

Programming Report

Eray Altun 77808

May 18, 2022

Contents

Guideline	2
Execution Steps GUI	2
Login/Register	2
Homepage	3
Profile Page	4
Groups Page	6
Users Page	7
Project Design	8
File Structure	8
UI Builder	9
Absolute Layout	9
Implemented types	10
frontend	10
backend	10
Reference	11

Guideline

Execution Steps GUI

Login/Register

When the application is launched you will be prompted to login.

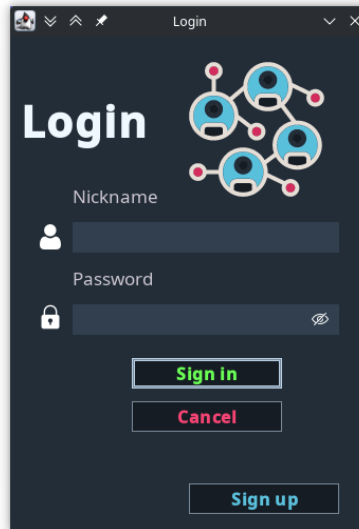
A dark-themed login window titled "Login". It features a large "Login" text on the left and a network icon of four blue circles connected by lines on the right. Below the title, there are two input fields: "Nickname" with a person icon and "Password" with a lock icon and a toggle eye icon. At the bottom, there are three buttons: "Sign in" (green text), "Cancel" (red text), and "Sign up" (blue text).

Figure 1: Login Prompt

When register is clicked, a register JFrame will be shown.

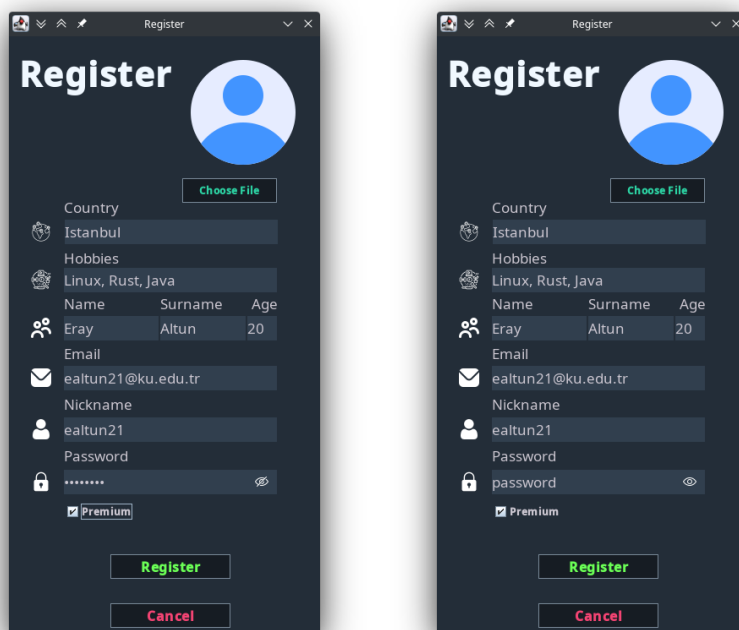
Two identical dark-themed register windows titled "Register". Each window features a large "Register" text on the left and a blue circle profile icon on the right. Below the title, there is a "Choose File" button. The form includes several input fields: "Country" (with a location pin icon and "Istanbul" text), "Hobbies" (with a book icon and "Linux, Rust, Java" text), "Name" (with a person icon and "Eray" text), "Surname" (with "Altun" text), "Age" (with "20" text), "Email" (with an envelope icon and "ealtun21@ku.edu.tr" text), "Nickname" (with "ealtun21" text), and "Password" (with a lock icon and "password" text). There is also a "Premium" checkbox which is checked. At the bottom, there are two buttons: "Register" (green text) and "Cancel" (red text).

Figure 2: Register Prompt

Homepage

Once logged in, you will be greeted with the home page. The homepage contains:

- Search (title based, case sensitive)
- Posts from users you follow.
- Posts from groups you follow.

