PHOTOCLOUDAPP

Sign up Class Information:

The SignupPage class represents a panel for user signup. It allows users to enter their personal information, such as name, surname, nickname, age, email, and password, and choose their user type. Users can submit the form to create an account.
This class extends JPanel and serves as a graphical user interface component for user signup.

Usage:
1. Create an instance of SignupPage.
2. Add the SignupPage instance to a parent container such as a JFrame.
3. The panel will display fields for entering personal information and choosing a user type.
4. Users can enter their information, select a user type, and submit the form to create an account.

Example:
```
SignupPage signupPage = new SignupPage();
frame.getContentPane().add(signupPage);
```

Note: The SignupPage class relies on other classes such as InvalidPasswordException, InvalidEmailFormatException,
InvalidNickNameException, EmailValidator, UserTier, and Logger for its functionality.
Make sure to have these classes available in the same package or import them as needed.

Sign Up Usage:

1. First, the sign up page will appear when the user launches the app. This page provides a form to enter the required information to create a new user account.

2. The user must fill in the personal information requested in the form. In the relevant text fields, he must enter his name, surname, username, age, e-mail address and password. This information is required to create the new user account.

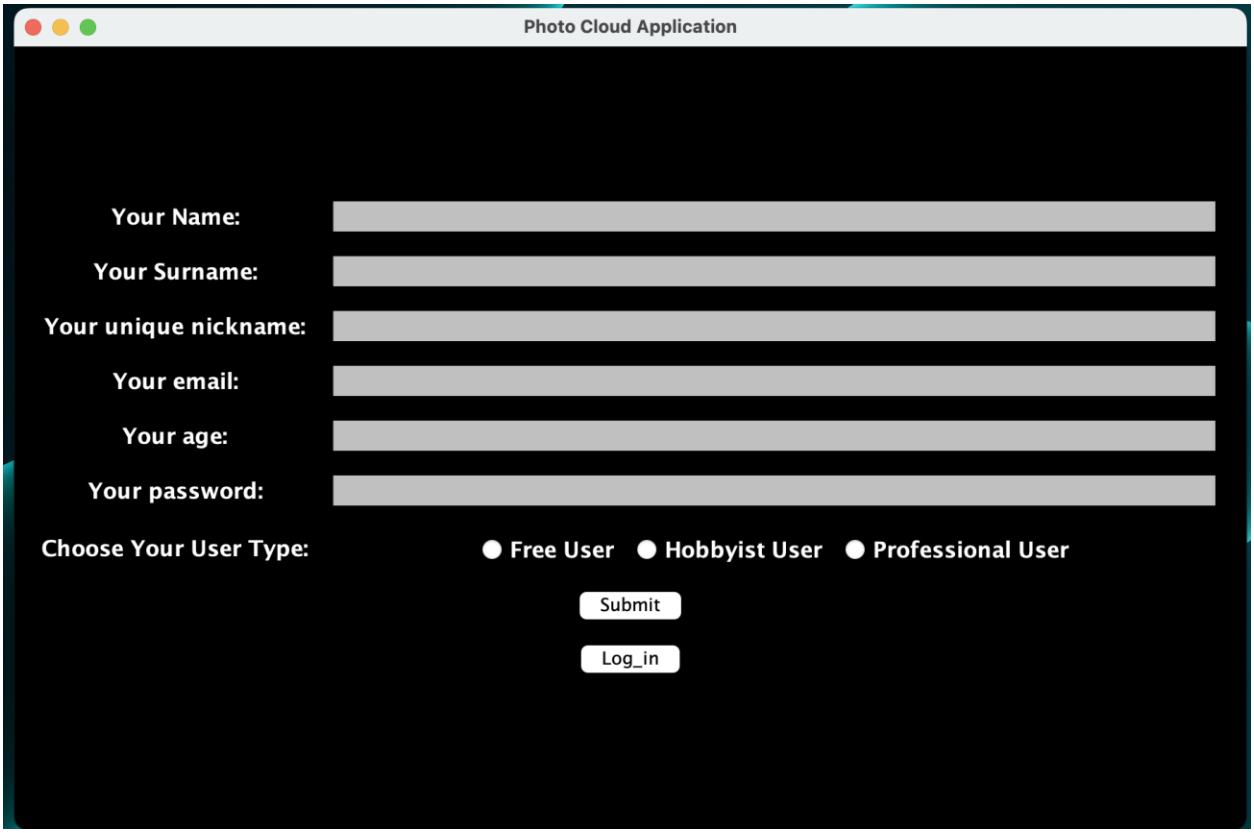
3. The user must also select the user type. User type can be defined as free user, hobby user or professional user. Users should choose their preferred user type.

4. After filling in all the necessary information in the form, the user should click on a button such as "Submit" or "Save". This button triggers the user to create a new account with the entered information.

5. When the account creation process is completed successfully, the user is shown a notification and is usually confirmed with a message such as "Account created".

6. The user may be redirected to a "Log in" page where he can log in after account creation. On this page, the user can log in with the account information they have created and start using the application.

Note: In case of related errors or incorrect entries, error messages or warnings may be displayed to the user. For example, error messages may appear if the password is insufficient or an invalid email address is entered.



LoginPage Class Information:

The LoginPage class represents the application's login page.
Allows users to enter their nicknames and passwords to log in to the system.
If login is successful, user will be redirected to Explore page.
This class extends JPanel and serves as a graphical user interface component for the login page.
Includes various Swing components such as labels, text fields and buttons to create the login form.

Use:
1. Create an instance of LoginPage and add it to a main container such as JFrame.
2. The user can enter his nickname and password in the relevant fields.
3. Clicking the "Log_in" button will try to log in the user by checking the provided credentials.
4. If login is successful, user will be redirected to Explore page.
5. If login fails due to wrong nickname or password, an error message is displayed.
6. Clicking the "Sign_up" button will redirect to SignupPage to create a new user account.

Example:
LoginPage loginPage = new LoginPage();
frame.getContentPane().add(loginPage);

Note: The LoginPage class relies on other classes such as User, UserTier and exceptions for its functionality.
Make sure these classes are in the same package or import them as needed.

