

Criterion B - Design

Flow Chart:

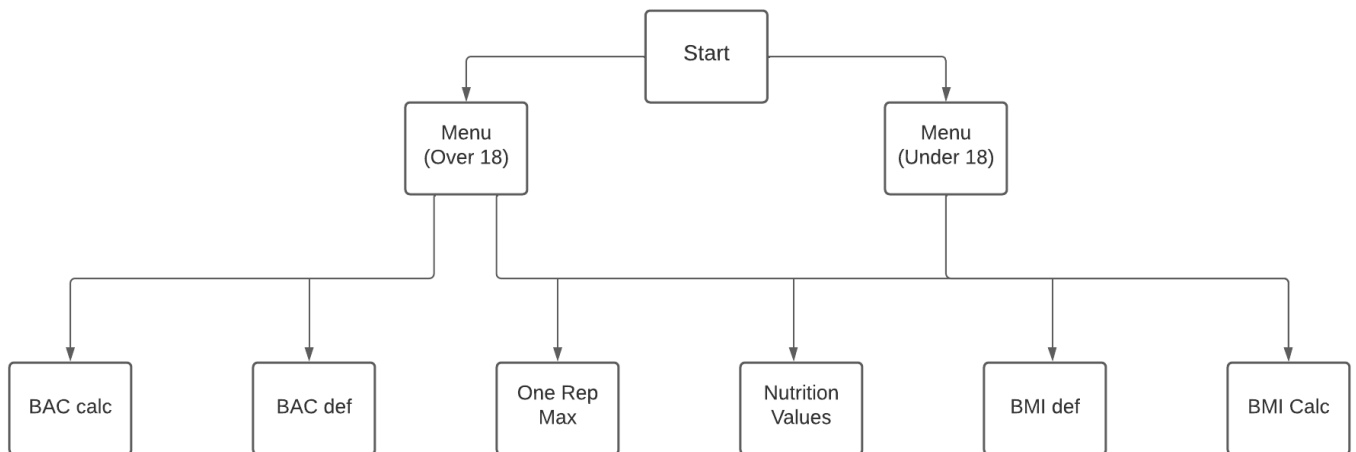


Figure 1. Structure chart representing the way my application works.

Thanks to the structure chart above I was able to figure out what classes my app should have. Those are:

Start - starting window in the app, allowing the age choice.

Menu, MenuUA - two menus, one for those over 18 years old and one for those below that age. They are going to differ, as **MenuUA** will not have BAC(Blood Alcohol Concentration).

BAC Calc - BAC Calculator shows the result in mg/100ml and in per mile (‰)

BAC def - Window showing BAC definition

One Rep Max - Calculator for user's One Rep max, also showing reps and weight ranges for improving hypertrophy, strength, and endurance

Nutrition Values - calculator for caloric value as well as some macronutrients. It is also going to include how long a user has to do certain activities in order to burn said calories.

BMI def - window with BMI(Body mass index) definition

BMI Calc - BMI calculator, showing BMI result, and BMI category with words as well as on chart.

Initial design and Backend planning :

After the first conversation with my client, I decided to draw out an initial design of what the app is going to look like. Thus, said drawings are shown below.

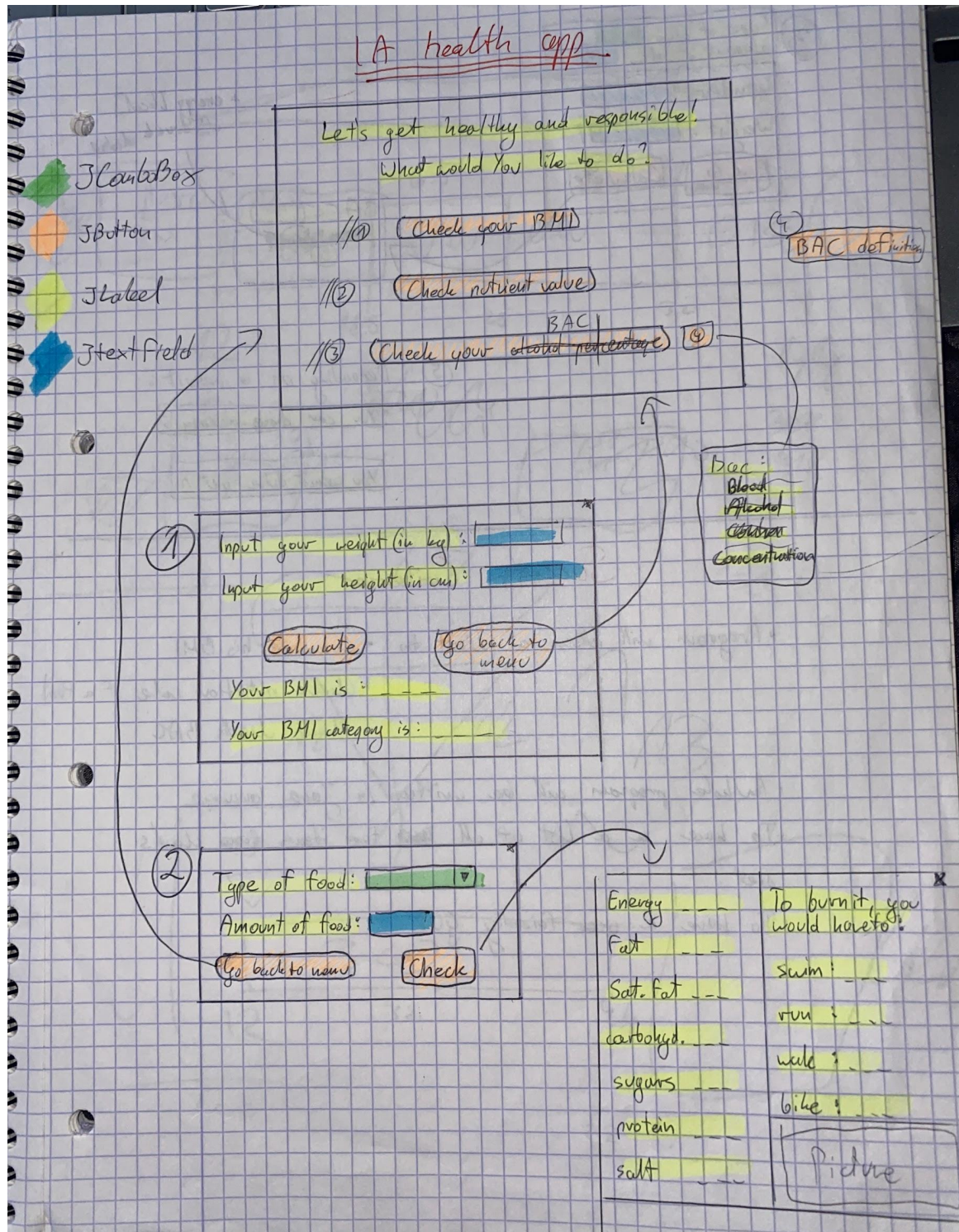


Figure 2. Initial design of the menu, BMI calculator, and Nutrition value calculator

