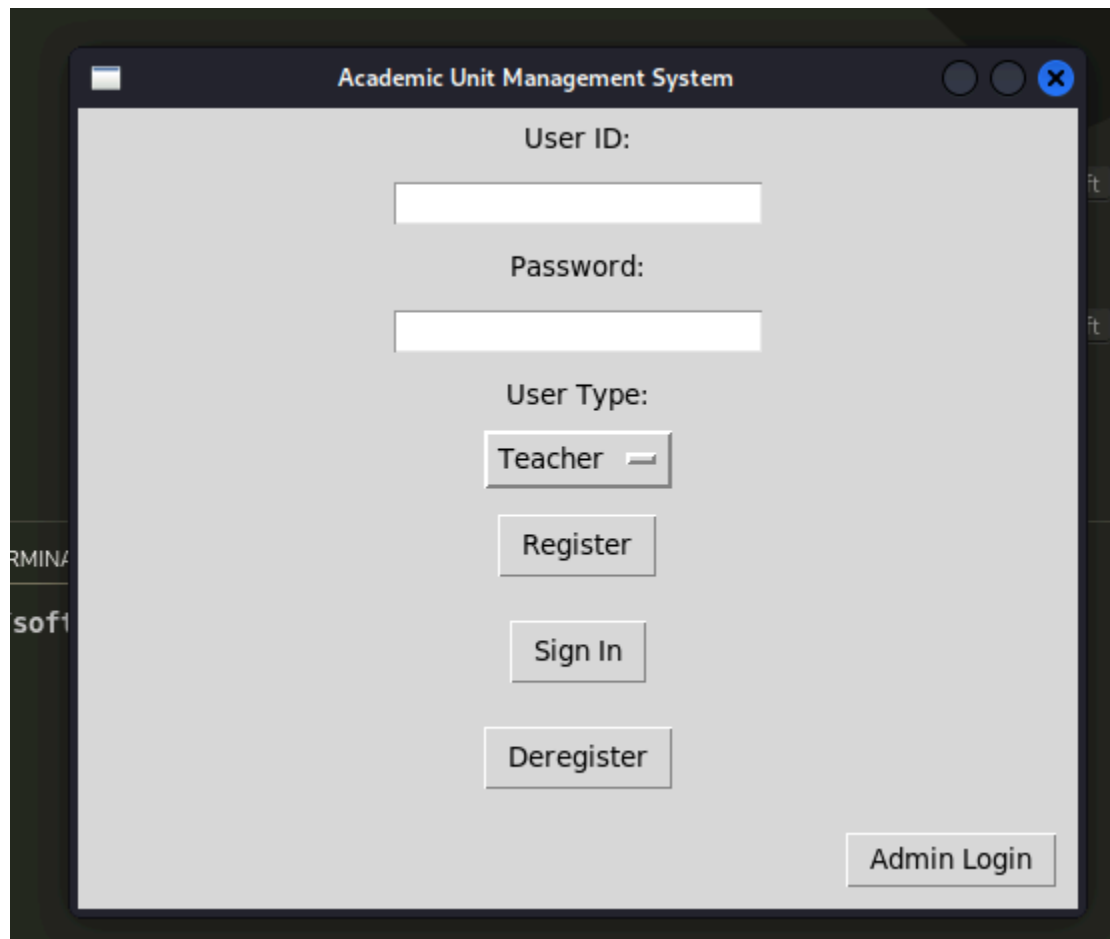
The application consists of the following codes:

- documentation.py- This file contains the application documentation.
- classes.py- This file contains the code for the classes of the application. It contains the code for the classes which are used in the application.
- main.py- This is the entry point of the application. It contains the main function, which is responsible for calling the other functions.
- gui.py- This file contains the code for the application's GUI. It has the code for the application's GUI. It is written in Tkinter.
- register_user.py- This file contains the code for the user's registration. It is responsible for taking the details of the user and storing it in the database.
- sign_in_user.py- This file contains the code for the sign-in of the user. It is responsible for taking the details of the user and checking it with the database.
- util_functions.py- This file contains the code for the utility functions of the application. It contains the code for the functions which are used in the application.
- password_check.py- This file contains the code for the user's password check. It is responsible for checking the user's password with the database. It also displays the password type if the correct matching of the password is not made.
- edit_user_data.py- This file contains the code for the editing of the user data. It is responsible for taking the details of the user and updating it in the database.
- edit_gui_functionality.py- This file contains the code for the GUI of the edit user data. It is responsible for taking the details of the user and updating it in the database. It mainly contains the code of the window, which is displayed when the user wants to edit his/her details.
- deregister_user.py- This file contains the code for the deregistration of the user. It is responsible for taking the details of the user and deleting them from the database.
- deregister_user_gui.py- This file contains the code for the GUI of the deregistration of the user. It is responsible for taking the details of the user and deleting it from the database. It mainly contains the code of the window, which is displayed when the user wants to deregister himself/herself.
- deactivate_account.py- This file contains the code for the deactivation of the user account. It is responsible for taking the details of the user and deactivating it from the database. The deactivation happens if the wrongly tries to enter someone's account after

- backend-py- This file contains the code for the backend of the application. It contains the code for the database of the application. It is responsible for loading and saving the user Records in the database.
- admin_gui_login.py- This file contains the code for the GUI of the admin login. It is responsible for taking the admin details and checking them with the database. The admin is a master person who is able to check the details of each and every user.
- admin_functions.py- This file contains the code for the admin functions. It is responsible for taking the admin details and checking them with the database. The admin is a master person who is able to check the details of each and every user. This file contains the implementations of all the functions which are used in admin_gui login.py
- admin_account_create.py- This file contains the code for the creation of the admin account. It is responsible for taking the details of the admin and storing it in the database.
- user_data.json- This file is the database of the application. It contains the details of the user in the form of JSON format using Python lists and dictionaries.

