Budapest University of Technology and Economics

WEB-based Laboratory

Written report

Beerculator

Authors:
Alvaro Gonzalez,
Maud Leray,
Patricio Sanchez,
Tomás Sekanina

Supervisor:
Dr. János Hamar

Contents

1 Introduction

As students, we often hang out with our friends in pubs or in nightclubs. In those places, we usually have a few drinks, most of them containing alcohol. We are aware that this can be risky, and we also know that we should not exceed a certain amount, especially if we need to drive after the party. But the problem is that most of us do not know their limits, and do not realize how long it takes for their body to remove the alcohol.

To create a more alcohol-conscious society, there should be tools to inform the general crowd about the quantity of alcohol a person can ingest and the effects it will have on him or her. This is why we created Beerculator, an interactive tool which calculates the blood alcohol concentration based on the drinks that a person has had in a certain amount of time. It will also indicate how long the person has to wait before being sober again.

This is very important to provide such tools because it is proven that alcohol can have really bad effects on our behaviour. One of the most important of them is the increasing probability to cause accidents. The trend on FIGURE 1 should help you realize the consequences.

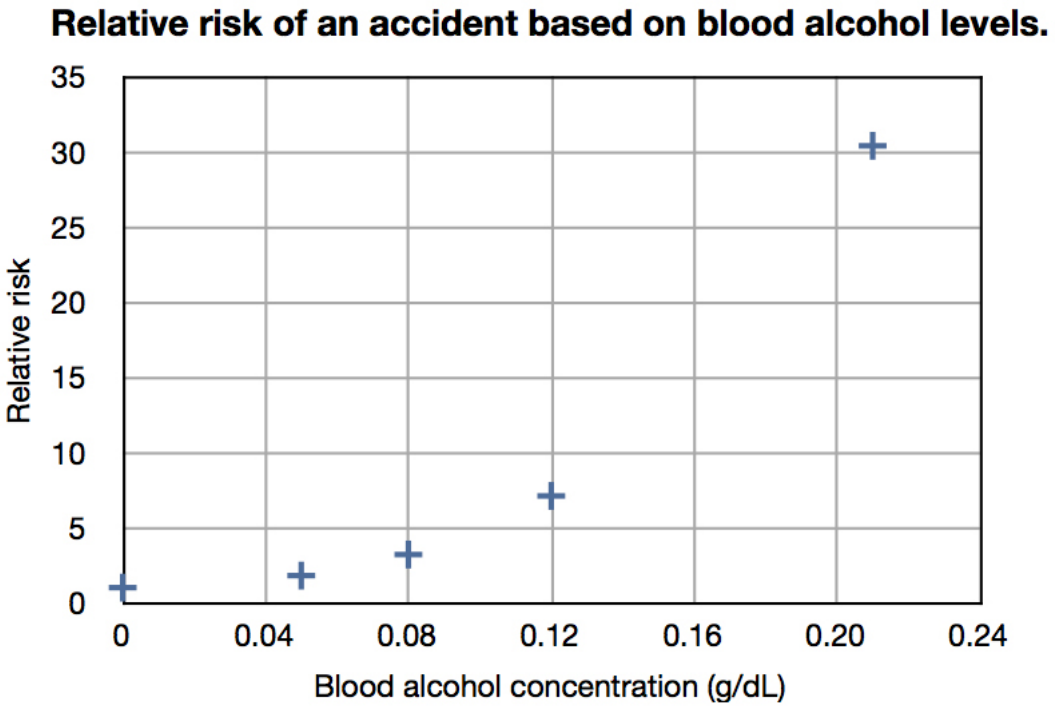

Figure 1: Relative risks of accidents based on the alcohol level

2 Requirement analysis

Bob (made-up client for the project) consumes alcohol and can feel its effects in his bloodstream, but does not know when to stop or if he is on a safe-to-drink amount of alcohol. Bob requires a simple interactive tool that he can use on the go to check his alcohol level without being too specific about the details.

Bob should not be the only one to access this tool, so it should be available on the Web. He does not need the tool to store informations about previous alcohol level calculations, he just wants it to dynamically calculate his current alcohol level. He also wants to know how long it is going to take for his body to remove all this alcohol, for example because he absolutely needs to drive tomorrow morning to go to work. So when he uses Beerculator, Bob would like to be able to perform all the actions described in FIGURE 2.