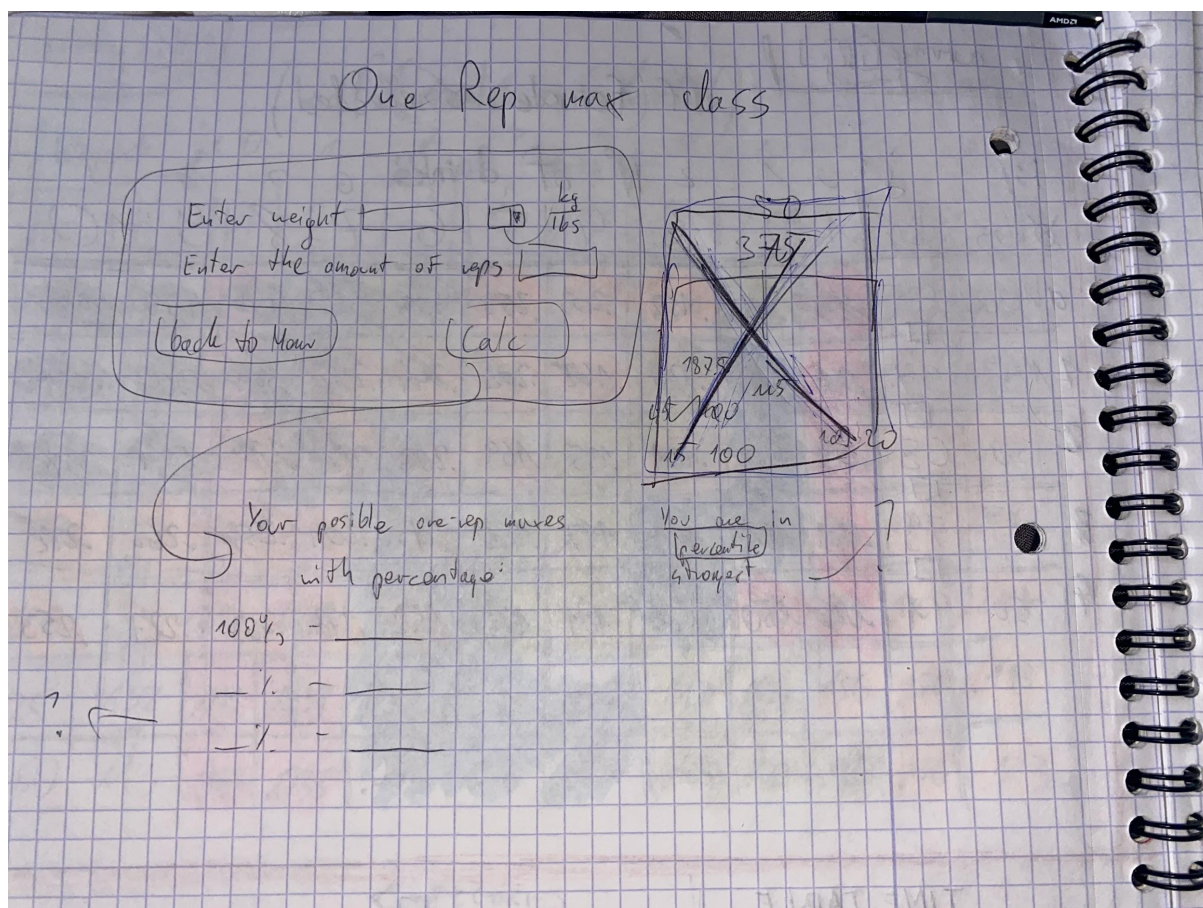


Figure 3. Initial design of the one-rep max calculator.