Application in use

To start the program file, we need to run the main.py file using the command `python3 main.py`



This is the start of the application.
On starting the code.

Let us register a user into the database

Academic Unit Management System

User ID:

user1@gmail.com

Password:

User Type:

Teacher

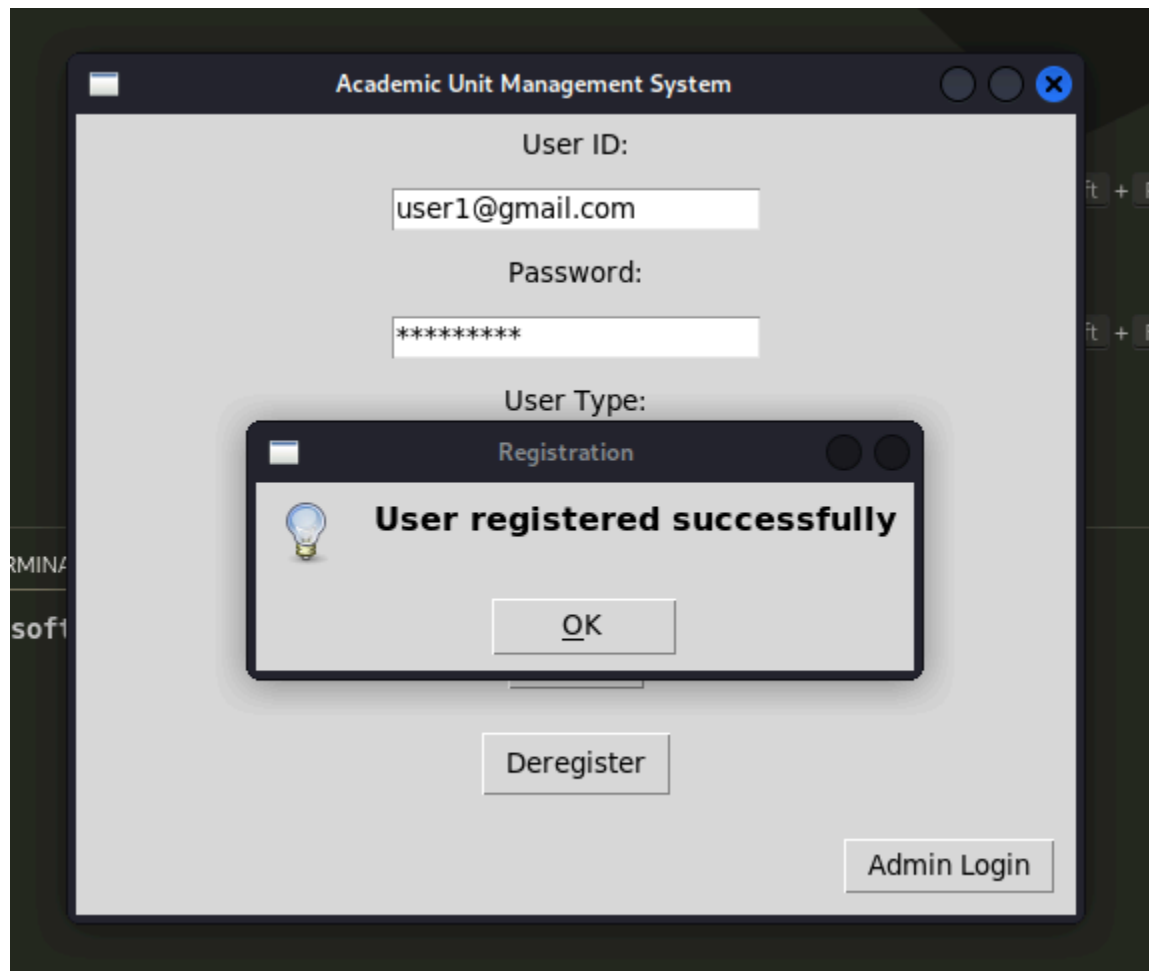
Register

Sign In

Deregister

Admin Login

Once we type the user ID and password and select the user Type, we create an account in the database. After successfully making an account, we receive a pop-up notification that we have registered successfully. Now, we can use these user credentials to log into the system.



This system's database follows a strict convention for the password of the user.

The password has the following criteria :