Login Page Usage:

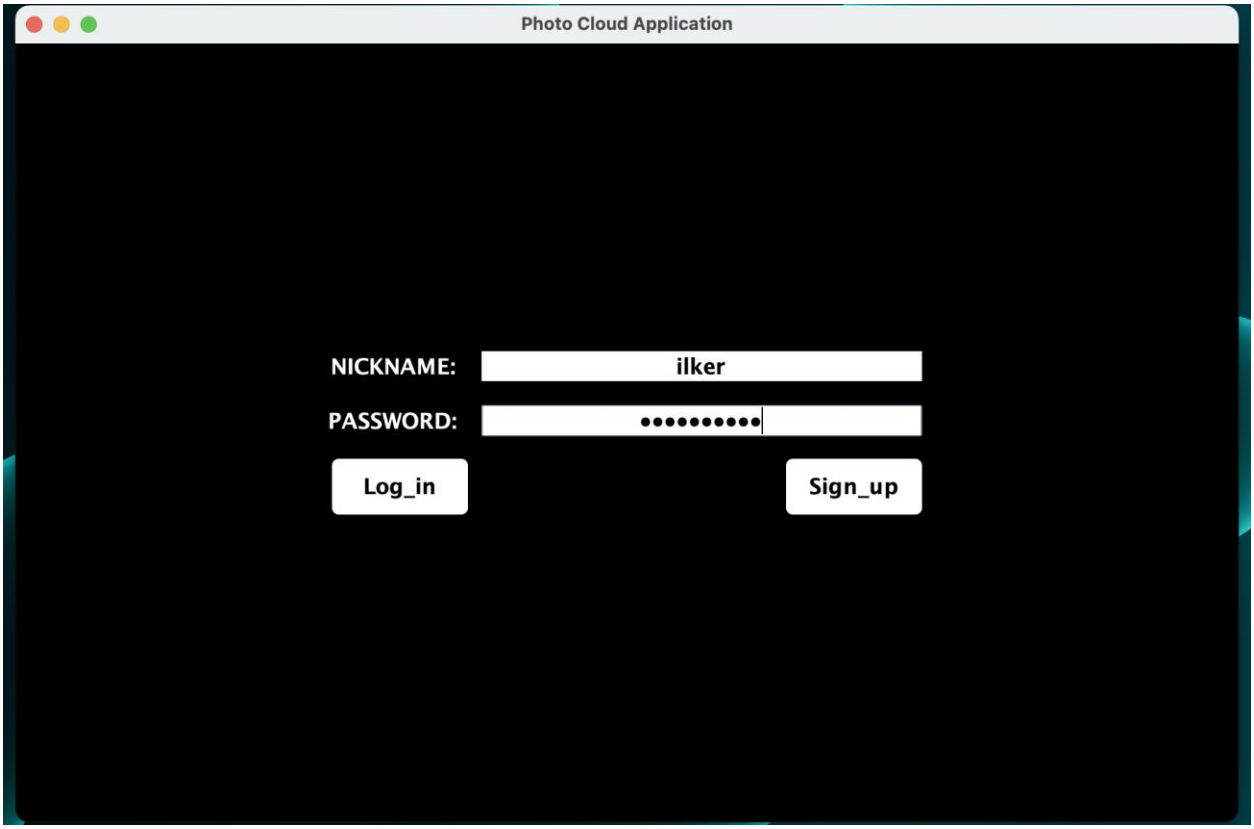
The user can login to your application and create a new account using the LoginPage and SignupPage classes. Here are the steps for the user how to use it:

Login:
1. The user opens the LoginPage and enters the relevant information in the username and password fields.
2. Clicks the "Log_in" button.
3. The application checks the credentials provided by the user and verifies their accuracy.
4. If the login is successful, the user is redirected to the Discover page and can use other features of the application.
5. If login fails, an error message is displayed and the user is prompted to enter the correct credentials again.
6.

Creating a New Account:
1. The user opens SignupPage and enters their personal information (name, surname, username, e-mail, age, etc.) into the relevant fields.
2. The user selects the type of user (Free User, Hobbyist User, Professional User, etc.).
3. Click the "Submit" button.
4. The application verifies the information provided (for example, password length check, email format check, etc.).

5. If the information is valid, the new user account is created and the user notification message is displayed.
6. If the information is not valid, an error message is displayed and the user is prompted to enter the correct information.

Users can sign in to your app and create new accounts by following these steps. These actions will enable users to use functions in your application.



Discover Page Class Information:

- The DiscoverPage class represents the panel for the Explore page in the Photo Cloud App.
- Displays shared photos and allows user interactions such as clicking on a photo or its owner's nickname, add comments and like/dislike.
- This class extends JPanel and serves as a graphical user interface component for the Explore page.
- Includes various Swing components such as buttons, labels, and panels to create the page layout and manage user interactions.

Use:
1. Create a DiscoverPage instance by passing the associated User object to the constructor. This User object represents the currently logged in user.
2. Add the DiscoverPage instance to a parent container such as JFrame.
3. Discover page, thumbnail of shared photos, owner nickname, description, interaction buttons and comments. Users can interact with photos by clicking on them, adding comments, and like/dislike.

Example:
* User user = new User ("username", "password");
* DiscoverPage discoveryPage = new DiscoverPage(user);
* frame.getContentPane().add(discoverPage);

Note: The DiscoverPage class relies on other classes such as Photo, PhotoPanel, User, and UserTier for its functionality.
Make sure these classes are in the same package or import them as needed.

Discover Page Usage:

Opening the Discover Page:
1. The user opens the page by creating an instance of DiscoverPage.
2. You pass as a parameter the User object that represents the current logged in user at the time of creation of the instance.
3. You add the DiscoverPage instance to a parent component, such as a parent component, JFrame.
4. The Discover page displays shared photos in small panels. Each panel contains a thumbnail, owner's username, description, interaction buttons, and comments.
5. Users can interact with photos, click photos, add comments and like/dislike.

The DiscoverPage class also has the ability to share photos and switch to other pages. Here is information on using these features:

1. Photo Sharing:

The Discover page has a button called "Share Picture".

