GYM APP

OOP Project 2022

Abstract

Java app meant for displaying useful gym-related information to users

Prof: Bindea Bogdan

Cristian Adrian Dora

doracristian@yahoo.com

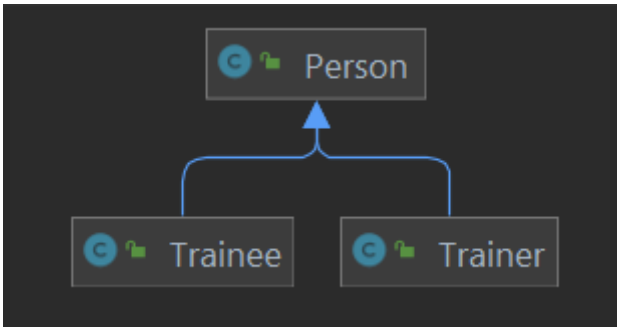
Introduction

The app helps people who want to start going to the gym or simply just exercising. After the user creates an account, he will see a new window where he can buy supplements and request a training regime from one of the available trainers. Also, the user has a bonus feature. After all the effort put into losing weight, the app can reflect the user's progress. The user has the option to change the weight displayed on the screen with his new weight.

For this application, besides Java programming language, I also used a personal Postgres database and Swing for GUI.

Implementation details

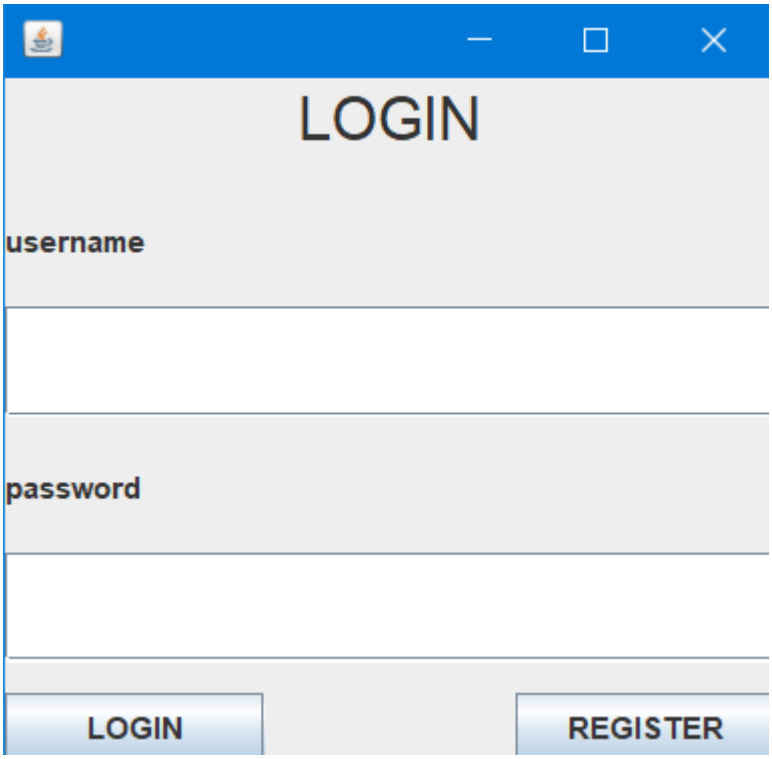
The application follows MVC structure. Each view has its control class where the view is instantiated. There are also classes for Trainers, Trainees (those 2 extend the class Person), for Supplements and Subscriptions. Those are part of the model package.



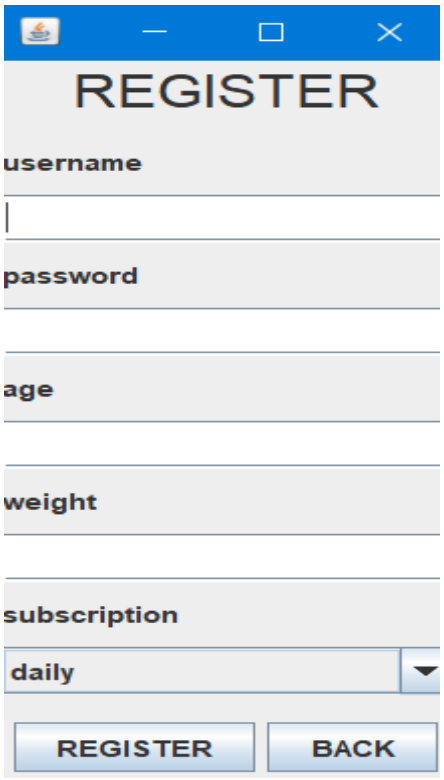
Each frame(view) is instantiated as an object and all 3 views are "glued" by the control classes which implement the ActionListener Interface. The control classes override the original ActionPerformed method.

```
@Override
public void actionPerformed(ActionEvent e) {
```