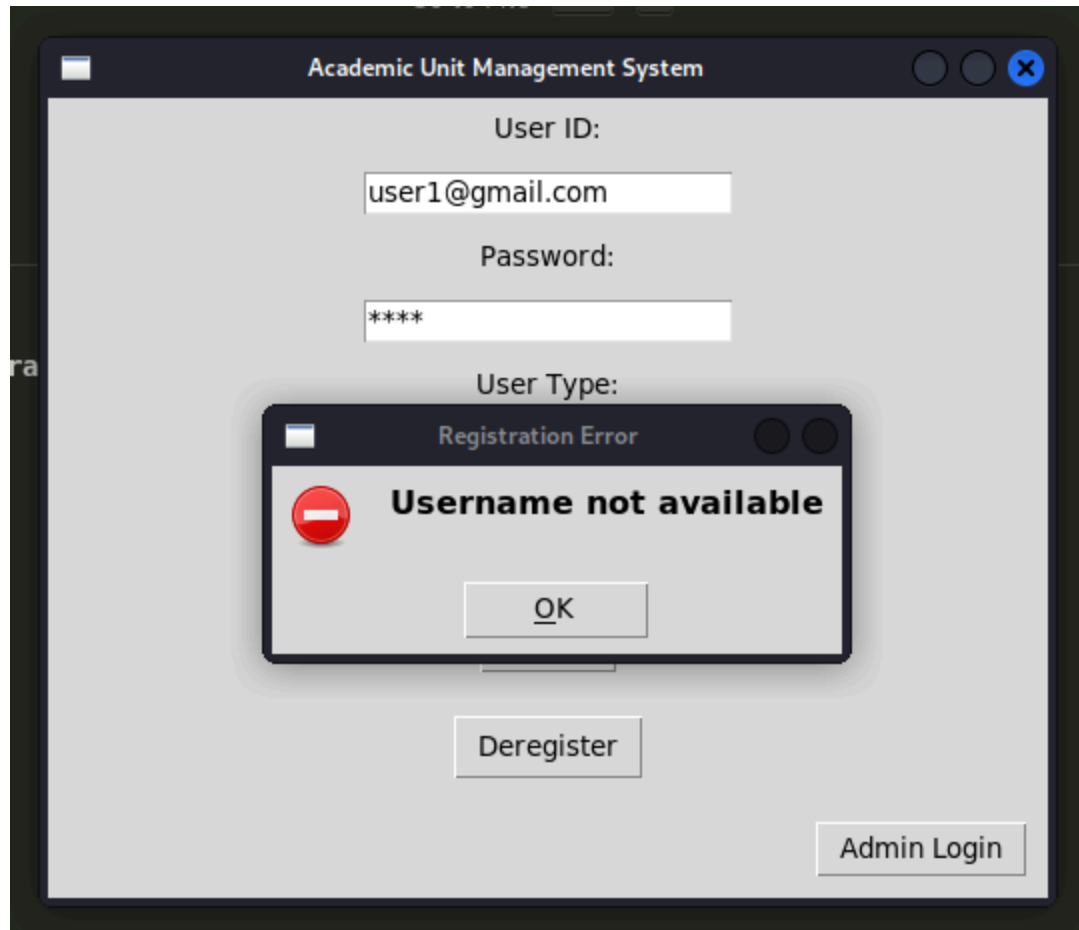
- a) It should be within 8-12 characters long.
- b) It should contain at least one upper case, one digit, and one lower case.
- c) It should contains one or more special character(s) from the list [! @ # \$ % & *]
- d) No blank space will be allowed.

If the correct password criteria are not used, then it will not register the account into the system, and the user will receive an error message about the Valid Password criteria.

Thus, only valid password criteria should be followed for this system.

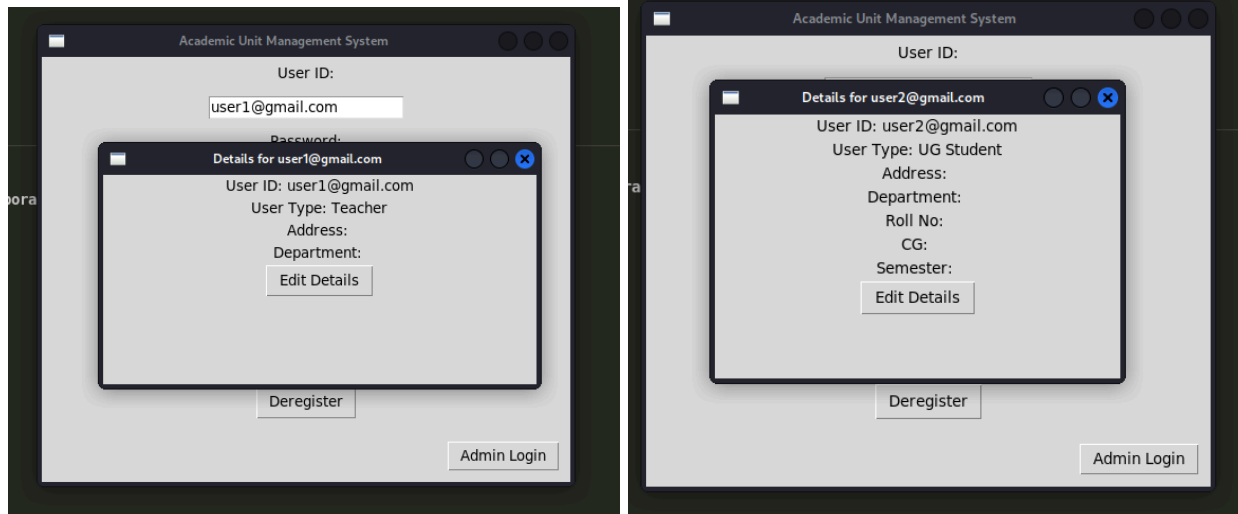


Also, a user can't sign with a user ID that already exists in the system. Like if I try to register with that email id, I will receive an error message like this username not available.