- When the user clicks this button, the photo sharing process starts.
- An ActionListener is used that listens for the click event.
- In ActionListener, an instance of the ShareImage class is created and transitioned to this page.
- User can upload and share photos on ShareImage page.

2. Transition to Profile Page:

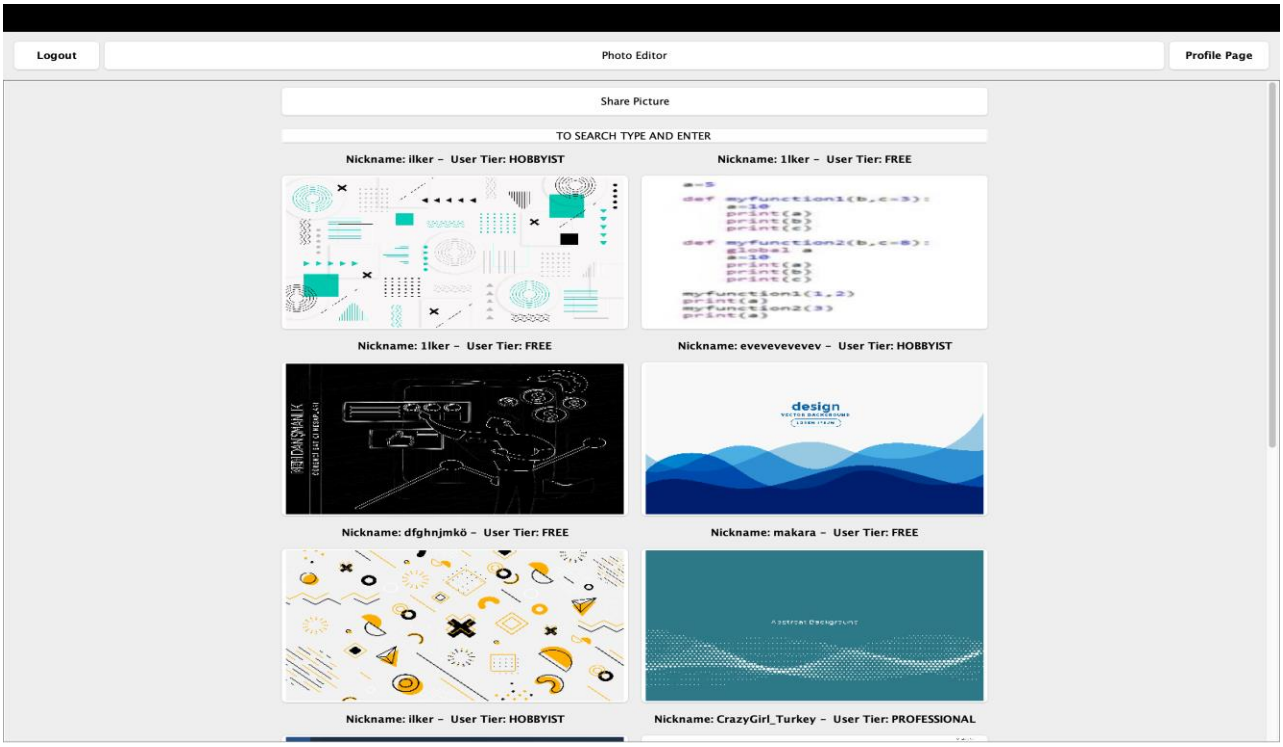
- The Discover page has a button called "Profile Page".
- The user can switch to their profile page by clicking this button.
- An ActionListener is used that listens for the click event.
- In ActionListener, an instance of the ProfilePage class is created and transitioned to this page.

3. Fotoğraf Düzenleme Sayfasına Geçiş:

- Discover sayfasında, "Photo Editor" adında bir düğme bulunur.
- Kullanıcı bu düğmeye tıklayarak fotoğraf düzenleme sayfasına geçebilir.
- Tıklama olayını dinleyen bir ActionListener kullanılır.
- ActionListener içinde, PhotoEditPage sınıfından bir örnek oluşturulur ve bu sayfaya geçiş yapılır.

4. User Search:

- The Discover page has a JTextField component with the text "TO SEARCH TYPE AND ENTER" on it.
- The user can search by entering a nickname in this field.
- ActionListener is added to JTextField to perform the search operation.
- Inside the ActionListener, it takes the entered nickname and searches for the corresponding user using the searchUser() function.
- If the user is found, an instance of the PublicProfilePage class is created and a transition to this page is made.
- If the user is not found, an error is reported.



Profile Page Class Information:

The ProfilePage class represents the user's profile page in the photo cloud application. This page displays the user's information, uploaded photos, and offers options to edit or delete photos. The class manages user interactions, such as editing the user's information or deleting photos.

The class's constructor takes a User object that represents the current user. This User object contains the user's information and photos. When the profile page is created, the user's information and photos are retrieved from this User object.

The profile page includes several components to handle user interactions. These include JTextFields for editing the user's information, JButtons for deleting or editing photos, and a JPanel for viewing the user's photos.

User can use "Upload Profile Picture" button to upload profile photo. Clicking this button gives the user the option to select a file from the file system and the selected file is set as the profile photo. The profile photo is resized by reading from the selected file and set as the profile photo of the user.

The user can also edit the profile information by clicking the "Edit and Save" button. When this button is clicked, the values inside the JTextFields are retrieved and the user's information is updated. This information is also updated in the User object, which represents the user. After the user's information is updated, a confirmation message is displayed.

The profile page also displays user uploaded photos. Photos are organized in a JPanel called userPhotosPanel and a JButton is created for each photo. The user can view the photos by clicking these buttons.

The profile page uses the file system to store the profile photo and photos of the user. The user's photo information is stored in a file named "src/imagesInfo.txt" and using this file, photos are uploaded and viewed.

Profile page provides an interactive interface that can update user's information and photos, upload photos and view photos. It provides the necessary functions for the user to edit their photos and update their profile information using the profile page.

Profile Page Usage:

1. Login to Profile Page:

After logging in to the application, the user has to click on the profile picture or username in the upper right corner of the main page to access the profile page.