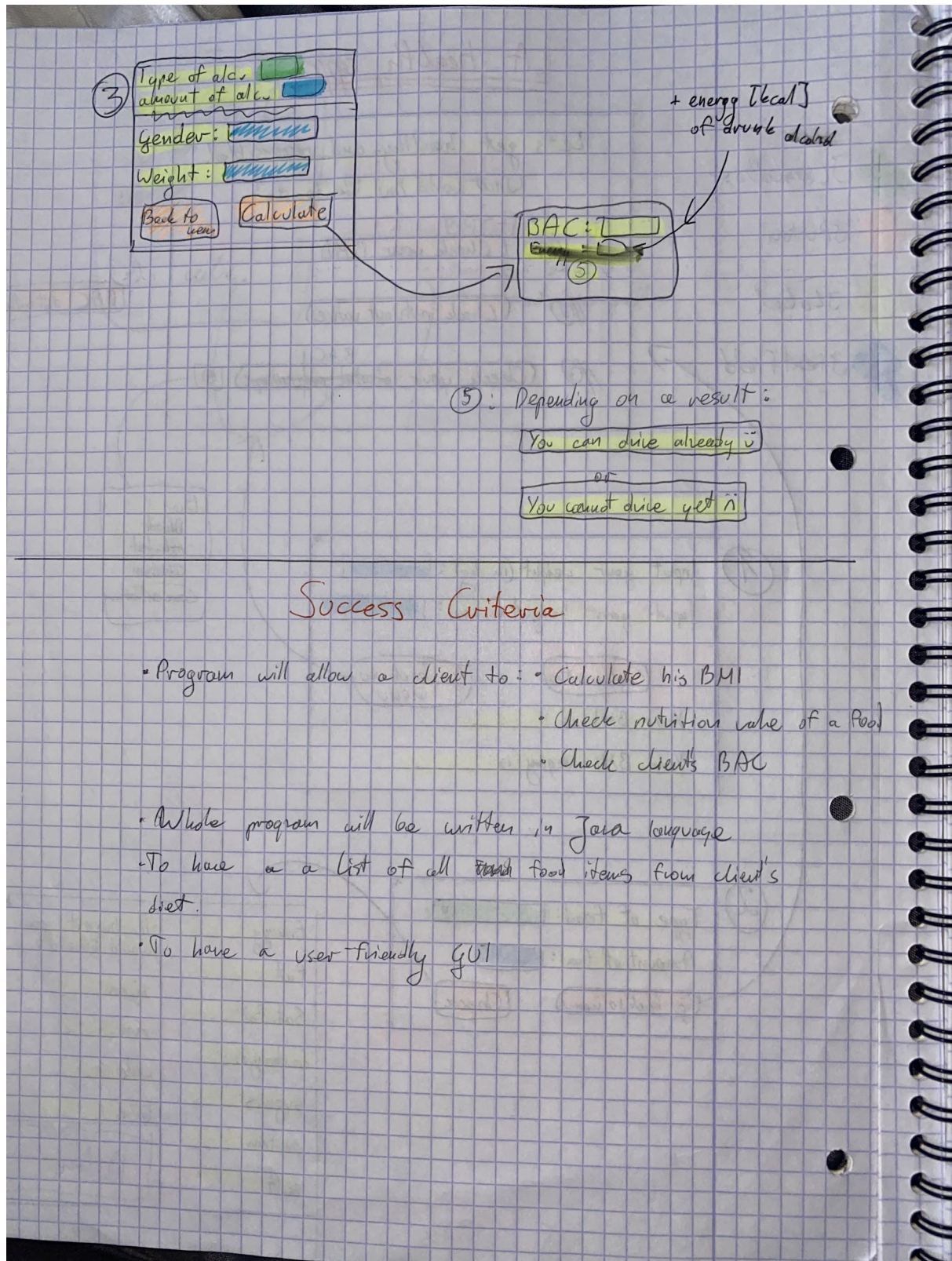


Figure 4. Initial design of the BAC calculator, and initial Success Criteria.

However, for the BAC calculator to work properly, I had to properly plan my work and write out the two two-dimensional matrices that analyze the result and assign the result to certain groups. I did that so that there wouldn't be any mistakes regarding the calculations and the analysis of the result.



Figure 5. Written backend plans for the BAC calculator

UML Diagrams

I decided on using the UML Diagrams, as they are going to allow to see classes' content as well as their decomposition

As the app has a lot of classes, each of them with numerous variables, I decided on doing multiple UML diagrams, each representing different classes. I am taking such a route in order for diagrams to be more readable.

However, as "Methods" class is used throughout most classes, respectively, it is also going to be represented in most of those diagrams.

UML diagram representing classes: **BMI**, **BMIdef**, **BAC**, **BACdef**, **Methods**, and **RoundBtn**.

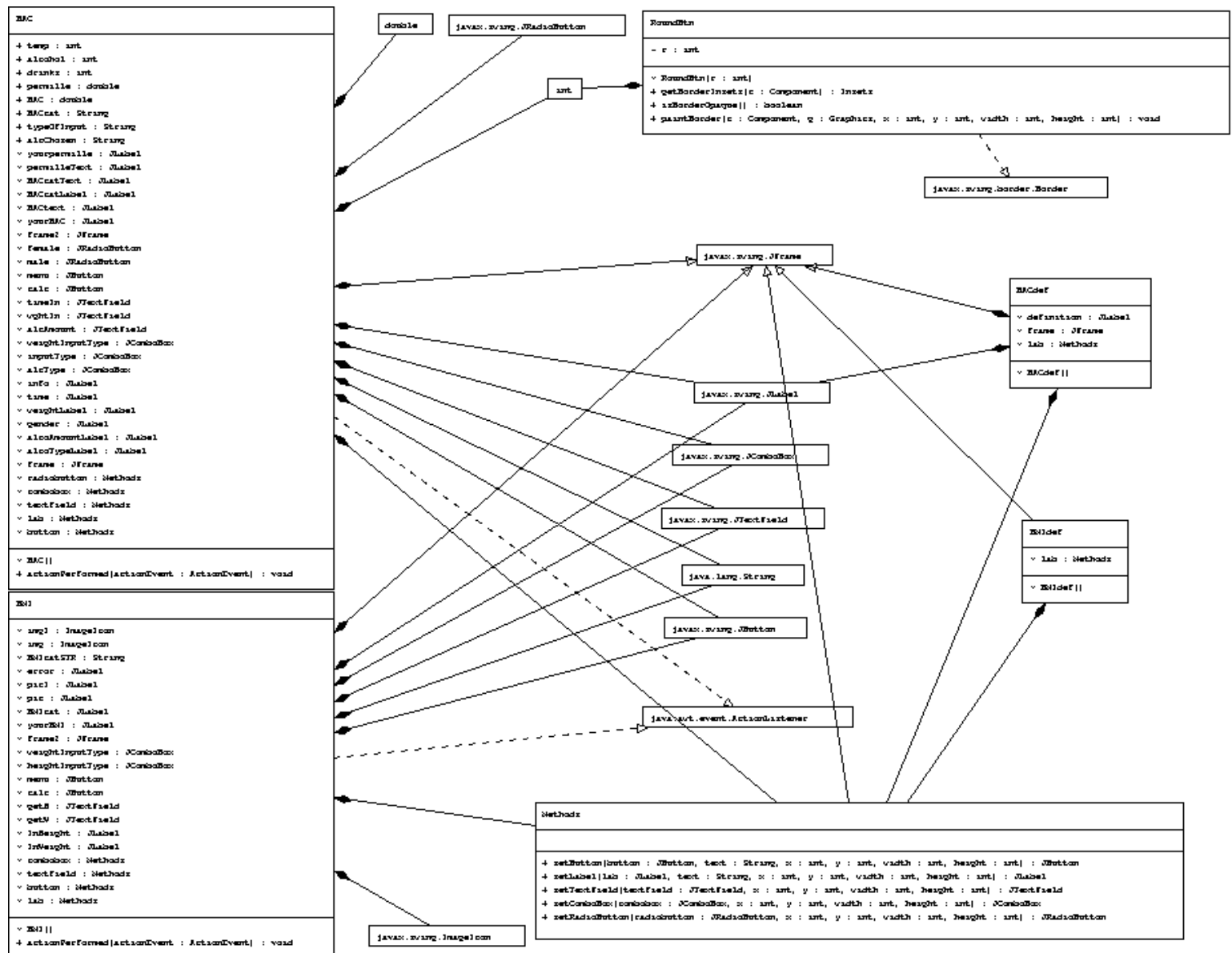


Figure 6. UML Diagram representing classes: BMI, BMIdef, BAC, BACdef, Methods, and RoundBtn.