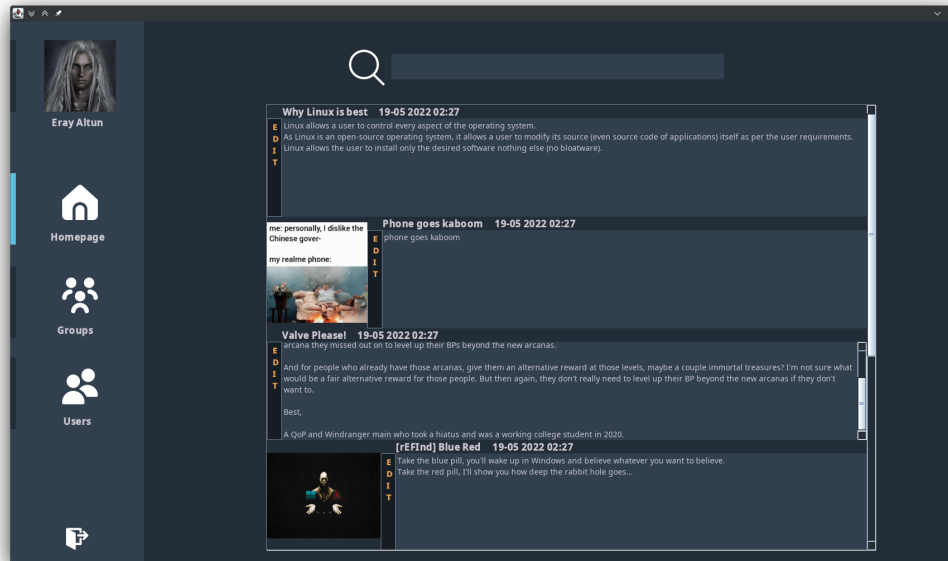


Figure 3: Homepage

Profile Page

You may go to the profile page to view your profile.

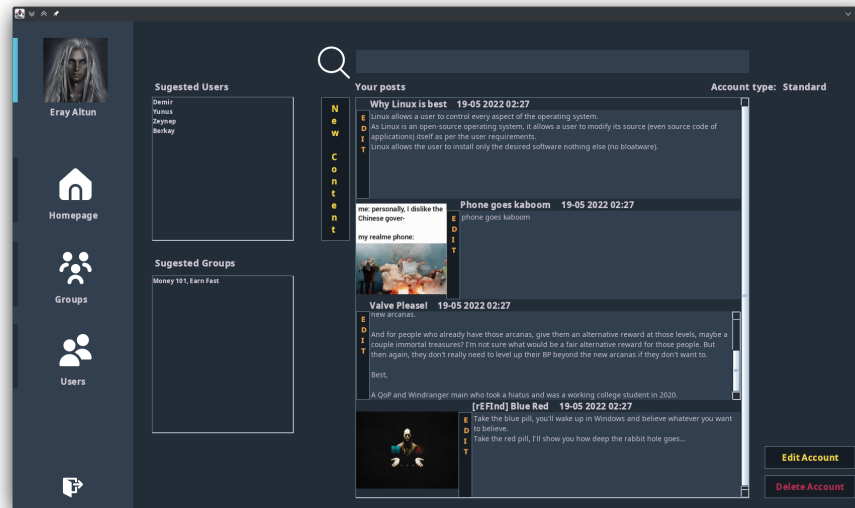


Figure 4: Profile Page Stander User

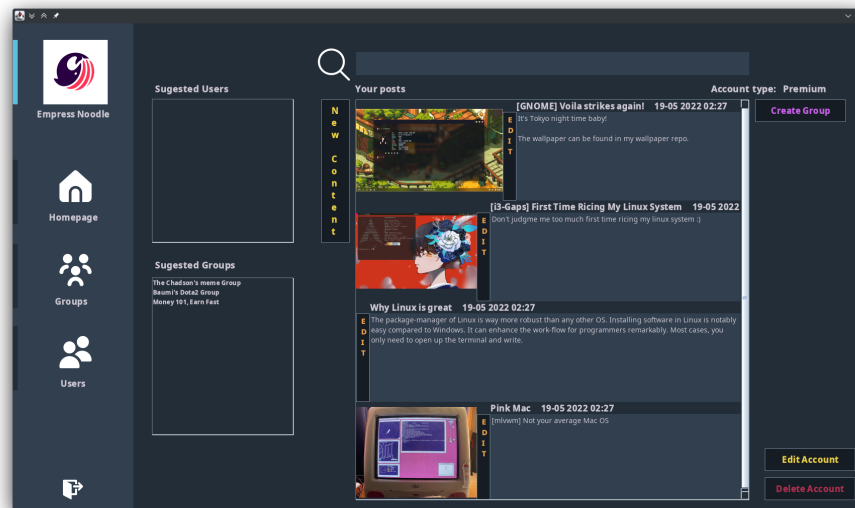


Figure 5: Profile Page Premium User

You may create a group if you are a premium user.