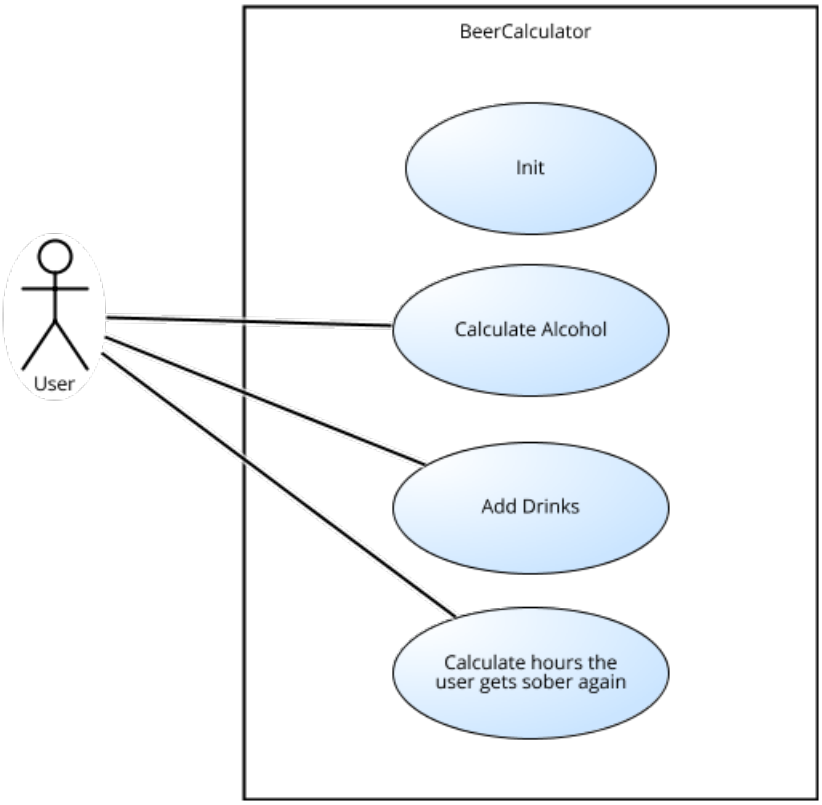

Figure 2: User case diagram for Beerculator

In order to perform the calculations, Beerculator needs to know a few informations about Bob: his gender, his weight, the amount of alcohol he has consumed, and also the time when the drinking started. This will be detailed in SECTION 3.

3 Specification

3.1 Needed informations

In order to calculate the user's blood alcohol concentration (BAC), we need a few informations about him and about his drinking. More precisely, we need the elements listed below:

- Gender (Male/Female);
- Weight;
- Number of hours since the drinking began;
- Amount of alcohol.

3.2 BAC calculation formula

To calculate the blood alcohol concentration, we will use the Widmark formula, which is the following:

$$BAC = A \div (R \times M) - 0.015 \times H \tag{1}$$

In this formula, we have:

- BAC : Blood Alcohol Concentration;
- A : Alcohol ingested, in grams;
- R : Ratio (0.70 for men, 0.55 for women);
- M : Body weight, in kilograms;
- H : Number of hours since the drinking began.

To obtain the value of A, we use this formula:

$$A = (V \times P \times 0.8) \div 100 \tag{2}$$

In this formula, we have:

- V : Volume of alcohol in milliliters;
- P : Alcohol degree, in percentage.

3.3 Elimination of alcohol

The median rate of decrease in BAC is considered to be 15 milligrams per cent (mg%) per hour. In our calculations, we used the approximations listed in TABLE 1.

BAC value	**Hours until sober**
0.016	1
0.05	3.75
0.08	5
0.10	6.25
0.16	10
0.20	12.5
0.24	15

Table 1: Hours until sober depending on the BAC value

4 Solvable tasks and team management

After a few meetings to determine the requirements and the objectives of the project, the implementation of Beerculator has been divided into several groups of solvable tasks, as detailed in TABLE 2.

Group	**Tasks**
Diagrams	UML for Java UML for databases User case Activity diagram
Databases	Create tables Set up methods Define relations Write script
Server	Define Java server Create Java beans
Calculation	Find formulas Implement them
Design	User interface

Table 2: Solvable tasks for the implementation of Beerculator

Some other tasks have been completed in parallel for the project, but are not referenced in TABLE 2 because they are not directly linked to the development of Beerculator. For example, writing the final report was also part of this web-based project, and every member of the team contributed to it.