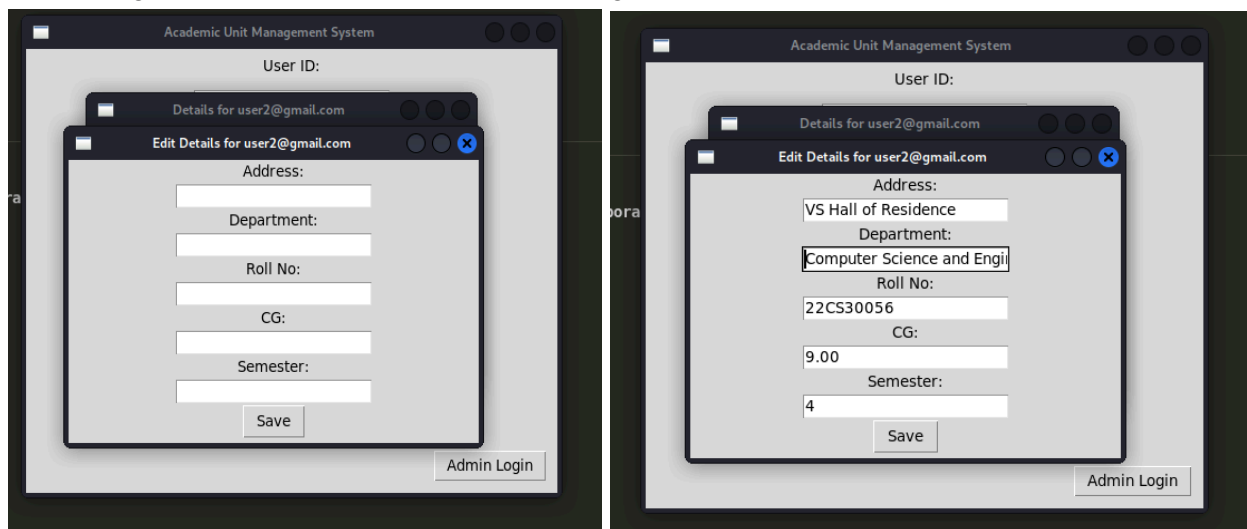


On successfully signing in To the system with the correct user ID and password We receive a message of signing in successfully and a window appears that shows all the user details like this. If the account is Initially signed in then the account details are empty and an Edit Details

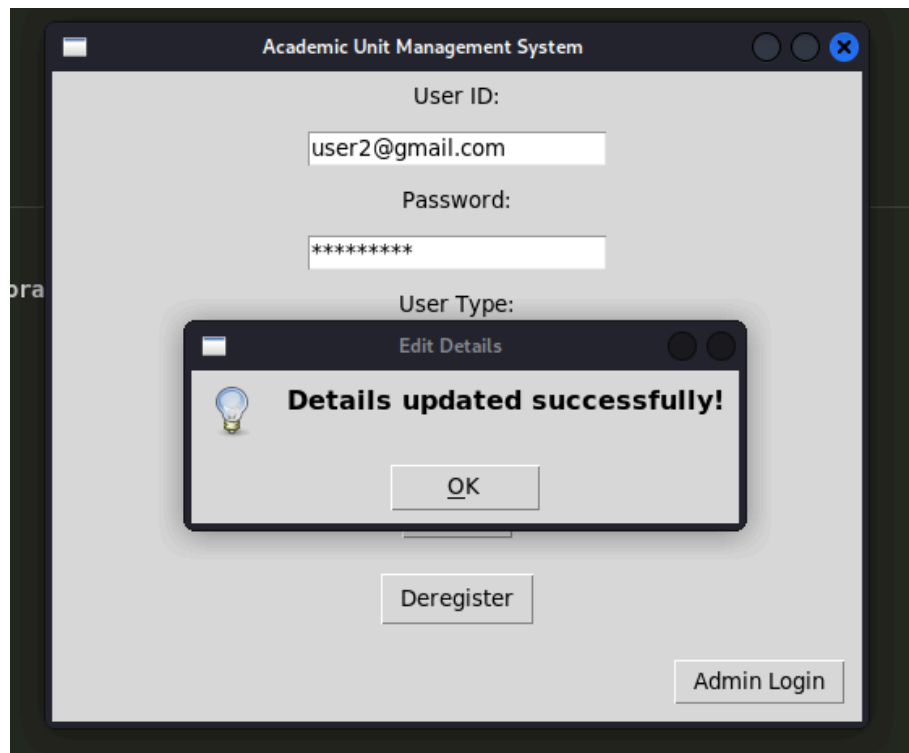
button is present which is there for the user to edit his /her details. The user can Edit his/her details by clicking on this button.