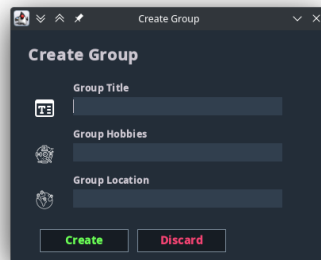


Figure 6: Group Creator

You may also edit your account.

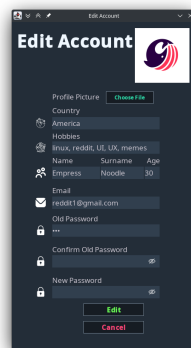


Figure 7: Account Editor

You can create your own personal contnet.

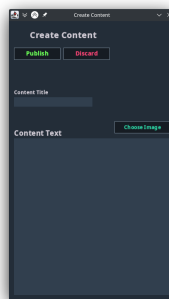


Figure 8: Cotent Creator

Groups Page

- You can join, leave groups.
- Edit your own content on the group page.

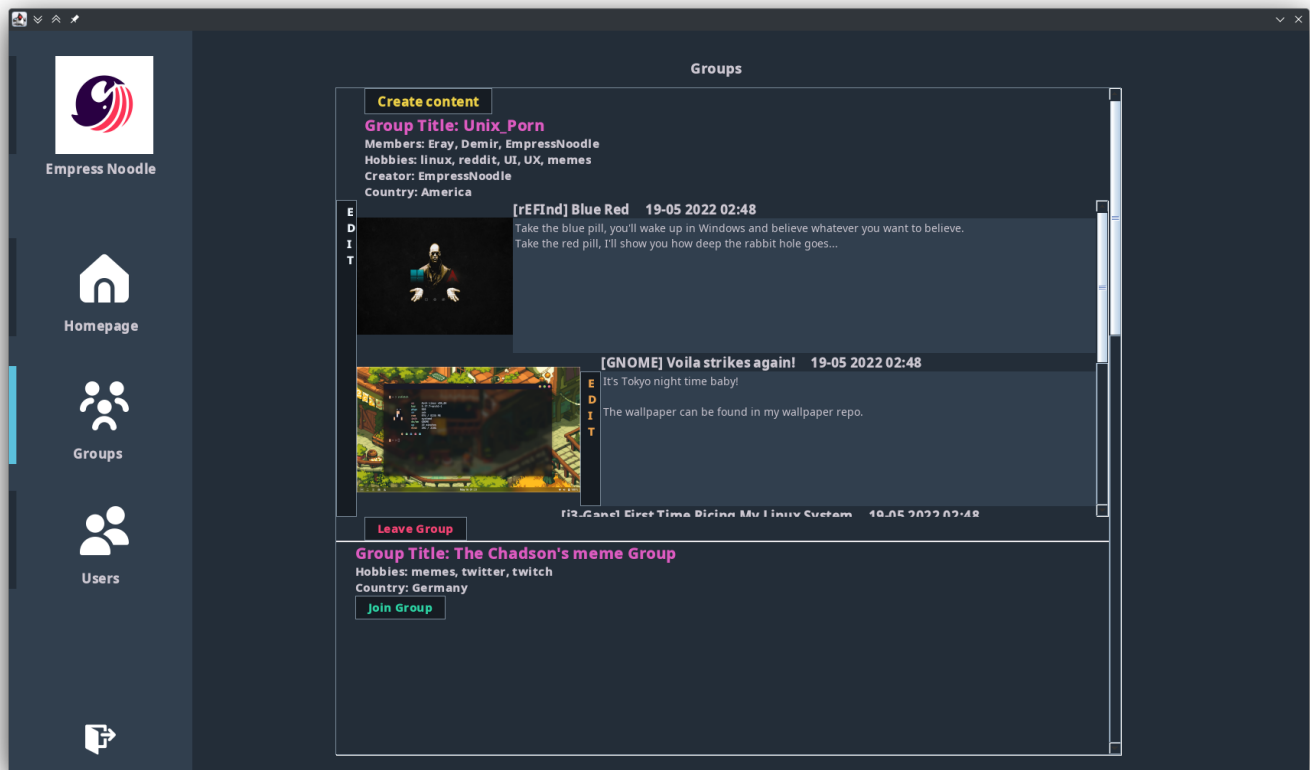


Figure 9: Group Page

Users Page

- You can follow, and unfollow users in the users page.
- You can search for users.