We often worked all together and we had very regular meetings, so everyone was always aware of the state of the project and of the remaining work. And of course if anyone had a problem on its task, we helped each other so we all contributed to the whole project. But in general, each group of tasks in TABLE 2 had a « manager » and the distribution was the following:

- **Alvaro**: Diagrams;
- **Maud**: Calculation;
- **Patricio**: Server;
- **Tomás**: Databases.

The last group, Design, is not in the previous list because it had no real responsible, everybody in the team equally contributed to it.

5 Selected technologies and alternatives

5.1 Selected technologies

The two main technologies used in this project are Java and XHTML. Java was used to code the main calculations of the program, handle the beans for variables and overall functionality, and to link the database and server with the index. XHTML was used to define the UI of shown web page. The UI was created via XHTML in conjunction with CSS3, JavaScript, and Boostrap library. Also the XHTML called bean functions to display the calculations made in the Java classes. This can be seen in the tables which are generated by the calling of:

PostgreSQL and its JDBC (Java Database Connectivity) driver were used to create, store and manage the various databases of the project. This has been recommanded at the beginning of the semester.

We also needed Glassfish4 to create and run the application server, and to synchronize it with the project.

We chose to work with the JSF (JavaServer Faces) framework, associated to Facelets, to develop the Web components.

5.2 Alternatives

The selected technologies introduced in SUBSECTION 5.1 are not the only ones that we could have used. This subsection is here to present a few alternatives.

There are many options to handle databases, for example we could have used MySQL instead of PostgreSQL. To build the user interface, it would have been possible to use PHP with CSS and HTML. Finally, we could also have used JSP (JavaServer Pages), and not JSF, to implement the web page.

6 Design

In this section we will first present the internal structure of the application (in SUBSECTION 6.1) and then we will introduce Beerculator's user interface (in SUBSECTION 6.2).

6.1 Program structure

Beerculator's implementation is organized in three main elements that are described in the next paragraphs.

Databases
There are three tables in the project, which are the following ones:

- drinks: contains all the predefined drinks that the user can select;
- drink_records: contains the drinks selected by the current user;
- users: contains all informations about the different users.

We also needed queries to fill the « drinks » database, they are all written in a SQL script called list_drinks.sql.

Java
The project consists in three main classes, Drink.java, DrinkRecord.java and User.java. Their respective UML representations are given by FIGURES 3, 4 and 5.

The calculation methods, calculateBAC() and hoursUntilSober(double), are implemented in User.java. They are based on the formulas previously introduced in SECTION 3.

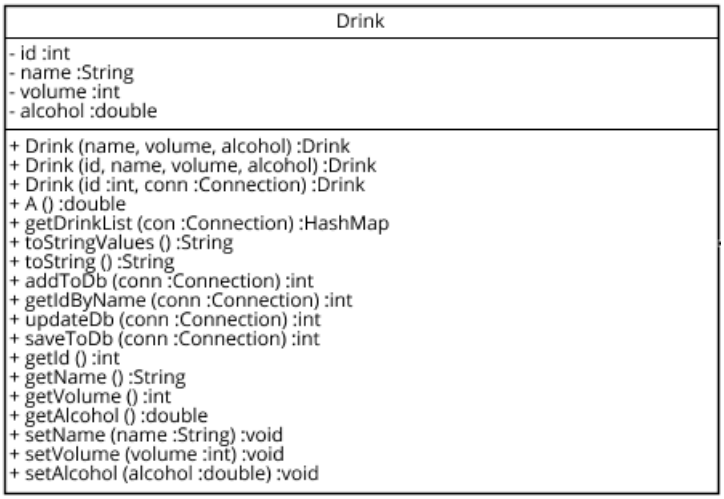

Figure 3: Class Drink.java, its attributes and its methods

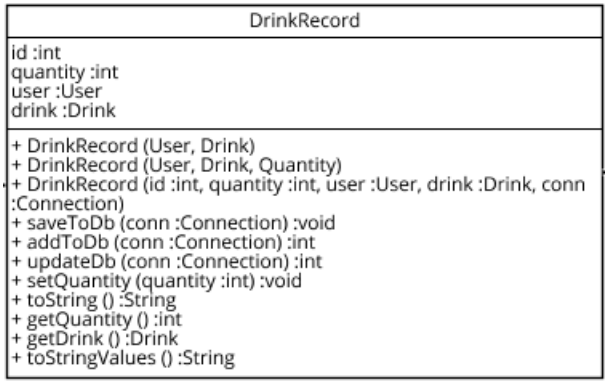

Figure 4: Class DrinkRecord.java, its attributes and its methods

Figure 5: Class User.java, its attributes and its methods

When the user starts Beerculator, the tool sends queries to the database in order to get the predefined drinks. When this is done, the user can interact with the application. He can add drinks, remove some, and enter his personal data (gender and weight). After that, the user can launch the calculation and Beerculator can send him the results. This behaviour is described in an activity diagram, on FIGURE 6.