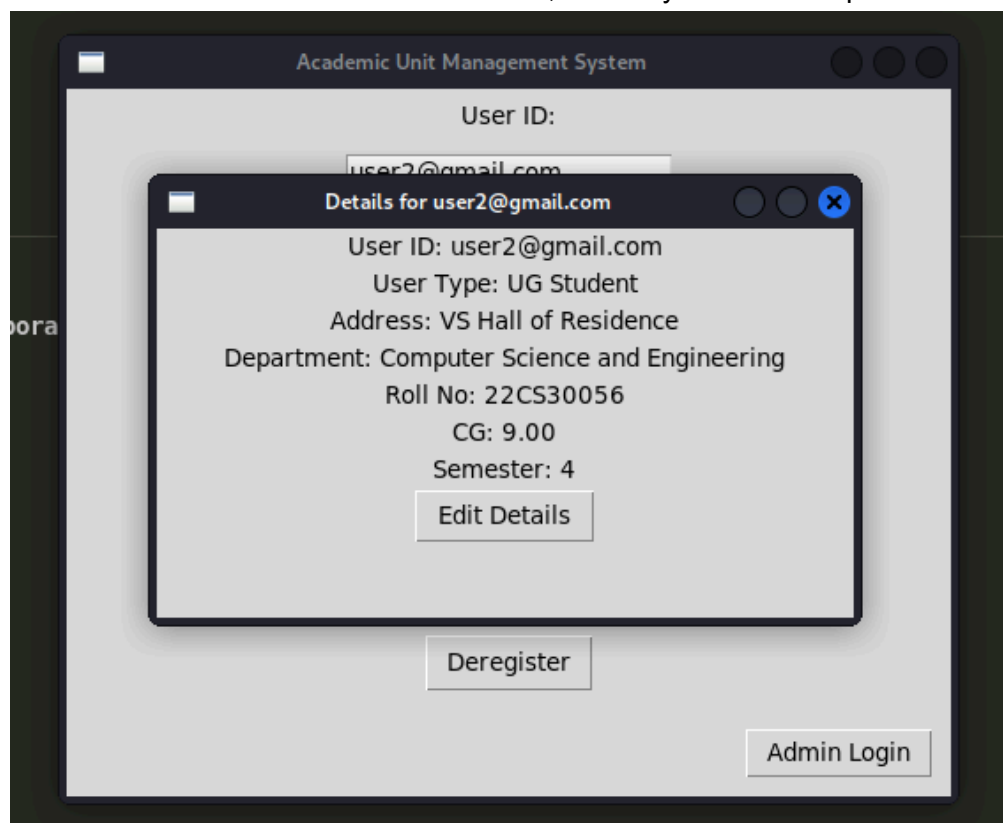
On clicking the Edit Details Button The following Window appears



After Editing the details successfully we can click on the save option, and the new details of the user get saved. On Saving we receive a pop-up message of Saved successfully like this.

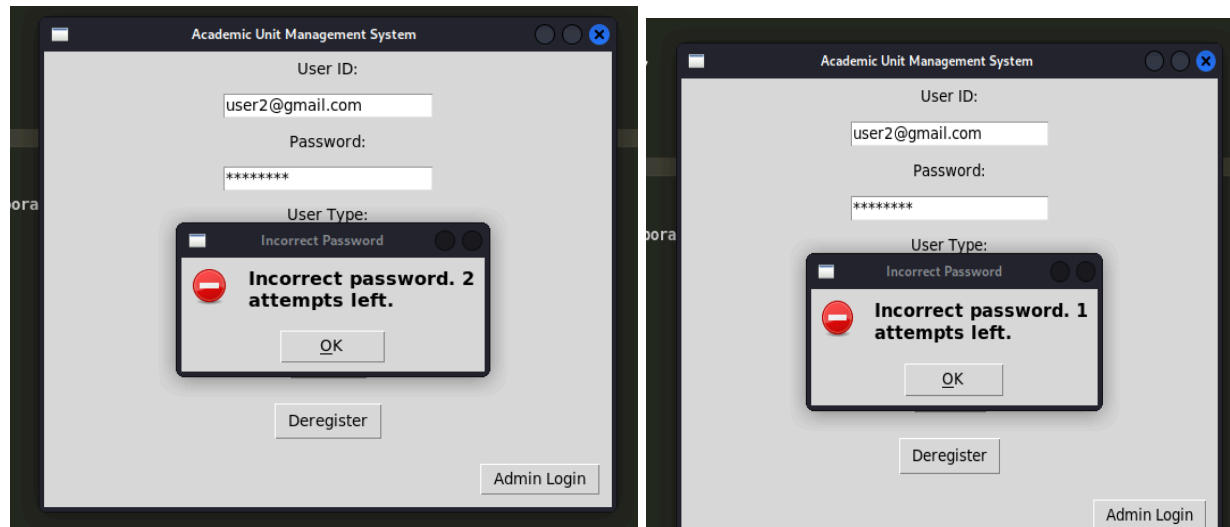


We can now check the details for the user, and they have been updated .

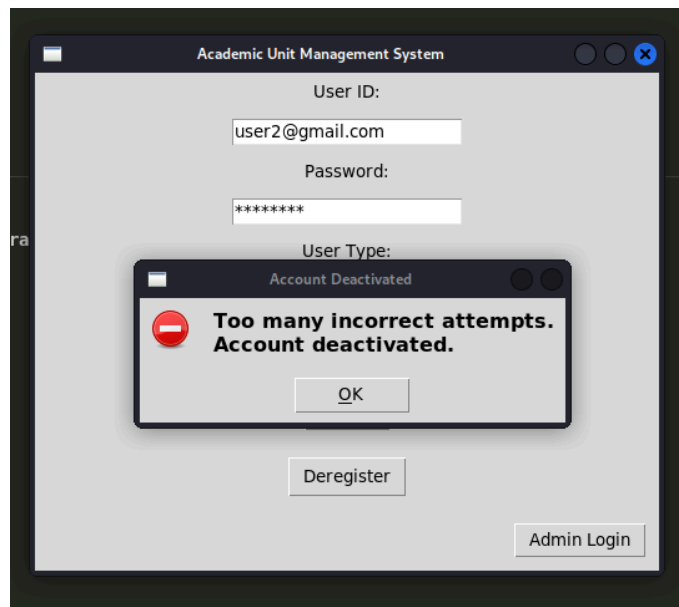


Sign-In Attempts.