UML Diagram representing classes: **Main**, **Menu**, **MenuUA**, **Start**, and **Methods**.

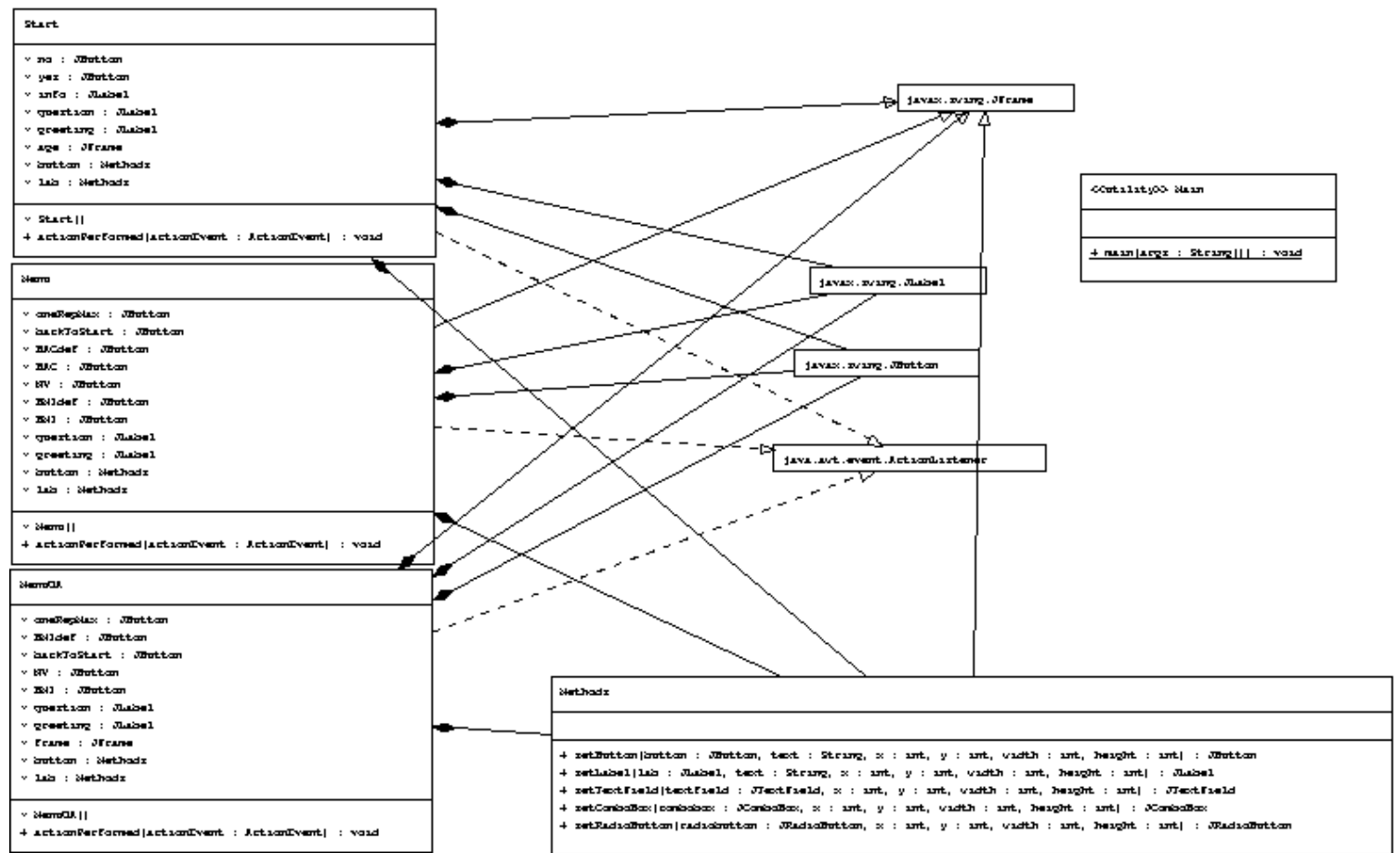


Figure 7. UML Diagram representing classes: **Main**, **Menu**, **MenuUA**, **Start**, and **Methods**.

UML Diagram showing classes: **NV**, **oneRepMax**, and once again **Methods**.