2. Viewing Profile Information:

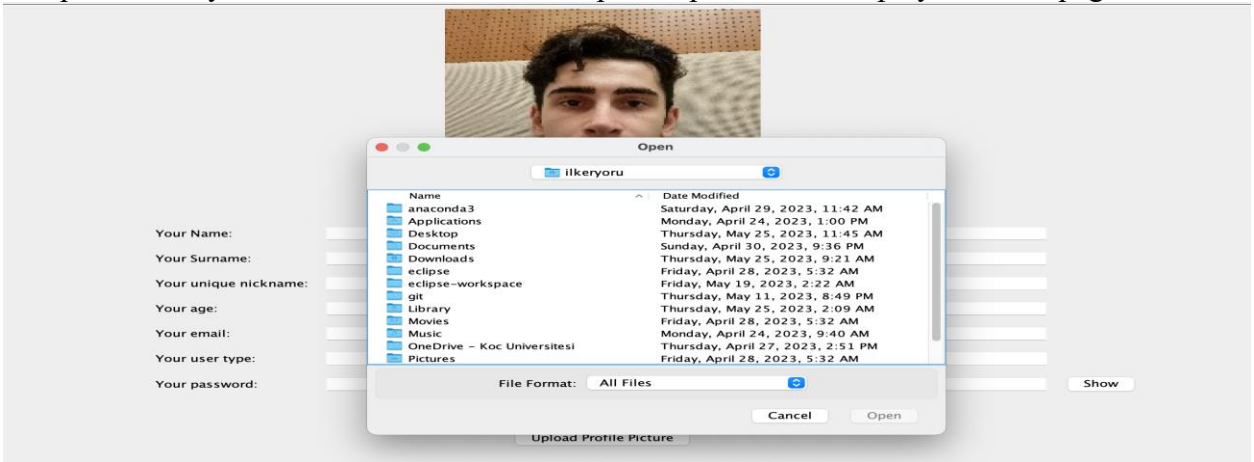
When you log in to the profile page, your profile picture and current profile information (name, surname, username, age, e-mail, user type) will be displayed.

3. Editing Profile Information:

You can edit the information in the relevant text boxes to update your profile information. You can change your information such as name, surname, age, e-mail and user type.
After completing the edits, you can save the changes by clicking the "Edit and Save" button.
A confirmation message will be displayed after the changes are saved.

4. Uploading Profile Picture:

You can click "Upload Profile Picture" button to change or upload profile picture.
The file selection dialog will open and you will be able to select an image file from your computer.
The picture file you selected will be set as the profile picture and displayed on the page.



5. Viewing Uploaded Photos:

On the profile page there is a list of photos that the user has uploaded.
You can click the corresponding button on each photo to view the photos.
Clicking on the photo will open a new page with the photo's details and editing options.

6. Photo Deletion:

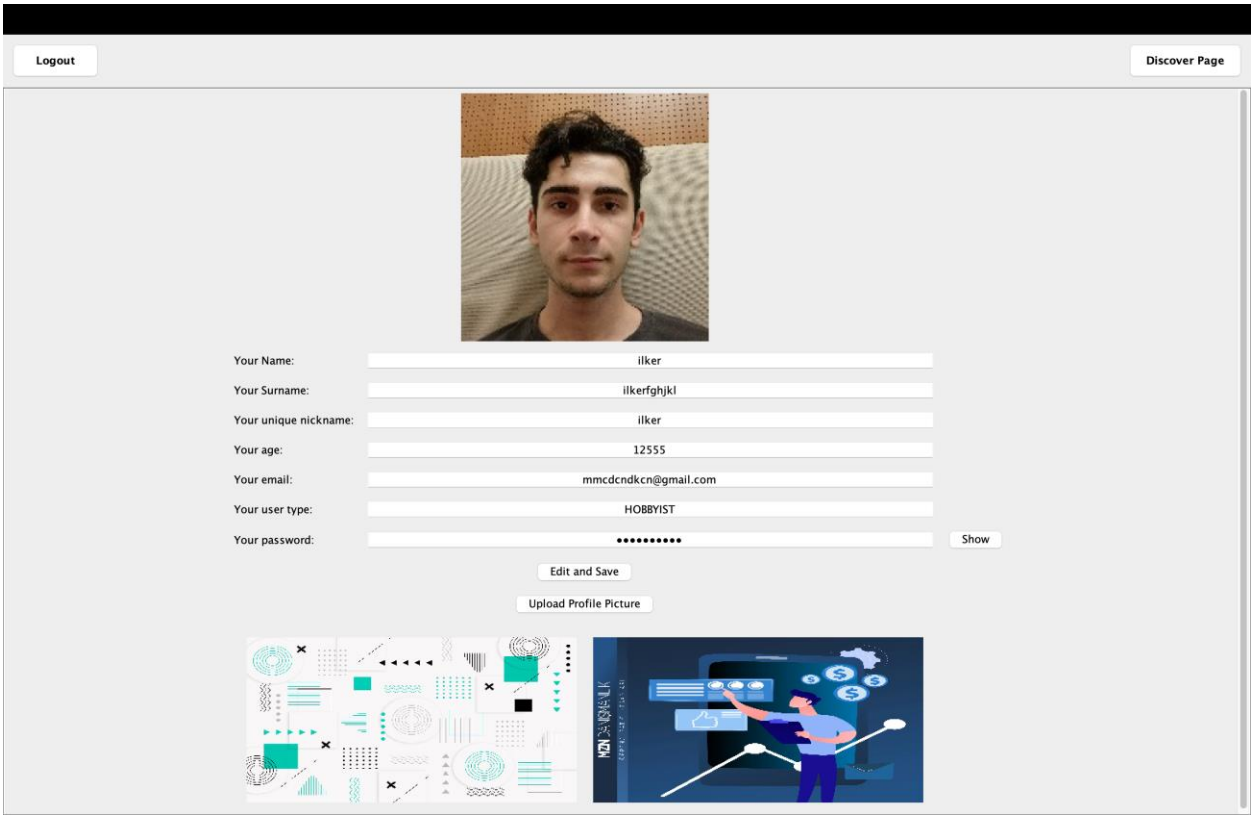
To delete photos, you can click the "Delete" or "Delete" button on the relevant photo.
Once you confirm the deletion, the photo will be permanently deleted.