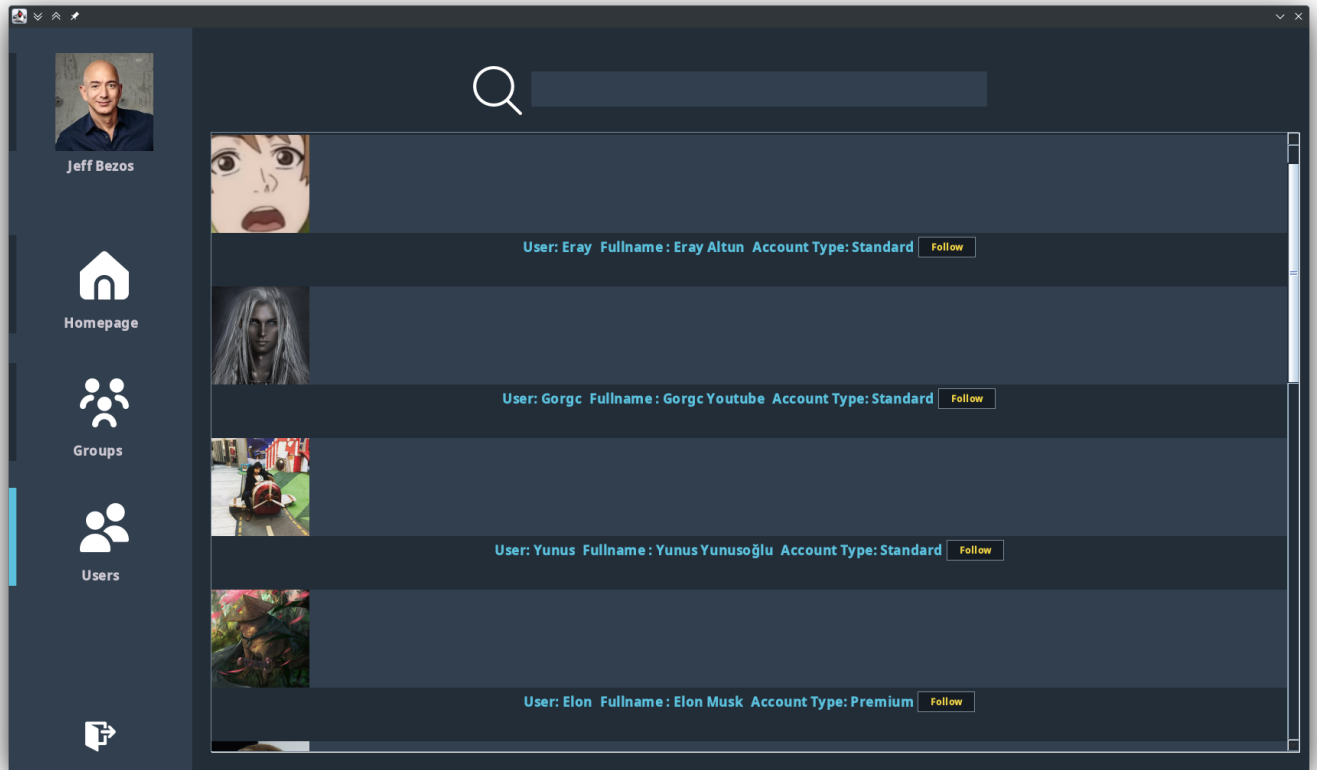


Figure 10: Users Page

Project Design

File Structure

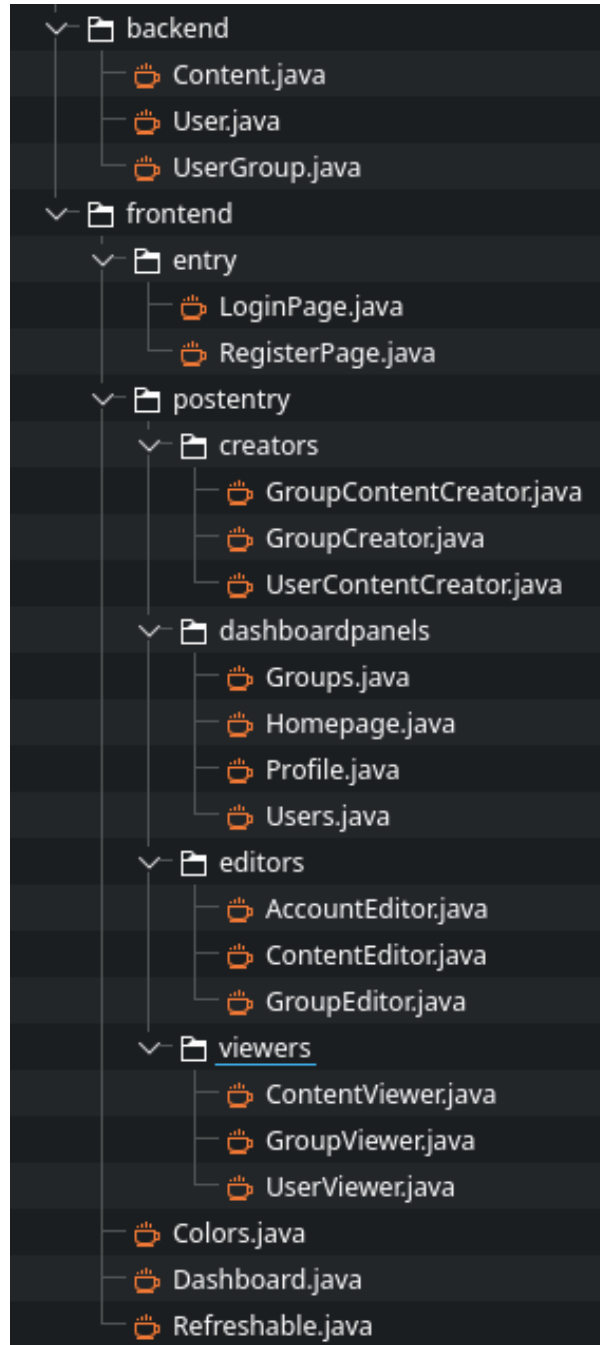


Figure 11: File Structure

UI Builder

I used the UI builder in order to set the position of the panels, especially when using an absolute layout. I benefit from the UI builder when trying to discern what methods are available for elements. I have modified the auto-generated code to suit my needs, at times completely rewriting it.

After using an absolute layout for a considerable amount of time, I have learned the following:

Absolute Layout

Pros:

- Pixel-Perfect Alignment
- Easy to design layout

Cons:

- Not Resizeable
- Breaks when OS uses any scaling

Overall thought:

- Not recommended

Implemented types

The program is separated into two parts:

frontend

- Defines the GUI aspects of the program.

Is separated into two parts:

entry

- Contains Login and Register JFrames.

postentry

- Contains Creators: That instantiate objects of the backend
 - UserGroups
 - UserContent
 - GroupContent
- Contains Editors: That modify the objects created by the creators
 - Account Editor
 - Content Editor
 - Group Editor
- Contains Viewers: Are JPanels that define how the objects created by the creators should be displayed (similar to toString methods but for GUI)
 - Content Viewer
 - Group Viewers
 - User Viewer
- Dashboard Panels: The panels that are held inside the dashboard
 - Groups
 - Profiles
 - Users
- Colors: Simple Class that extends swings Color class in order to make a custom color palette.