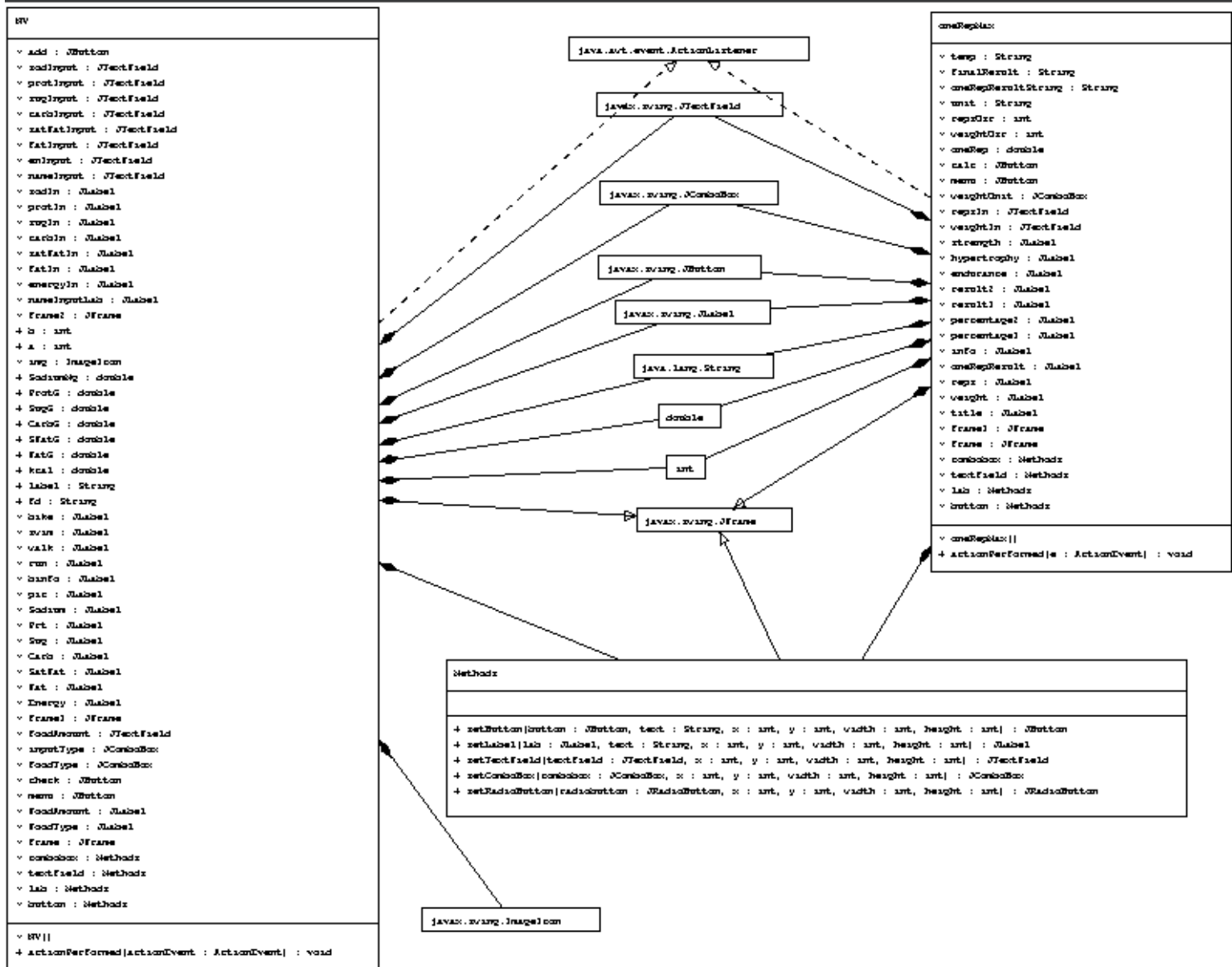


Figure 8. UML Diagram representing classes: **NV**, **oneRepMax**, and once again **Methods**.

Prototype:

The app was made based on the aforementioned initial design. However, there are some changes, as during the process of development I decided on adding / removing / changing certain aspects, so that the app would be much more pleasurable to use.

First window of the program.

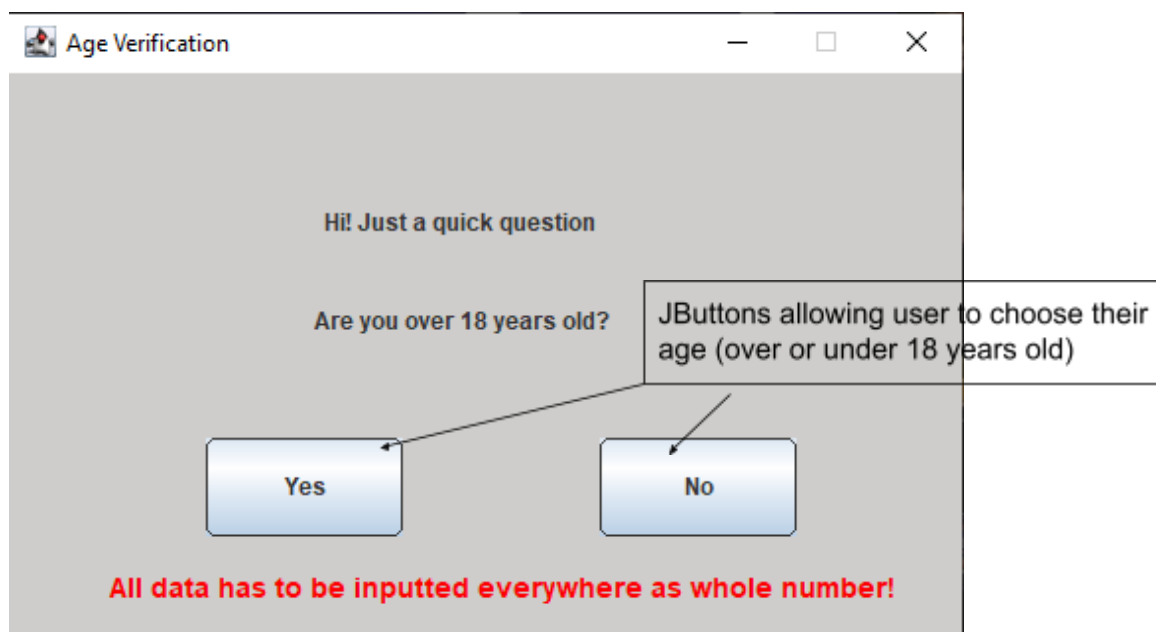


Figure 9. Visualization of the first window of the program.

After choosing their age, the user is redirected to the second window of the program - the menu. As there are two menus (for those over and under 18 years old), the window depends on the option chosen in the first window of the program. Both menus can be seen below.