7. Switching to the Discover Page:

On the profile page, you can switch to the discover page by clicking the "Discover Page" or "Discover Page" button.
On the Explore page, you can discover and interact with photos shared by other users.

8. Logout:

On the profile page, you can log out by clicking the "Logout" or "Logout" button.
After logging out, you will be redirected to the application's login page.



ShareImage Class Information:

The ShareImage class represents a panel for sharing an image. It allows the user to upload an image, add a description and save it in a database. This class extends JPanel, serving as a graphical user interface component.

Use:

1. Create a ShareImage instance with the specified user.
2. Add the ShareImage instance to a parent container such as a JFrame.
3. The panel will offer options to upload an image, add a description and submit the image.
4. When the picture is submitted, it will be saved in a database with the description provided.

Example:

```
User user = // get current user
ShareImage shareImage = new ShareImage(user);
frame.getContentPane().add(shareImage);
```

Note: The ShareImage class relies on other classes such as User and Logger for its functionality. Make sure to ensure these classes are in the same package or imported as needed.

Explanation:

- The ShareImage class represents the image sharing panel.
- A ShareImage instance is created with the User object linked to that user.
- Editing and properties of JPanel and other components are handled here.
- The shared image and description are stored in the database and related operations are performed.
- Added buttons for logout, go to profile page, and other interactions.
- User interactions and actions are handled through Swing components and event listeners.
- Image uploading is done with the file selection dialog and image processing operations.
- Error management and providing feedback to the user are performed while saving the image and description to the database.

ShareImage Usage:

1. The user can upload an image by clicking the "Upload Picture" button on the top panel. The file selection dialog will open and the user will be able to select an image file.

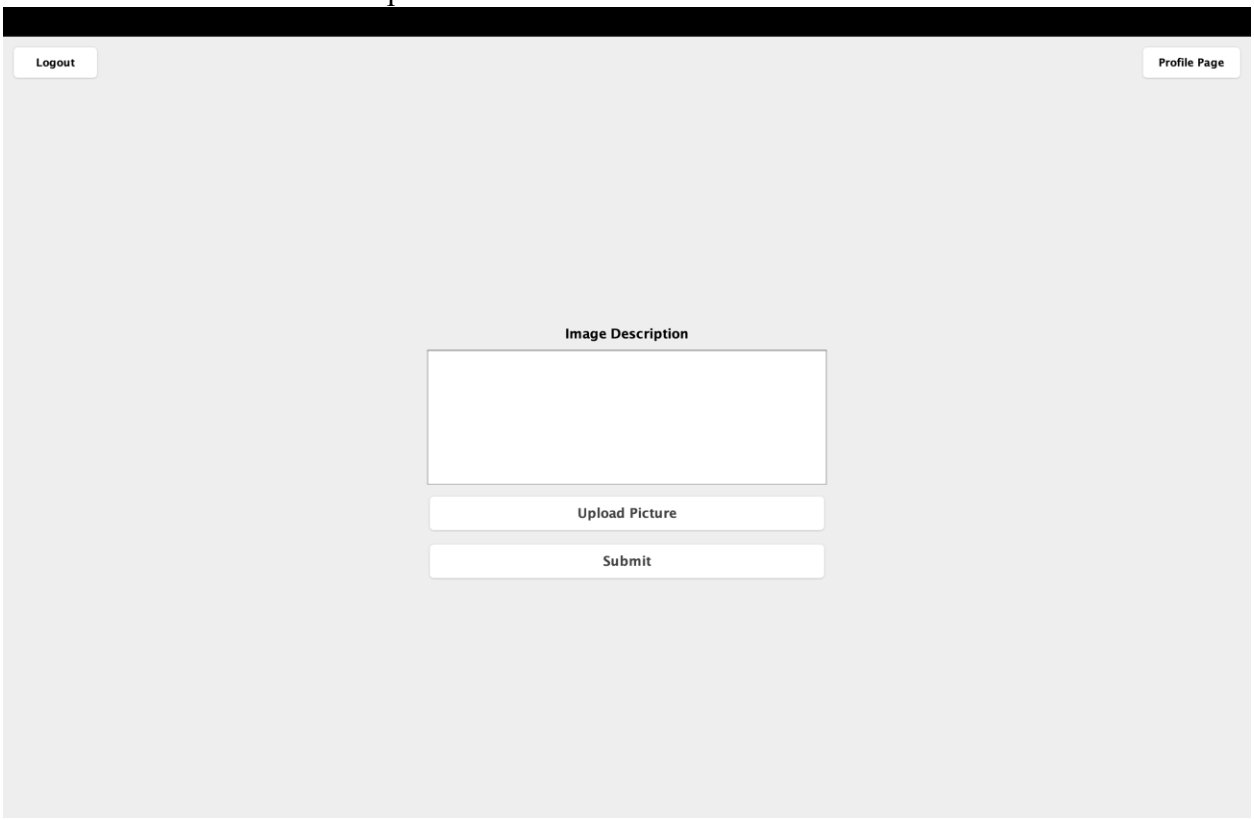
2. The selected image will be automatically resized and saved in the "src/images" folder. A copy of the file whose name begins with "resized_" will also be created. The user will be notified with a dialog if this step is successful.

3. The user can write a description in the text field under the "Image Description" title, where he can enter the description of the image.

4. The user can save the image and description to the database by clicking the "Submit" button. The user will be notified with a dialog box when this operation has been performed successfully.

5. The image and description will be saved in the file "src/imagesInfo.txt" using a specific format. This file can be used as a database where images and related information are stored.

6. After the image sharing process is complete, you can click the "Profile Page" button to direct the user to their profile.



PhotoPanel Class Informatin:

The PhotoPanel class represents a panel in which a photo is displayed using Swing components.