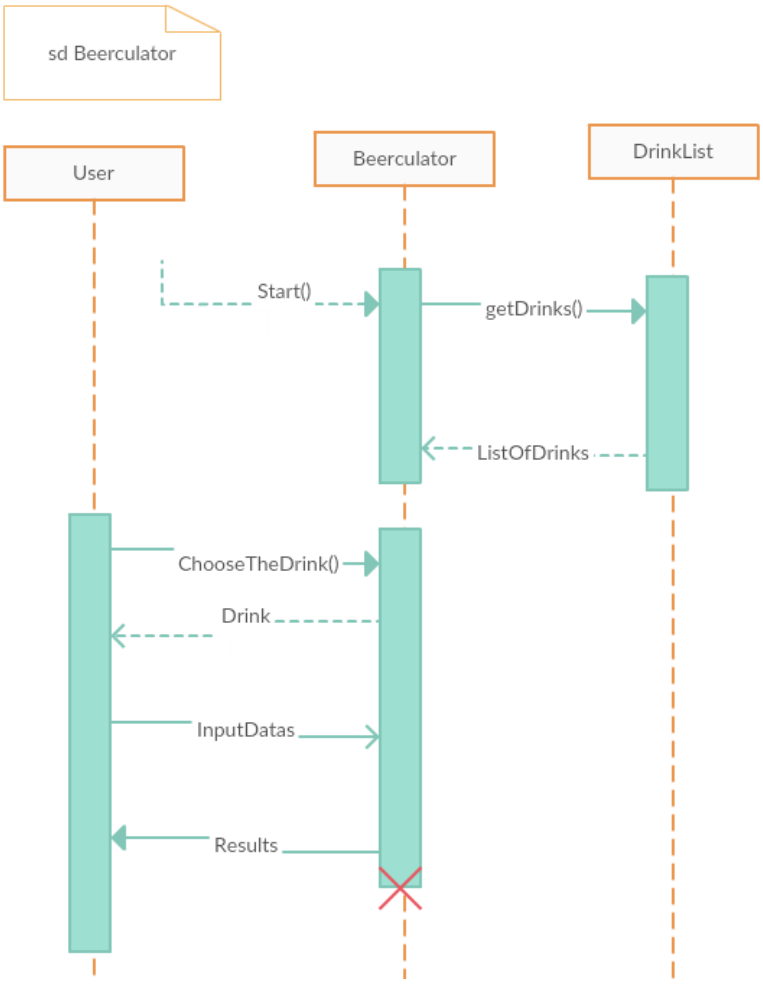

Figure 6: Activity diagram for the use of Beerculator

Web content
The web content is located in the file Index.xhtml. It is used to generate different URLs based on the session ID. This means that instead of creating a new page for each user, it is just redirected to the page with the right session ID.

The web content also includes a folder called WEB-INF, that contains two files: web.xml and faces-config.xml. They are used to manage the server and to handle the Java beans.

6.2 User interface

Three main elements were needed in Beerculator's interface: one to get the user information (weight, gender, etc), one to display the list of drinks and one to show the results. The user interface can be seen in integrality on FIGURE 7. It will also be described tab by tab in SECTION 7.

Figure 7: User interface of Beerculator

On the top left part of the interface, the first information is the session ID, that can be used by the user to get back to the page later. Then the user can add all informations that Beerculator needs from him. He can indicate its gender (Male or Female) by picking in a list, and its weight thanks to a text box. Another text box is available for him to tell the time that he started drinking. He can save all those informations, in case he wants to come back later, by clicking on the blue « Save » button.

On the right side of the web page, the user can see a list of predefined drinks (with their volume and their alcohol percentage). With each alcohol there are two squares: one green and on red. The user can click on the green one to add one item of the corresponding alcohol to his list of drinks. If he clicks on the red button, one item of this alcohol is removed from the list of drinks. He can also save his list of drinks, for instance if the party is not over and he intends to add some more later, by clicking on the blue « Save » button.

When he is done, the user can click on the « Calculate » green button so that Beerculator starts the computation. The results appear on the bottom left corner, in the « Calculation » tab. First the BAC value is indicated, next to label « Alcohol level » , and below it the number of hours until the user gets sober again (next to label « Sober after »).