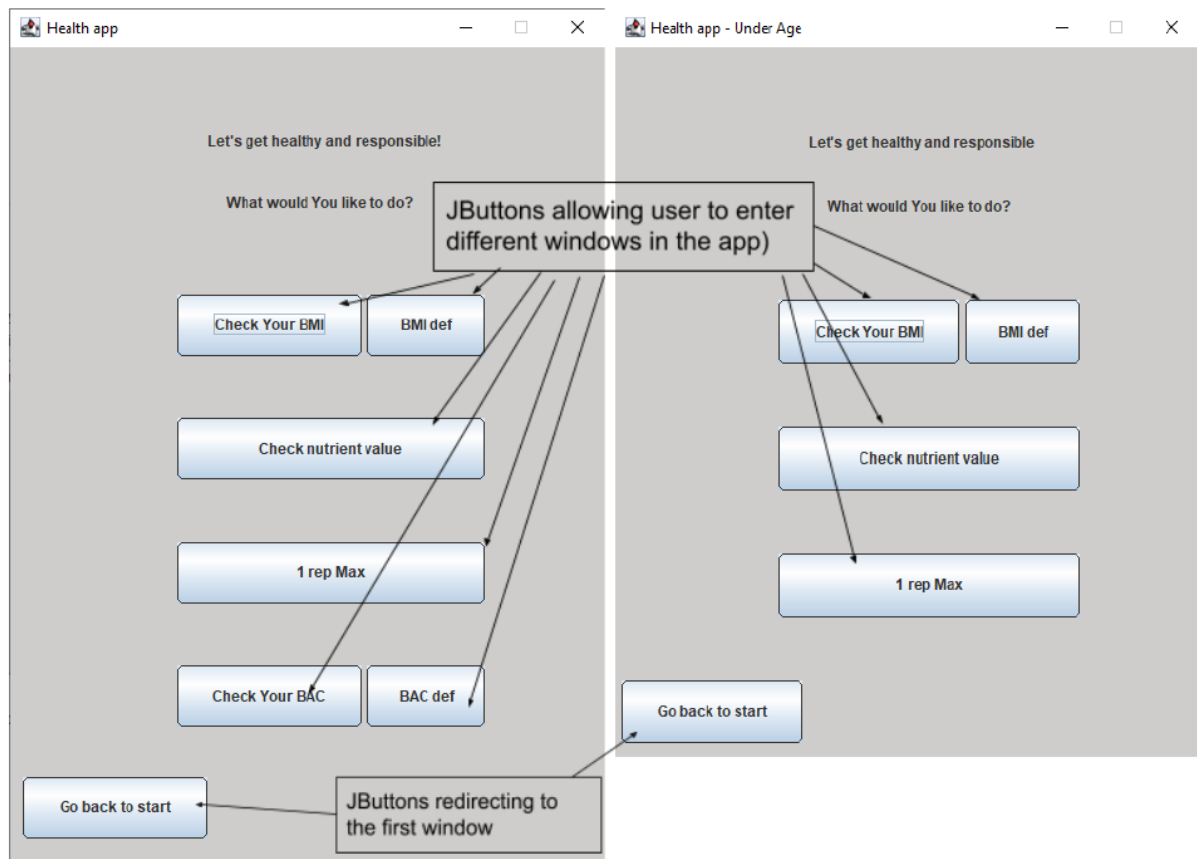


Figure 10. Visualization of both normal and underage menus (respectively)

After choosing the first option from both menus, users are redirected, as the text on the button suggests, to the BMI calculator.

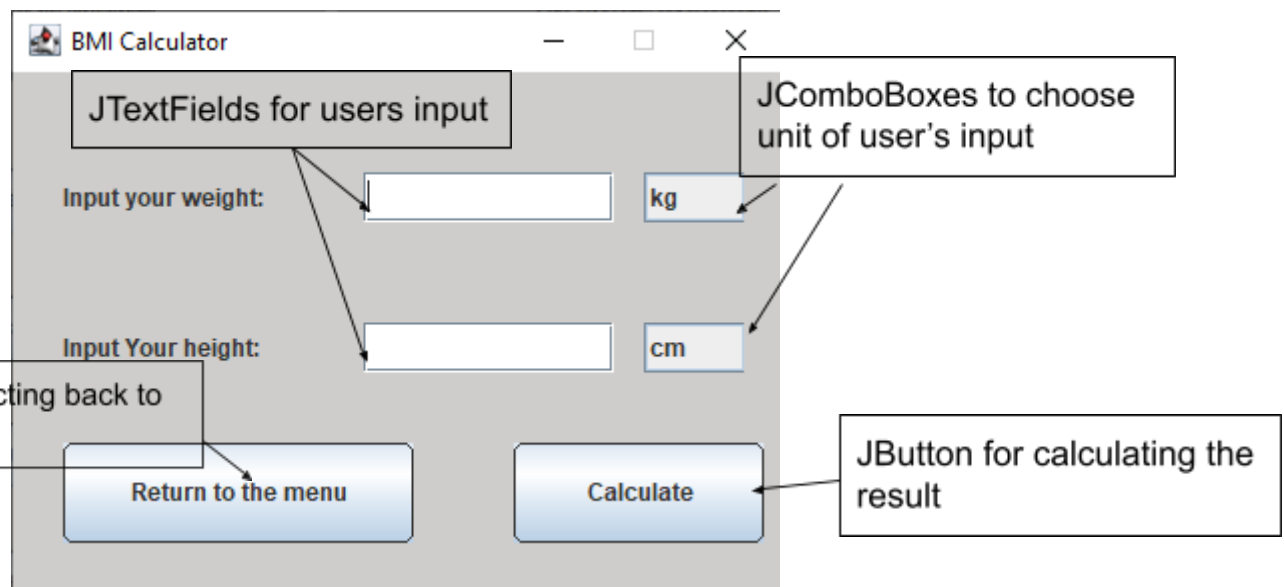


Figure 11. Visualization of the BMI calculator window.

After inputting data, the following window with results appears.



Figure 12. Visualization of the window with BMI calculation results.

After choosing the second option from both menus, the window with BMI definition and the formula for calculating it appears. This window consists of only **JLabels**.

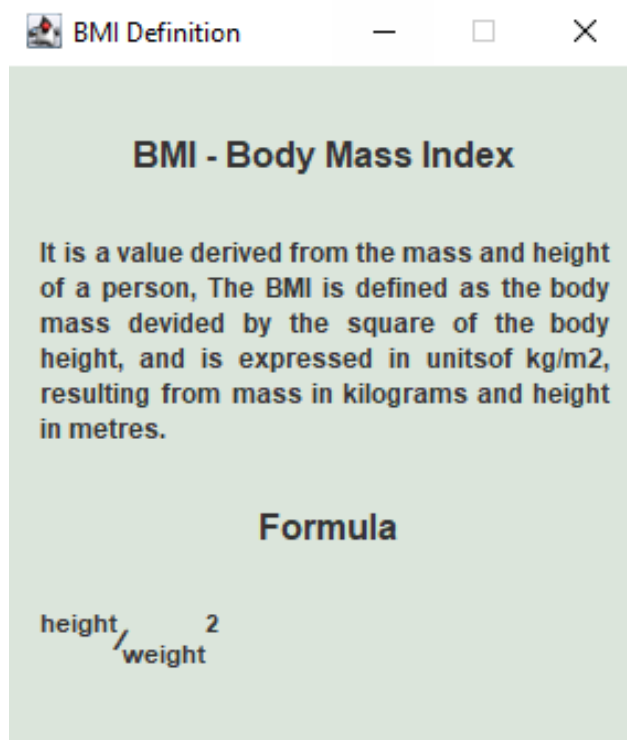


Figure 13. Visualization of the window showing BMI definition.

After choosing the third button in both menus, the user is redirected to the nutrition values calculator.