This panel includes the following features:

- A JLabel component that displays a thumbnail of a photo.
- A JLabel component that displays the name of the owner of the photo.
- A JTextArea component used to display the description of the photo.
- A JButton component to like the photo.
- A JButton component to dislike the photo.
- A JLabel component that displays the number of likes for the photo.
- A JLabel component that displays the number of dislikes for the photo.
- A list of CommentPanel objects displaying comments made to the photo.

The PhotoPanel class also has the following functions:

- Contains the updateImagesInfoFile() method that takes a Comment object as a parameter to update the imagesInfo.txt file.
- It includes the deleteImageInfo() method to delete a photo from imagesInfo.txt when the user is owner or administrator.
- It includes the updateTopPanel() method to update the top panel when the user owns or is the administrator.
- The PhotoPanel class can be used to display the details of a photo and allows users to interact with the photo.

PhotoPanel Class Usage:

1. PhotoPanel will display the actual image of the photo, owner information, description, like and dislike buttons, comments and user interactions.

2. The picture of the photo will be displayed automatically.

3. The information (real name and nickname) of the owner of the photo will be displayed.

4. If the photo has a description, it will be displayed in the description text area.

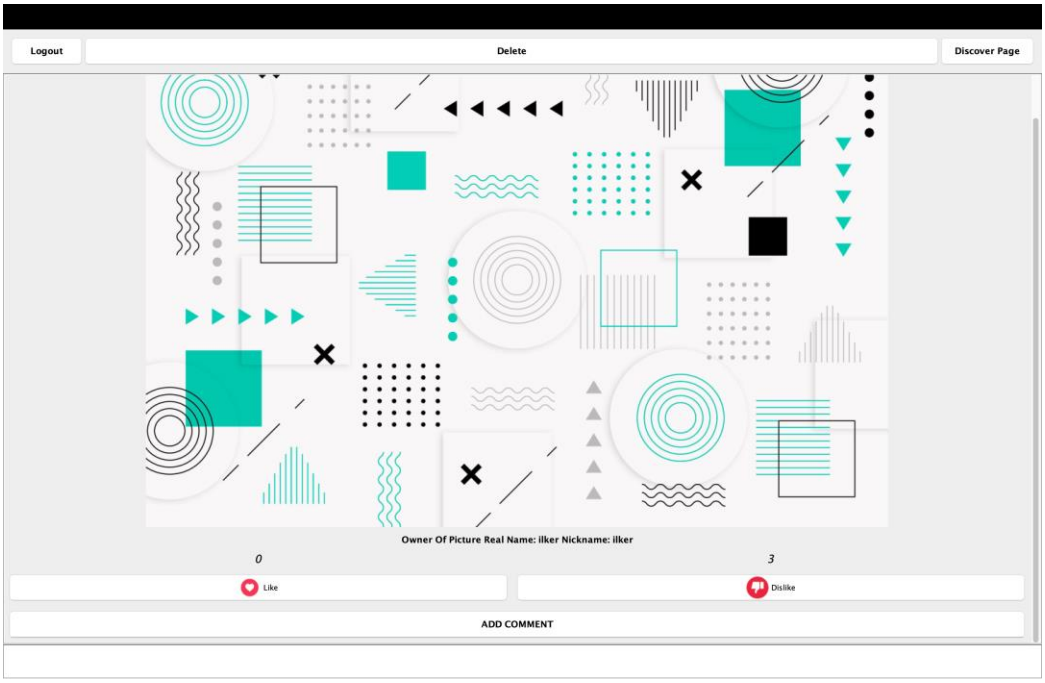
5. Like and dislike buttons will be displayed below the photo. You can like or dislike the photo by clicking these buttons. The number of likes and dislikes will be displayed next to the buttons.

6. Comments on the photo will be listed below. You can read comments and add new comments.

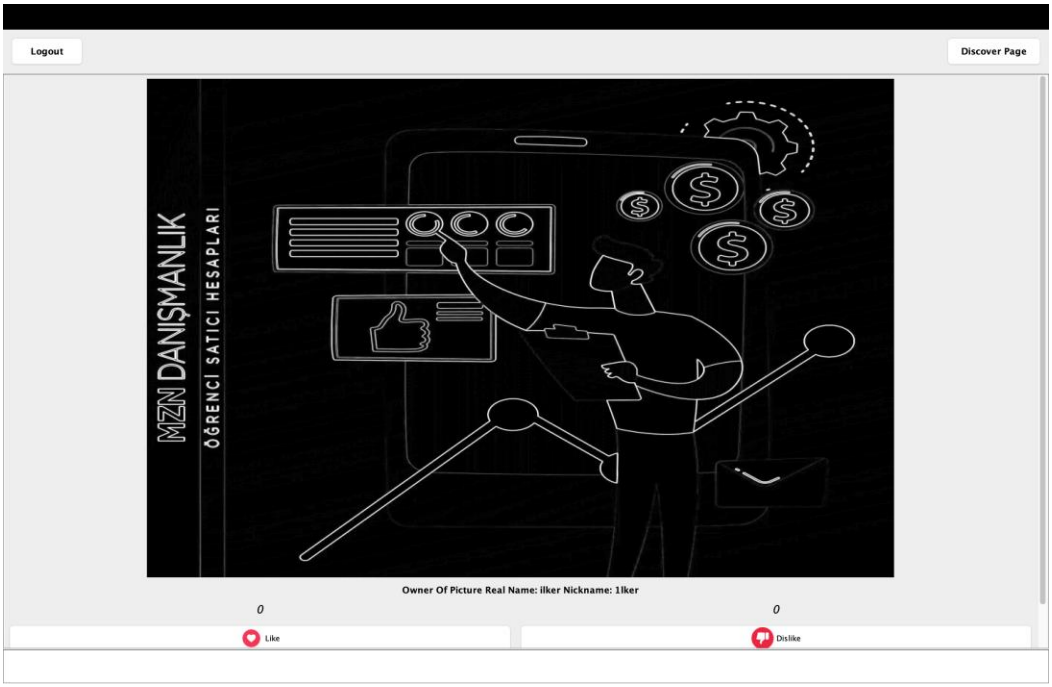
7. To add a new comment, click the "ADD COMMENT" button or type it in the comment text field and press Enter.