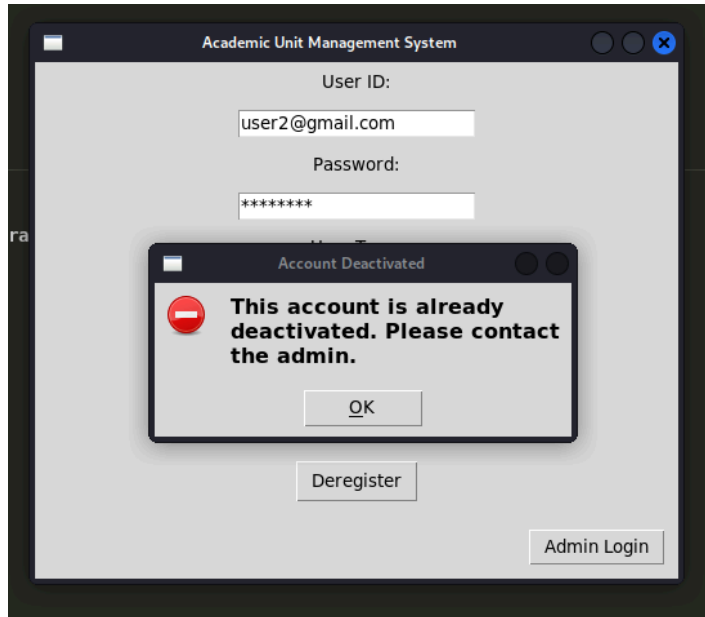
This system is been designed so that if the user enters a wrong password for the user Id, then after three unsuccessful attempts of the password, the account will get deactivated and the user will not be able to sign in to the system with the particular user ID.



After three unsuccessful attempts we receive this message.

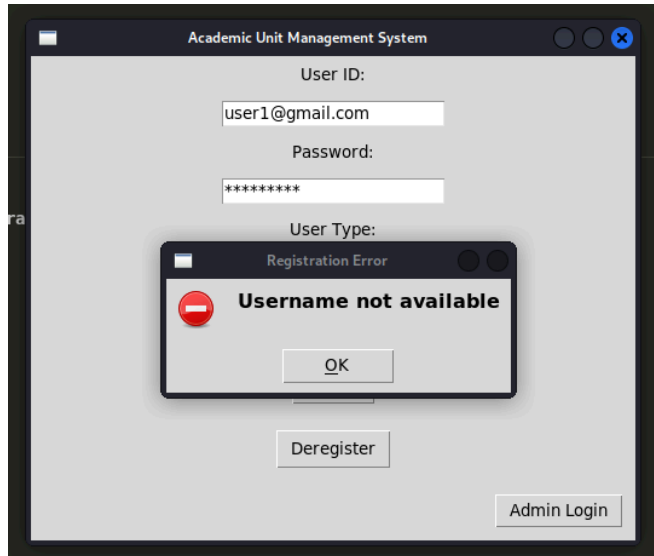


On again sign in with this user ID we receive a message like this to contact he admin of the system.



What happens when the username is not available?

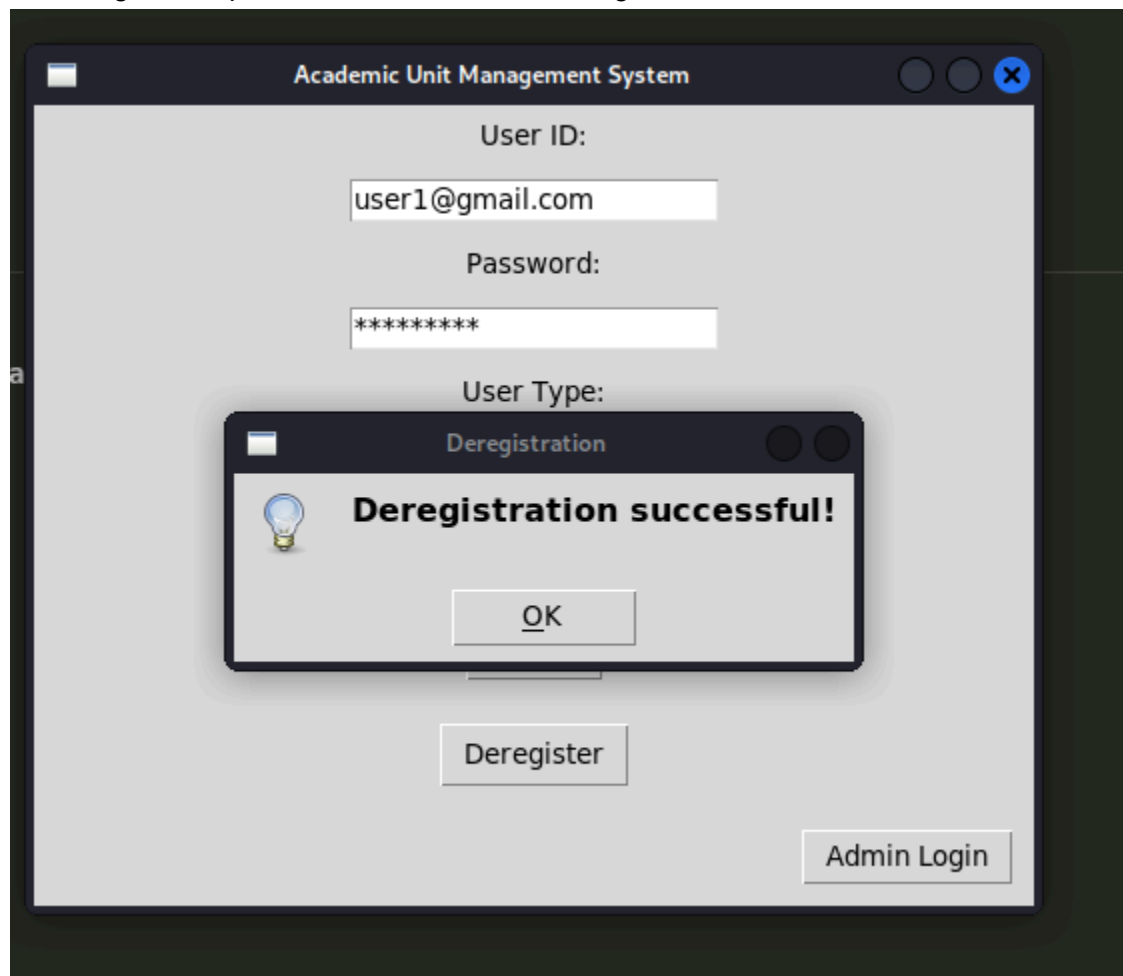
If the user tries to register with an email ID /user Id that is already present in the system, he will receive a message that the username is not available.