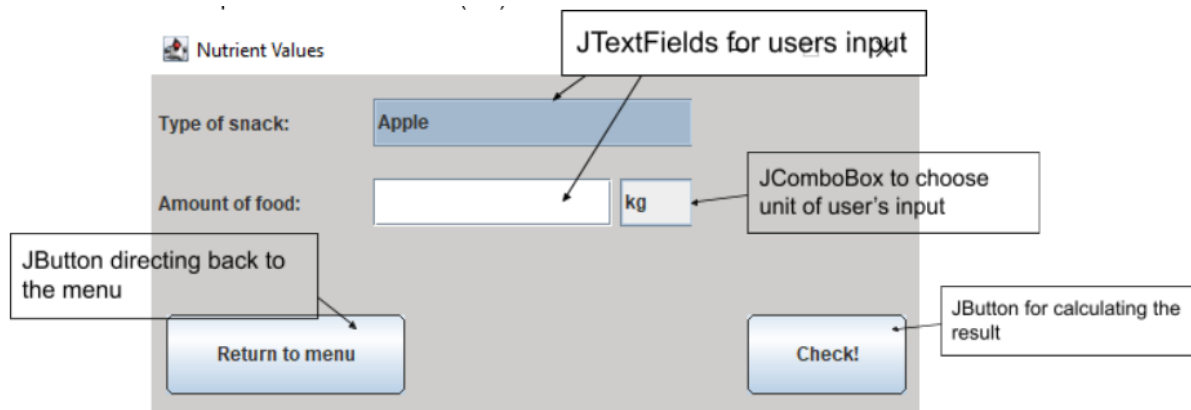


Figure 14. Visualization of the Nutrition Values calculator window.

After inputting necessary data, the following window appears. This window also contains only from **JLabels**, as there is no user's input and only the result.

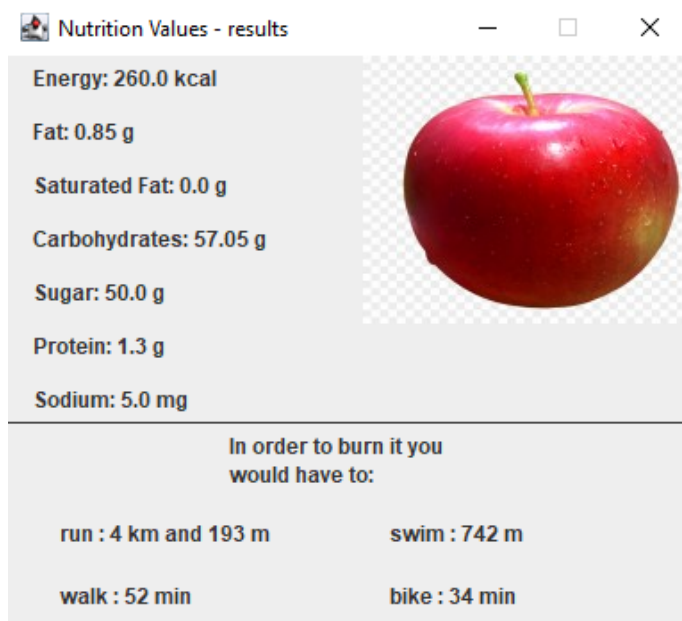


Figure 15. Visualization of the window showing results from Nutritional Values calculator.

After choosing the fourth option in both menus, the user is redirected to the one rep max calculator.

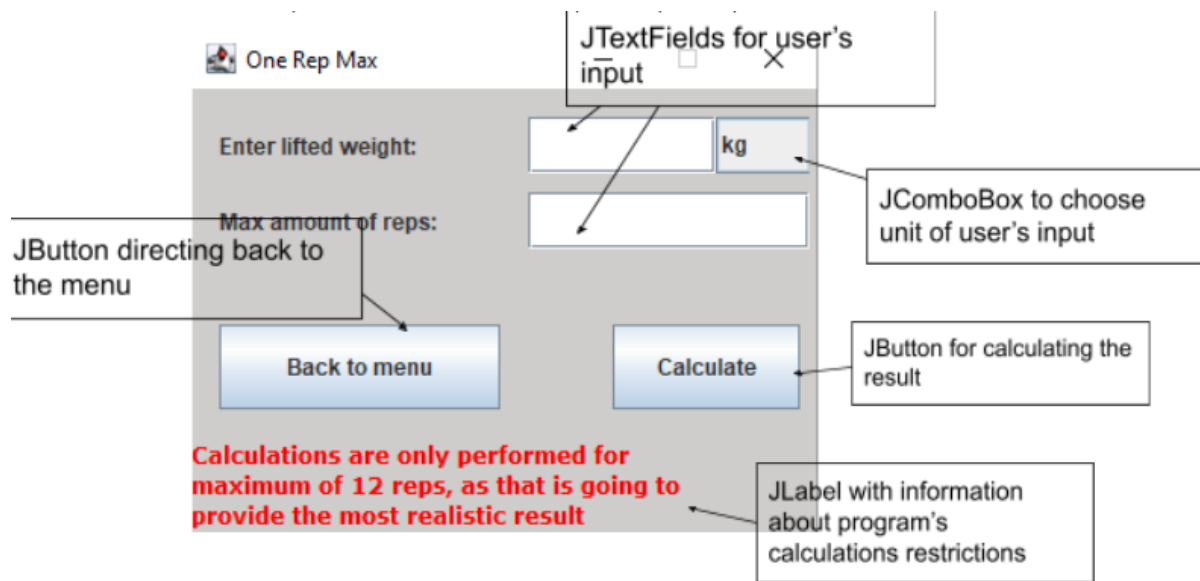


Figure 16. Visualization of the one rep max calculator window.

After inputting all necessary data, the following window appears. It consists only of **JLabels**, as there is no user's input and only the result.