8. If you are the owner or administrator of the photo, you can click the "Delete" button to delete the photo.

   If the user is ADMIN or owner of Picture:



   If not the owner:



Search Feature and Public Profile Page Class Information:

The PublicProfilePage class represents a panel used to display the searched user's public profile.

This panel includes the following features:

- A JLabel component that displays the user's profile photo.
- A JLabel component that displays the user's first and last name.
- A JLabel component that displays the user's nickname.
- A JLabel component that displays the age of the user.
- A JPanel component used to display user uploaded photos.

The PublicProfilePage class also has the following functions:

- There is getPhotosForUser() method to get user uploaded photos from imagesInfo.txt file.
- There is a method loadProfilePhoto() to load and display the profile photo.
- DisplayUserPhotos() method is available to display user's photos in user photos panel.
- There is a method createPhotoLabel() to create a button for a photo and when clicking that button show the details of that photo.

The PublicProfilePage class can be used to display the searched user's public profile and provides the ability to view the user's photos.

Search Feature and Public Profile Page Usage:

PublicProfilePage is a page designed to display the public profile of the searched user. How the user can use this page can be explained by the following steps:

PublicProfilePage will open: The PublicProfilePage opens when the user clicks on the name of the user whose public profile they want to view.

User Information: When the page is opened, the user's profile photo, name, nickname and age are displayed.

Photos: At the bottom of the page is a list of user uploaded photos. Each photo is represented as a button.

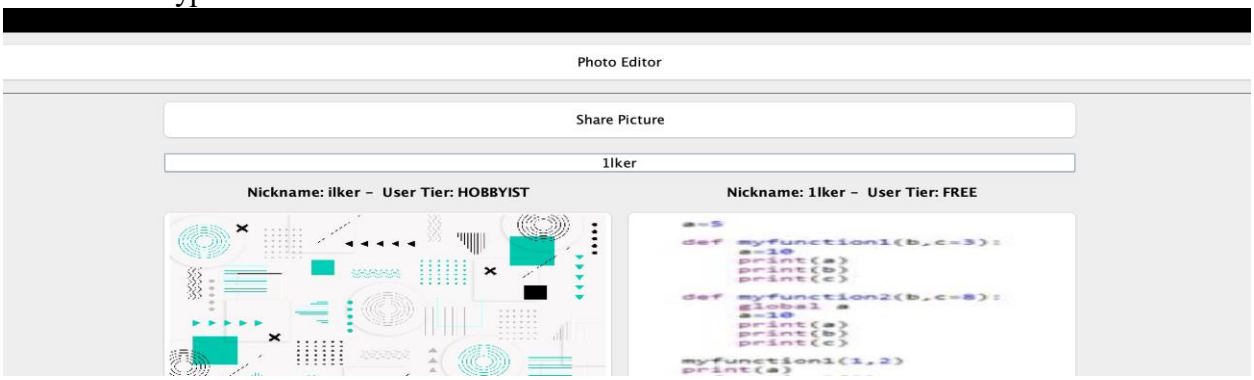
Photo Details: When the user clicks any of the photo buttons, they are taken to PhotoPanel to see the details of the selected photo. PhotoPanel includes a large image of the clicked photo, owner's information, description, likes and dislikes, and comments.

Switching to Discover Page: The user can switch to DiscoverPage by clicking the "Discover Page" button at the top of the page. This page allows the user to discover new photos and view profiles of other users.

Logout: The user can log out by clicking the "Logout" button at the top of the page and is directed to the Login page.

PublicProfilePage allows the user to view the profile information of the searched user and allows the user to discover the photos he has uploaded. It also redirects to PhotoPanel to see the details of the clicked photo.

First type and enter:



Then comes public page:

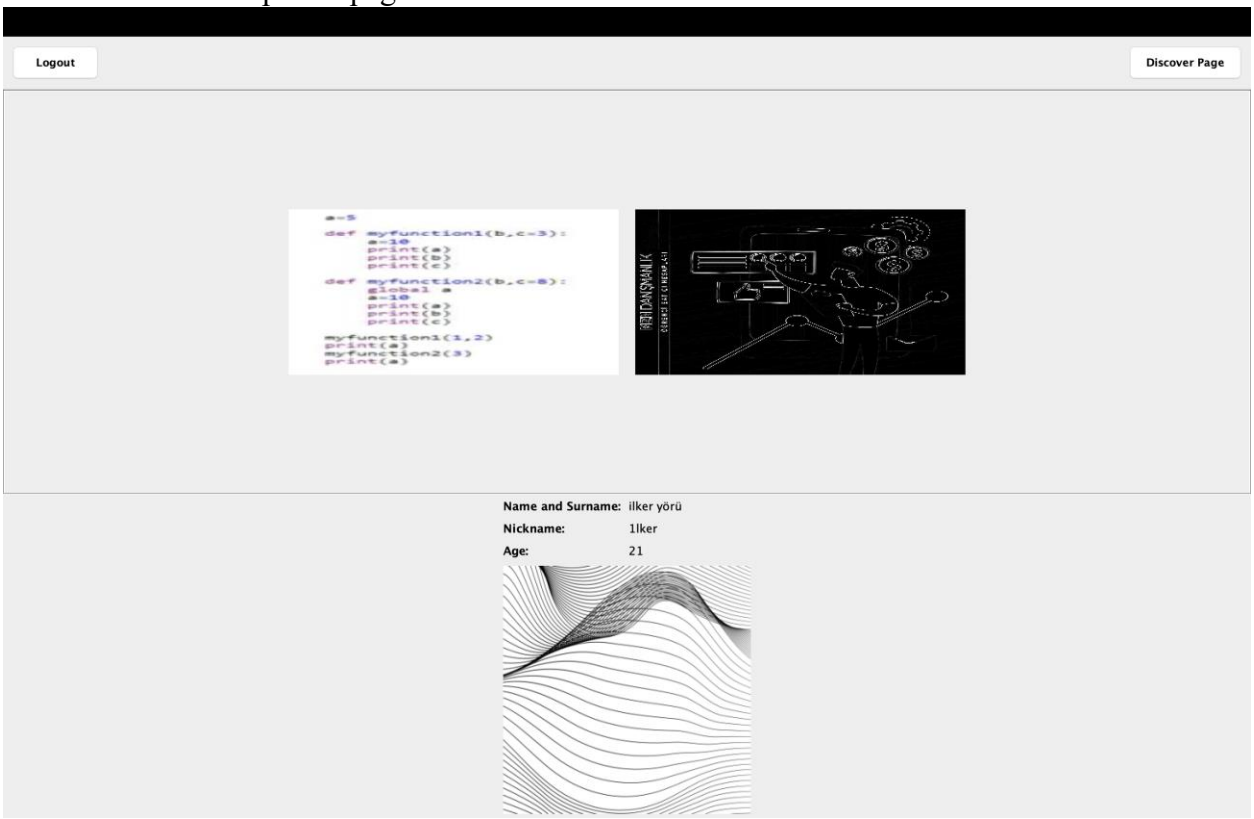


Photo Editor Page Class Information:

The PhotoEditPage class represents a JPanel used to edit photos. It allows the user to select an image, apply various filters and save the edited image.

Open PhotoEditPage: User opens this page to edit photos.

Top Panel: When the page is opened, the top panel has the "Logout", "Discover Page" and "Profile Page" buttons.

- Logout: User can log out by clicking the "Logout" button and he will be redirected to the Login page.

- Discover Page: User can switch to DiscoverPage by clicking "Discover Page" button. This page allows the user to discover new photos and view profiles of other users.

- Profile Page: User can switch to ProfilePage by clicking "Profile Page" button. This page allows the user to view their own profile.

Image Selection: The user can initiate an image selection process by clicking the "Select Image" button. When a file is selected, ImageMatrix is updated according to the selected image and the new image is displayed.

Filters: In the center of the page are buttons to apply different filters.

- Blur Filter: User can apply blur filter to selected image by clicking "Blur Filter" button. This filter blurs the image.
- Sharpen Filter: User can apply a sharpening filter to the selected image by clicking the "Sharpen Filter" button. This filter sharpens the image.
- Contrast Filter: User can apply contrast filter to selected image by clicking "Contrast Filter" button. This filter increases the contrast of the image.
- Edge Detection Filter: User can apply edge detection filter to selected image by clicking "Edge Detection Filter" button. This filter highlights the edges of the image.
- Grayscale Filter: User can apply a grayscale filter to the selected image by clicking the "Grayscale Filter" button. This filter makes the image black and white.
- Brightness Filter: User can apply a brightness filter to the selected image by clicking the "Brightness Filter" button. This filter adjusts the brightness of the image.
- Save Image: User can save edited image by clicking "SAVE" button. A file will be selected for the save operation and the image file will be saved.

The PhotoEditPage class allows the user to edit photos. The user can select an image, apply filters and save the edited image.

Photo Editor Page Class Usage:

PhotoEditPage is a page where the user can edit photos. The user can use this page by following the steps below:

Page Opening:

- To access PhotoEditPage, the user must use the corresponding menus or navigations in the application.
- When PhotoEditPage opens, an image is displayed with an image select button and apply filters buttons.

Image Selection:

- Click the "Select Image" button. The file explorer window will open to select a file.
- In the file explorer window, select the desired image and click the "Open" button.
- The selected image will be displayed in the image area on the page.

Filter Application:

- You can edit the image using the buttons for different filters.
- For example, you can apply a blur filter to the image by clicking the "Blur Filter" button.
- You can also apply different filters using other buttons such as "Sharpen Filter", "Contrast Filter", "Edge Detection Filter", "Grayscale Filter" and "Brightness Filter".
- When you click on the filter buttons, the selected filter will be automatically applied to the image and the result will be instantly displayed in the image area.

Saving the Image:

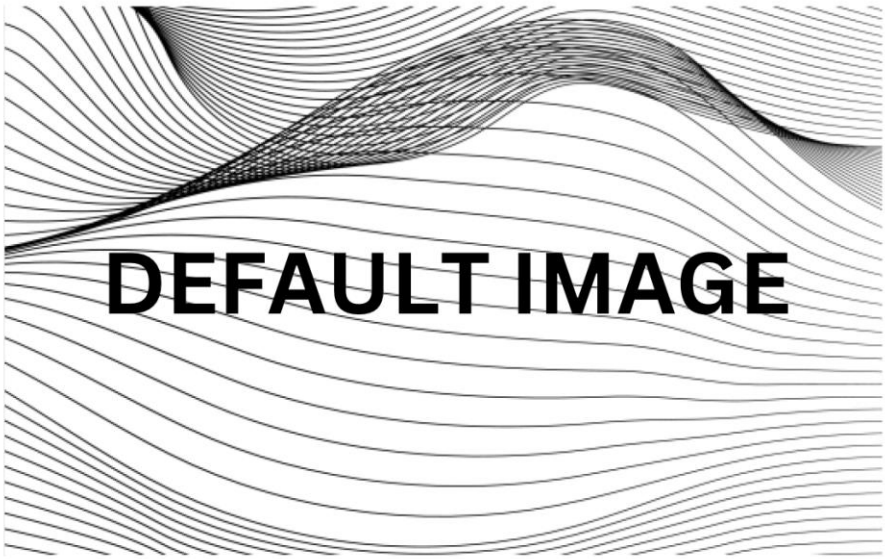
- Click the "SAVE" button to save your edited image.
- A save file window will open for the save operation.
- In the file save window, select the location and filename where you want to save the image, then click the "Save" button.
- Your edited image will be saved in the location you selected.

Leaving the Page:

- Click the "Logout" button to log out. This will redirect the user to the login page.
- You can switch to Discover Page by clicking the "Discover Page" button. This page lets you discover new photos.
- You can switch to Profile Page by clicking the "Profile Page" button. This page allows you to view your own profile.

By following these steps, the user can open PhotoEditPage, select an image, apply filters and save the edited image.

Logout | Görsel Seç | Discover Page | KAYDET | Profile Page



Apply Blur Filter | Apply Sharpen Filter | Apply Contrast Filter | Apply Edge Detection Filter | Apply Grayscale Filter | Apply Brightness Filter