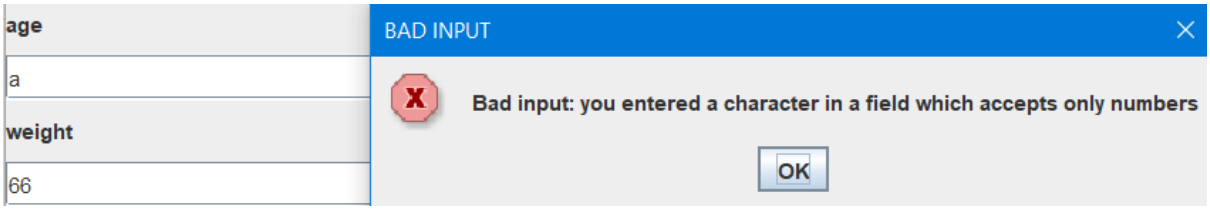
GUI

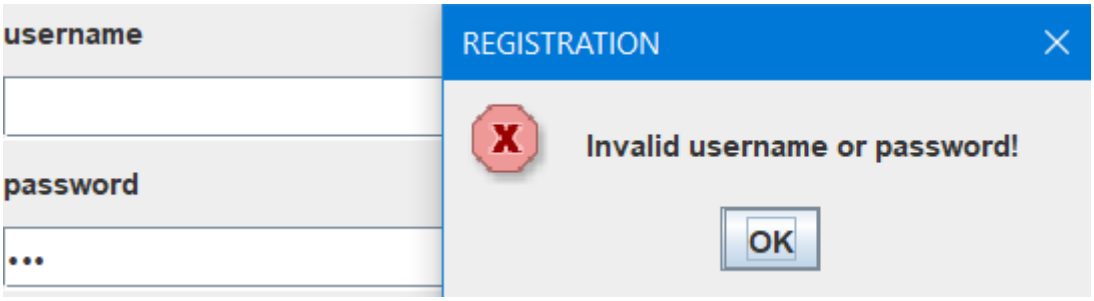


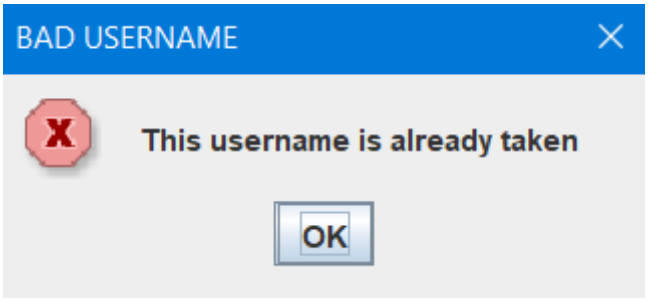
This is the main(first) window. From here the user can log into his/her account with a valid username/password. The process is completed when LOGIN button is pressed. If the user doesn't have an account, a new window will appear, with a registration form.



There are a few error checks like the following:

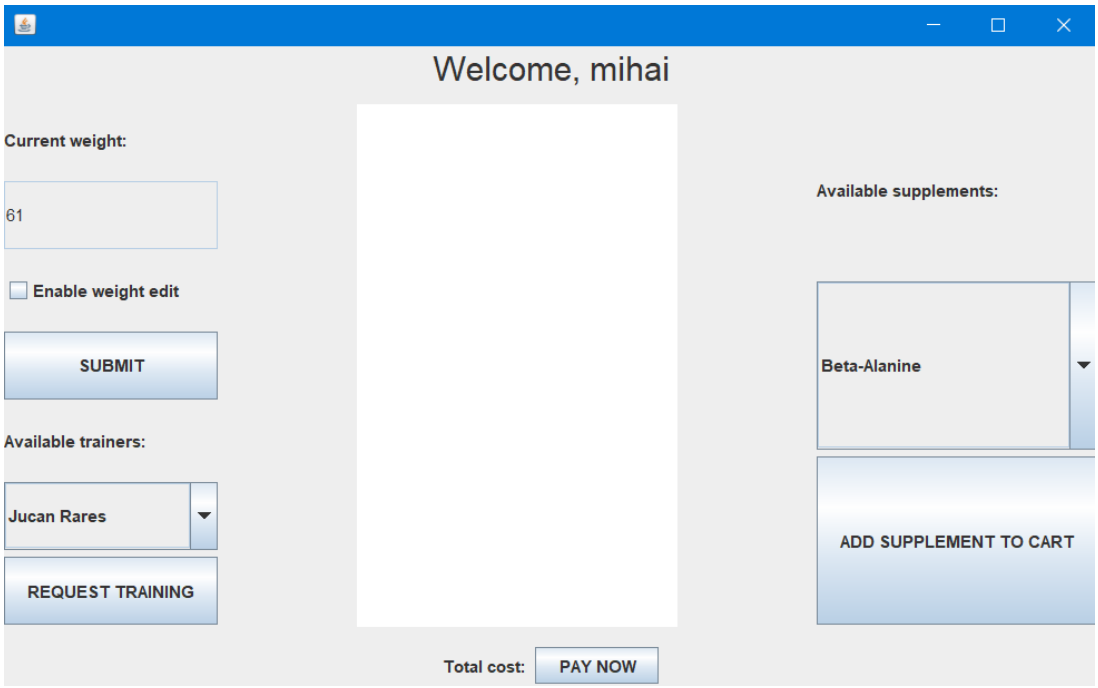






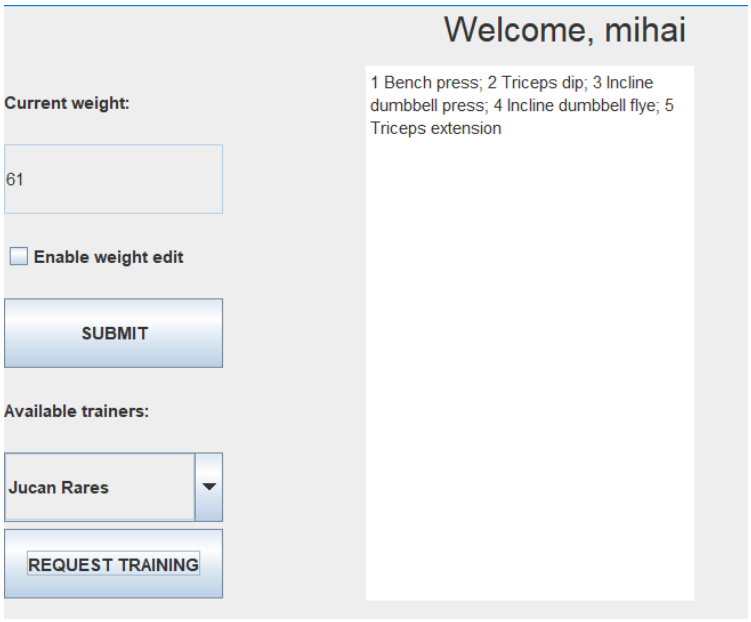
The username &password must not be blank, and weight &age must be valid numbers. In case of bad input, an appropriate error message is shown.

After a successful login, the user is now in his account where he/she can see useful information:

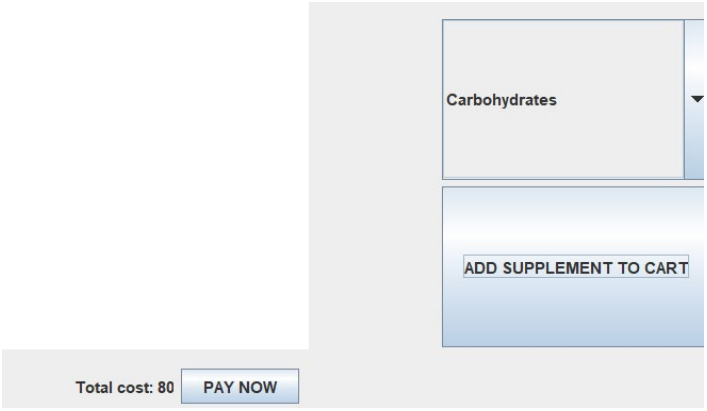


If the user presses the enable weight checkbox, he will be able to edit his/her weight but the value will be updated only after pressing submit.

If the user presses REQUEST TRAINING, the routine of the chosen trainer is shown in the middle.



On the right, the user can select a supplement, and add it to the cart. The final price will be displayed on the bottom. The user has the option to pay for them.



Conclusions

This project was a fun challenge, giving me the opportunity to develop my java knowledge. For future updates, the app may include:

- Automatic due date completion so the user knows when his/her current subscription expires.
- Better GUI aspect
- Option to remove items from the cart
- Ability to log in as trainer and add specific routines based on weight/age