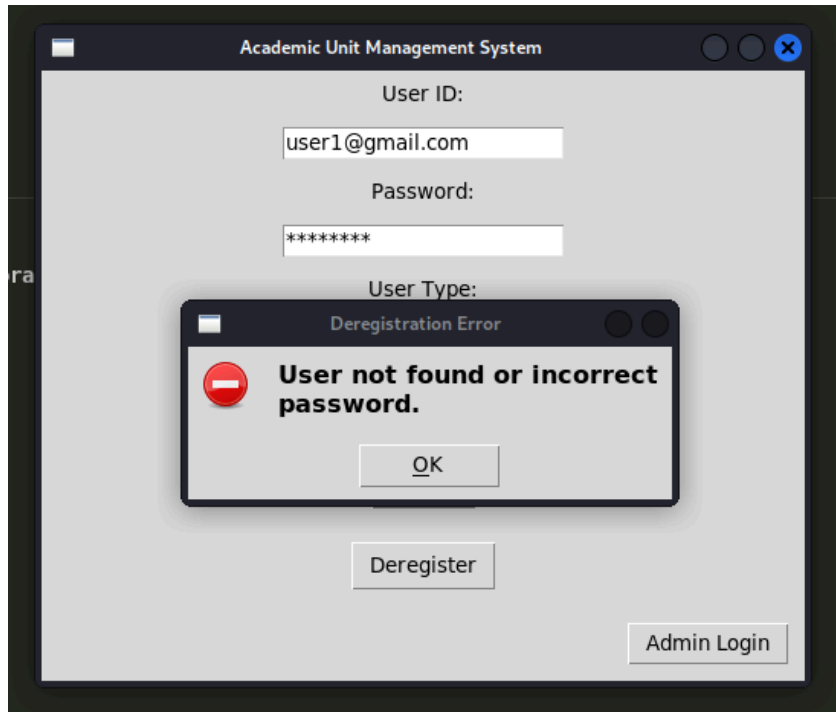
Deregistering the User

A user can delete his or her account by clicking on the deregister Button. For deregistering the account the user needs to type his/her UserId and Password and on successful submission of

the deregister request, the account of the user gets deleted from the database.



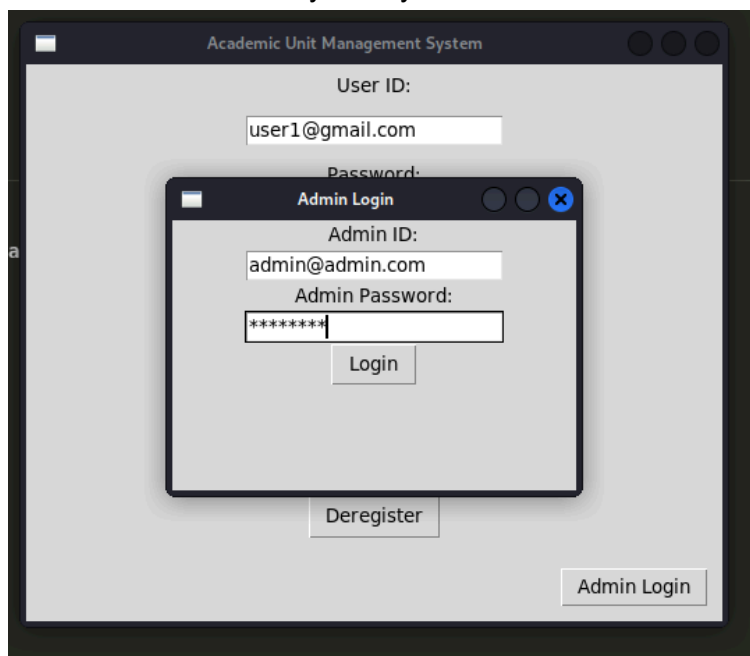
Note:-Deristration of the user is possible only if the correct user ID and Password are entered. If the wrong user ID or Password is entered then the account will not deleted from the database. Thus only valid users are able to delete their accounts.



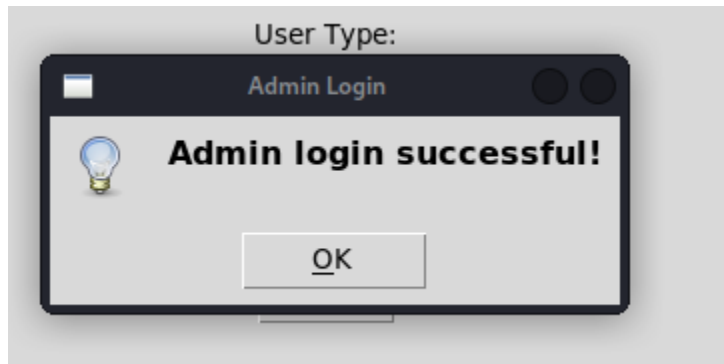
The Master Admin Functionality

In this system, there is a master admin functionality created which helps the admin of the user to view all the details of all the users.