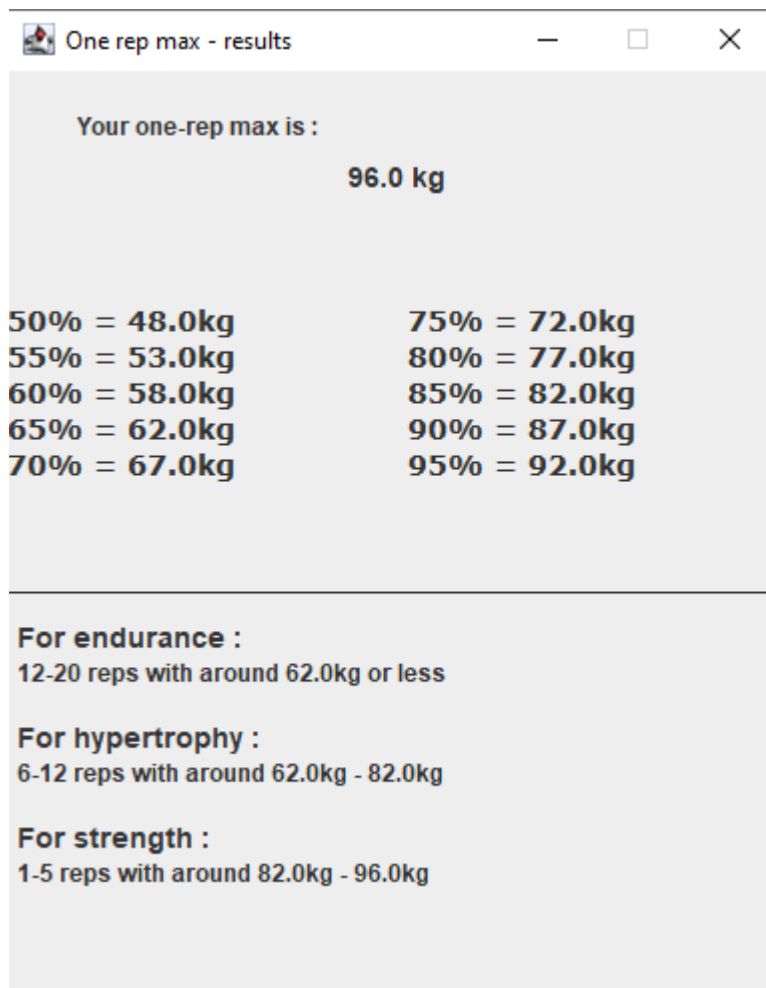


Figure 17. Visualization of the results window in the one rep max calculator.

After choosing the fifth option which is present only in the menu for those over 18 years old, the user is redirected to the window containing BAC Calculator.

The image shows a Java Swing window titled "Blood Alcohol Concentration". The window contains the following elements and annotations:

- JLabels:** A box on the left with arrows pointing to the labels "Type of alcohol", "Alcohol amount:", "What is Your gender?", "How much do you weight?", and "Hours since 1st drink?".
- Type of alcohol:** A JComboBox showing "Vodka".
- Alcohol amount:** A label followed by a JTextField and a JComboBox showing "shot 40ml/1.35oz".
- What is Your gender?:** Two radio buttons labeled "Male" and "Female".
- How much do you weight?:** A JTextField followed by a JComboBox showing "kg".
- Hours since 1st drink?:** A JTextField.
- JComboBoxes to choose type and unit (serving) of alcohol:** A box with arrows pointing to the "Vodka" and "shot 40ml/1.35oz" JComboBoxes.
- JRadioButtons for choosing user's sex:** A box with arrows pointing to the "Male" and "Female" radio buttons.
- All fields must be filled:** A red text label in the center of the window.
- Buttons:** Two buttons at the bottom: "Return to menu" and "Calculate".

Figure 18. Visualization of the BAC calculator window.

After inputting all necessary data, the following window appears with the results. This window has only **JLabels**, as there is no user's input and only the result.

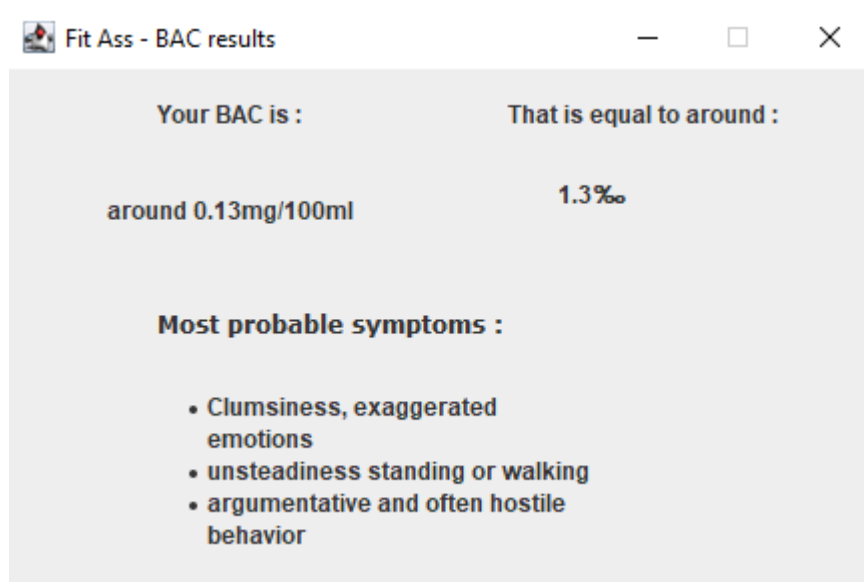


Figure 19. Visualization of window containing results from BAC calculator.

After choosing the sixth option, that is also only in the menu for those over 18 years old, the user is redirected to the BAC Definition. That window, similarly to BMI Definition, consists of only **JLabels**.

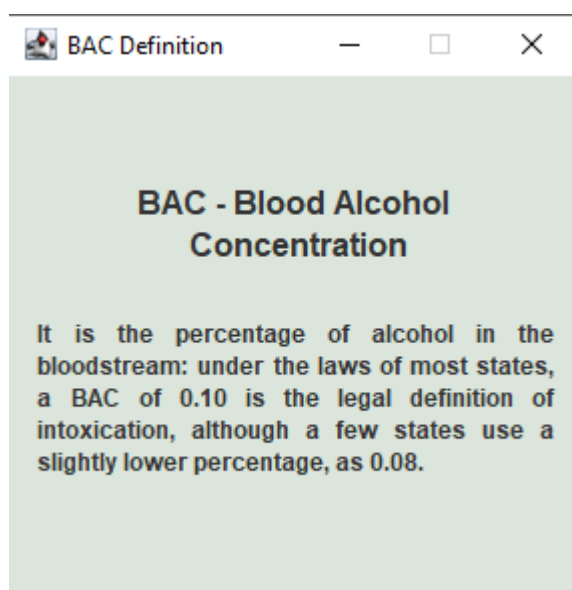


Figure 20. Visualization of the window containing BAC definition.