Finally, if the user wants to erase everything and start over, he can press the « Reset » red button on the bottom of the page.

7 Testing instructions

When you get to Beerculator's web page, the first thing to do is to enter your personal informations. To do so, in the « User data » tab (FIGURE 8), in the first text box you have to indicate your weight in kilogramms. Then, you have to pick your gender in the list: male or female. Once you are done here, you should click on the « Save » button.

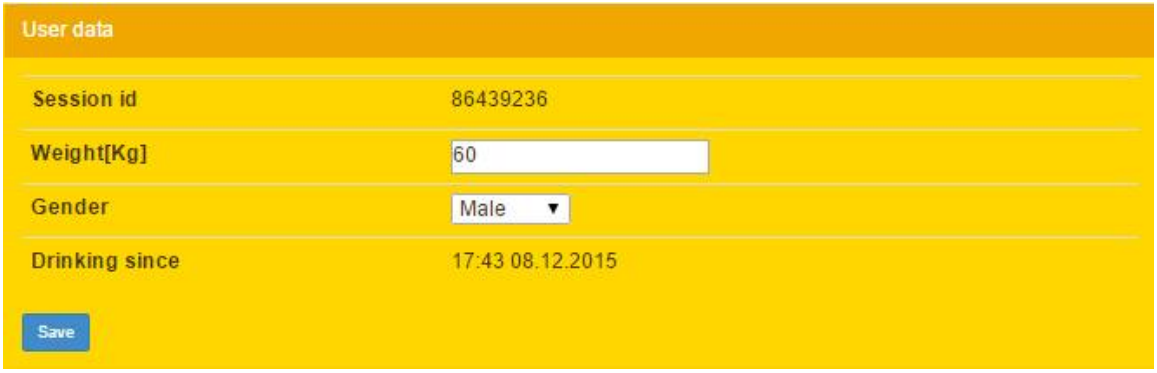

Figure 8: User data tab in the user interface of Beerculator

After that, you can move on to the right side of the screen and start adding drinks to your list, in the « Drinks » tab (FIGURE 9). To do so, you can click on the green squares corresponding to the drinks you have had. If you made a mistake, you can click on the associated red square in order to cancel and remove this drink. If you update your drinks this way, they are automatically saved. But you can also edit the quantity of each drink directly in the corresponding text box, which allows you to add or remove several drinks at the same time. If you do so, you should click on the « Save » button. Also, if you want to cancel and start all over again, you can press the red « Reset » button on the bottom of the page.

Figure 9: Drinks tab in the user interface of Beerculator

Then, you can launch the computation in the « Calculation » tab (FIGURE 10) by clicking on the « Calculate » button. The BAC value is indicated in the first text block, and it is followed by the number of hours before you get sober again.

Figure 10: Calculation tab in the user interface of Beerculator

8 Conclusion

Night clubs, parties, and actually any drinking with friends are good ways to have fun. However, people should drink responsibly and know their limits in the amounts of alcohol they can ingest. Beerculator could be developed further to provide a tool for people that need a simple and effective solution, for a problem that most of them are not aware of having: reckless drinking. It is important to mention that the project is not over and that it is only on its first stage, however the results have nicely surprised us because we did not expect a tool like this to be as effective as it is with such a simplicity of usage.

Definitely, some problems happened in the process of making this project, but through teamwork and dedication this hurdles could be surpassed. Everyone on the team had a certain goal and each one of us helped to reach that goal, we collaborated throughout the weeks that we had to make sure that no important details were missed and that deadlines would be respected. The final version provides an effective solution to the problem we decided to solve, and complies with our goals. There is room for improvement and good ideas for the future of the project, but for the time being we feel satisfied with the final version.

9 Further plans

Beerculator is already a very useful tool, but it could still be extended with some new functionalities.

New drinks Beerculator allows the user to choose what he has consumed among a list of predefined drinks. It would be nice for him to be able to customize the application, and more precisely it would be interesting if he could add some new drinks to the database, for example his favourite ones.

Mobile version Beerculator has been coded for a desktop use, it could be interesting to make its interface responsive so that people can use it on their mobile phones. It could also become an Android application that users can download on their device and then customize.

Artificial Intelligence Artificial Intelligence could be added to the Beerculator for customization of the real limits of the user. Since everyone has different organisms and limits, the application could ask for a real sample of your alcohol level and compare it to the averages in formulas. Then it could tailor a certain average for the user instead of using the predefined ones.