- Dashboard: A JFrame that holds login on switching between the dashboard panels.
- Refreshable Interface: An interface that defines refresh.

backend

- Defines data classes that will be used by the frontend.
- Defines logic on how the data should be manipulated.

Examples:

- Content Class
- User Class
- UserGroup Class

Reference

- <https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>
- <https://thebadprogrammer.com/swing-uimanager-keys/>
- <https://alvinalexander.com/java/java-uimanager-color-keys-list/>
- <https://www.youtube.com/watch?v=1nEluKUOic8>
- <https://www.geeksforgeeks.org/functional-programming-in-java-with-examples/>
- <https://www.reddit.com/r/unixporn/>
- <https://www.flaticon.com/>
- <https://krita.org/en/>
- <https://stackoverflow.com/questions/10850753/how-to-convert-a-liststring-into-a-comma-separated-string-without-iterating-li>
- <https://stackoverflow.com/questions/29318404/how-to-set-javafx-password-field-echo-char>
- <https://www.youtube.com/watch?v=dCIef0griJ0>
- <https://www.youtube.com/watch?v=Kmgo00avvEw>
- <https://docs.oracle.com/javase/8/docs/api/java/util/Collections.html>
- <https://docs.oracle.com/javase/8/docs/api/java/lang/Comparable.html>
- <https://docs.oracle.com/javase/8/docs/api/java/util/Iterator.html>
- <https://docs.oracle.com/javase/8/docs/api/java/util/Comparator.html>
- <https://docs.oracle.com/javase/8/docs/api/java/util/List.html>
- <https://docs.oracle.com/javase/8/docs/api/java/util/Set.html>
- <https://docs.oracle.com/javase/8/docs/api/java/util/Map.html>