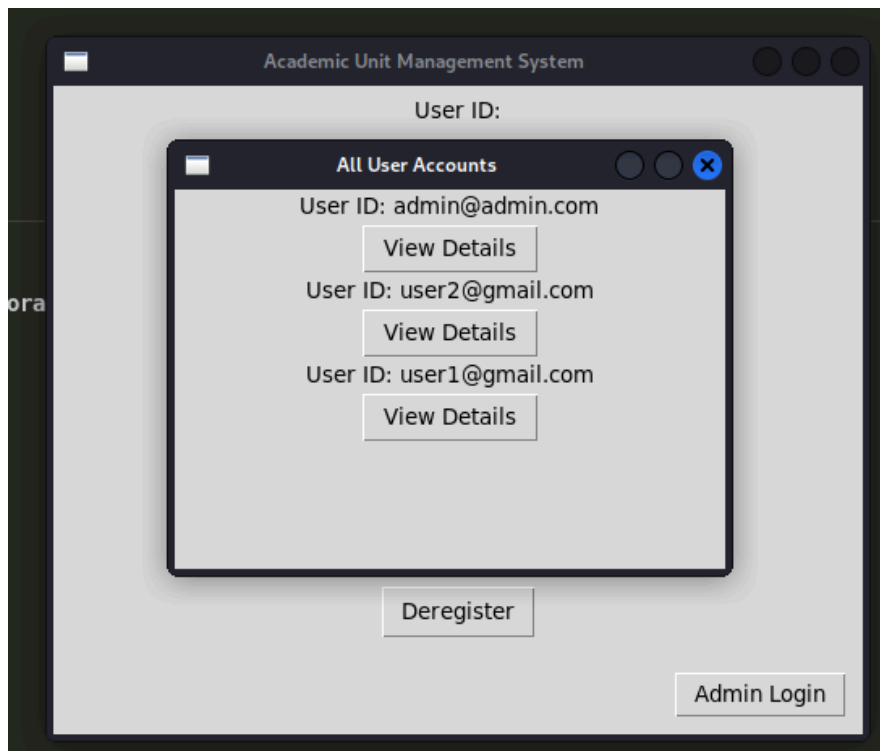
Click on the Admin login Button on the bottom right corner of the window, and enter the admin details. This functionality is only created for the admins of the system.



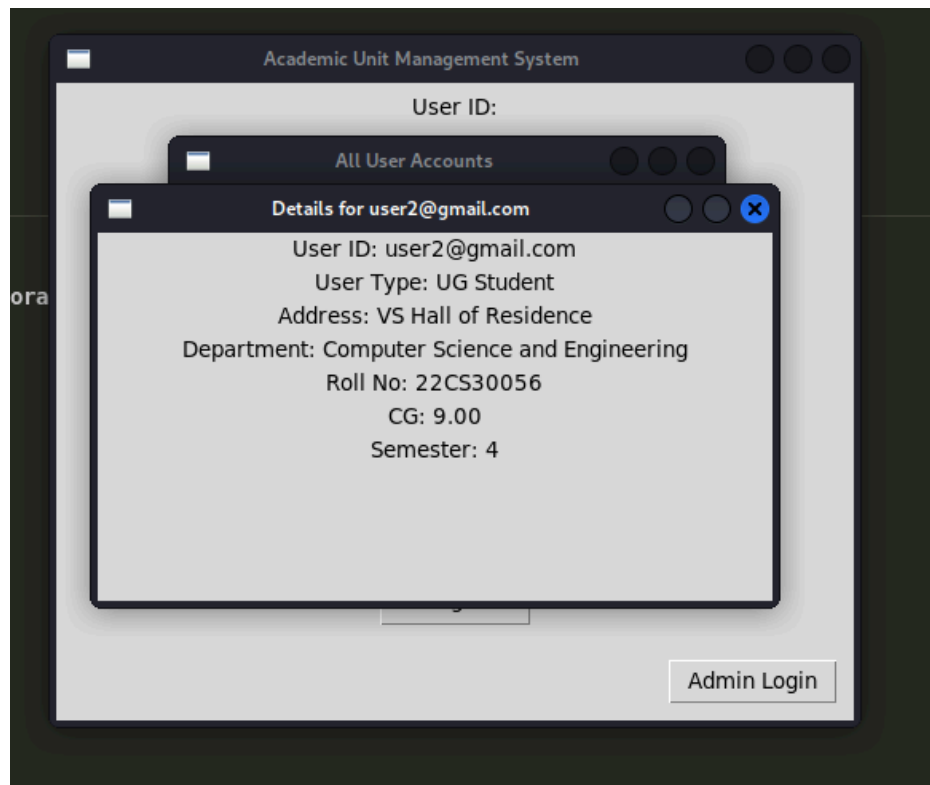
Enter the Admin User ID and Password to sign in to the admin account. On entering the correct Admin Details this message appears.



On clicking on the Ok Button the admin panel appears which shows the details of each and every user present in the database of the system. The Admin panel looks like this.



Each of the users present in the database is shown with his/her User ID to the Admin. The Admin Person can view all the details of a particular Person by clicking the View Details Button. An Example of the View Details Button is as follows:-



All the details of user2@gmail.com are visible to the Admin on clicking his View Details Button.

Validity of the Admin:-If the Admin Details are not matched then an error message pop-up window will appear on the screen like this.

