

GITHUB LINK : <https://github.com/Zishan23/AP-Assignment2.git>

Documentation for the code

CombinedDashboard Class

The **CombinedDashboard** class is the main class of your JavaFX application, responsible for creating the user dashboard and managing user interactions. It includes methods for displaying the user dashboard and handles user login and role-based access to different features.

- **displayUserDashboard(Stage primaryStage, User user):** A static method used to display the user dashboard for the provided user.

EditProfileUI Class

The **EditProfileUI** class is responsible for allowing users to edit their profiles. It provides a UI for users to update their first name, last name, username, and password. It checks for valid inputs and updates the user's profile.

- **displayEditProfileUI(Stage primaryStage, User user):** Displays the edit profile window.

AddPostUI Class

The **AddPostUI** class provides a UI for users to add new social media posts. Users can input the post content, number of likes, and number of shares. It also handles post addition and displays success or error messages.

- **displayAddPostUI(Stage primaryStage, User user):** Displays the "Add New Post" window.

RetrievePostUI Class

The **RetrievePostUI** class allows users to retrieve social media posts by their ID. Users input a post ID, and it displays post details (post ID, content, likes, shares) if the post is found. It uses the **posts** object to search for posts.

- **displayRetrievePostUI(Stage primaryStage, User user):** Displays the "Retrieve Post by ID" window.

RemovePostUI Class

The **RemovePostUI** class provides a way to remove posts by their ID. Users input the post ID to be removed, and it handles the removal. It displays success or error messages for post removal.

- **displayRemovePostUI(Stage primaryStage, User user):** Displays the "Remove Post by ID" window.

TopPostsUI Class

The **TopPostsUI** class allows users to view the top N posts based on the number of likes and shares. Users input the value of N, and it displays the top posts' details, including post ID, user ID, content, likes, and shares.

- **displayTopPostsUI(Stage primaryStage, User user):** Displays the "View Top Posts" window.

ExportPostUI Class

The **ExportPostUI** class enables users to export a post by its ID to a CSV file. Users input the post ID and select a folder for export. It provides functionality for exporting the post details to a CSV file.

- **displayExportPostUI(Stage primaryStage, User user):** Displays the "Export Post to CSV" window.

LoginUI Class

The **LoginUI** class is responsible for providing a user interface for user login and registration within your JavaFX application. It allows users to enter their username and password to log in and offers the option to register as a new user. The class manages user authentication and transitions to the user dashboard upon successful login.

Methods:

- **displayLoginUI(Stage primaryStage):** Displays the login form and handles user interactions.
 - Parameters:
 - **primaryStage:** The primary stage for displaying the login form.

User Class

The **User** class represents a user in the application, encapsulating information such as user ID, first name, last name, username, and password. This class provides methods to access and modify these attributes.

Attributes:

- **userId:** An integer that uniquely identifies the user.
- **firstName:** A string representing the user's first name.
- **lastName:** A string representing the user's last name.
- **username:** A string representing the user's username.
- **password:** A string representing the user's password.

Methods:

- **User(int userId, String firstName, String lastName, String username, String password):** Constructor for creating a **User** instance with the specified attributes.
- **getUserId():** Returns the user's unique identifier.
- **getFirstName():** Returns the user's first name.
- **setFirstName(String firstName):** Updates the user's first name.
- **getLastName():** Returns the user's last name.
- **setLastName(String lastName):** Updates the user's last name.
- **getUsername():** Returns the user's username.
- **setUsername(String username):** Updates the user's username.
- **getPassword():** Returns the user's password.
- **setPassword(String password):** Updates the user's password.
- **getFullName():** Returns the user's full name by concatenating the first name and last name.

UserManagement Class

The **UserManagement** class is responsible for managing user-related operations within the application. It maintains a list of users and provides methods to add users and retrieve users by their usernames.

Attributes:

- **users:** A list that stores instances of the **User** class.

Methods:

- **addUser(User user):** Adds a **User** object to the list of users.
- **getUserByUsername(String username):** Searches for a user in the list by their username and returns the corresponding **User** object. If the user is not found, it returns **null**.

UserRegistration Class

The **UserRegistration** class manages user registration, login, and post management functionalities. It interacts with a MySQL database to store user and post information.

Attributes:

- **DB_URL**: A string representing the URL of the MySQL database.
- **DB_USER**: A string representing the database user's username.
- **DB_PASSWORD**: A string representing the database user's password.

Methods:

- **registerUser(String firstName, String lastName, String username, String password)**: Registers a new user with the provided first name, last name, username, and password. Inserts user information into the database. Returns **true** if registration is successful, **false** otherwise.
- **loginUser(String username, String password)**: Authenticates a user with the provided username and password by querying the database. Returns a **User** object if authentication is successful, or **null** if it fails.
- **addPost(int userId, String content, int likes, int shares)**: Adds a new post to the database associated with a specific user. It records the user ID, post content, number of likes, and number of shares. Returns **true** if the post is added successfully, **false** otherwise.
- **getPostsByUserId(int userId)**: Retrieves a list of posts associated with a specific user, given their user ID. Returns a list of **Post** objects.
- **deletePost(int postId)**: Deletes a post from the database by its post ID. Returns **true** if the deletion is successful, **false** otherwise.
- **updateUserProfile(String username, String newFirstName, String newLastName, String newUsername, String newPassword)**: Updates a user's profile information such as first name, last name, username, and password. It uses transactions to ensure data integrity. Returns **true** if the update is successful, **false** otherwise.
- **getTopPostsByLikesAndShares(int topN)**: Retrieves a list of the top N posts with the highest likes and shares from the database. Returns a list of **Post** objects.

The **UserRegistration** class acts as a bridge between the application and the underlying database, providing essential user management and post management functions.