

BMI Calculator Website

The purpose with this project was to, first-and-foremost, learn how to handle website-coding, and, secondly, to be able to produce a website that other people could access. The project was centred on creating a BMI calculator and sign-in / sign-up site that was linked to a database, by following tutorial videos provided by the teachers.

The development of the various features, such as the calculator, the sign-in, sign-up, etc., was based on various user-stories, e.g. “*As an administrator, I want to be able to view all BMI-calculations on a list.*”

Though we weren’t provided with a concrete background of what kind of company would outsource the development of this kind of website, it seems obvious that it would be for a company centred on health & wellness and/or tackling weight aberrancy.

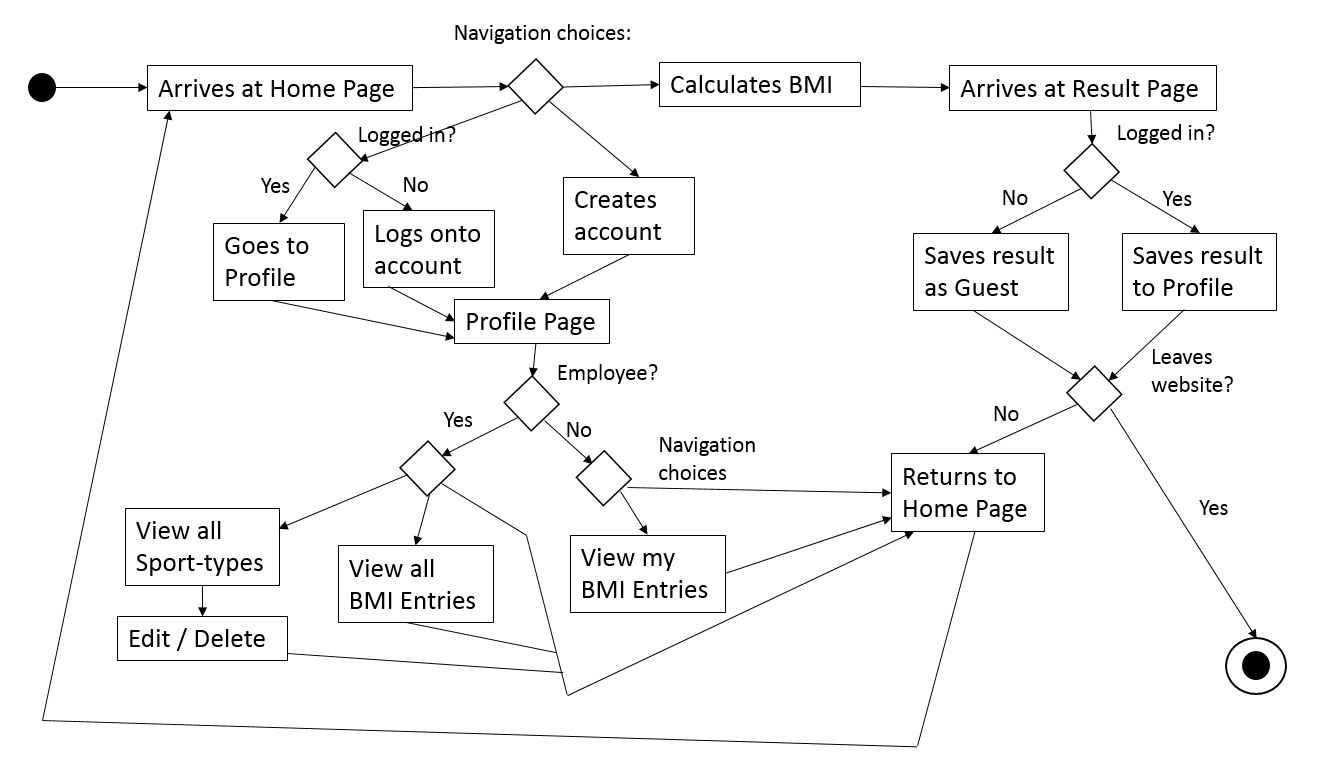
The following is a list of the user-stories we were provided with (translated to English from Danish, from the instructions found [*here*](https://cphbusiness.mrooms.net/mod/lesson/view.php?id=323702&pageid=4259)):

* **US-1: “*As a user, I want to be able to register additional information, when I perform a calculation, such as gender, age, sport-interest. My choices should be shown on the result-page.*”**
* **US-2: “*As an administration, I want the*** *‘****Calculate’ button to save users’ data and a timestamp to a database.*”**
* **US-3: “***As an administrator, I want to be able to view all BMI-calculations on a list.*”
* **US-4: “*As a visitor, I want to be able to sign-up as a user on the website, so that I can later view my history of calculations.*”**
* **US-5: “*As a logged-in user, I want to be able to view my own calculations.*”**
* **US-6: “*As an administrator, I want to be able to view a list with all the available sport-interests, so that I can keep an overview of all the options.*”**
* **US-7: “*As an administrator, I want to be able to remove a sport-interest, if it hasn’t been selected by any user, to avoid clogging the frontpage dropdown-list with unwanted choices.*”**
* **US-8: “*As an administrator, I want to be able to edit a sport-interest, i.e. the name, to potentially correct mistakes.*”**

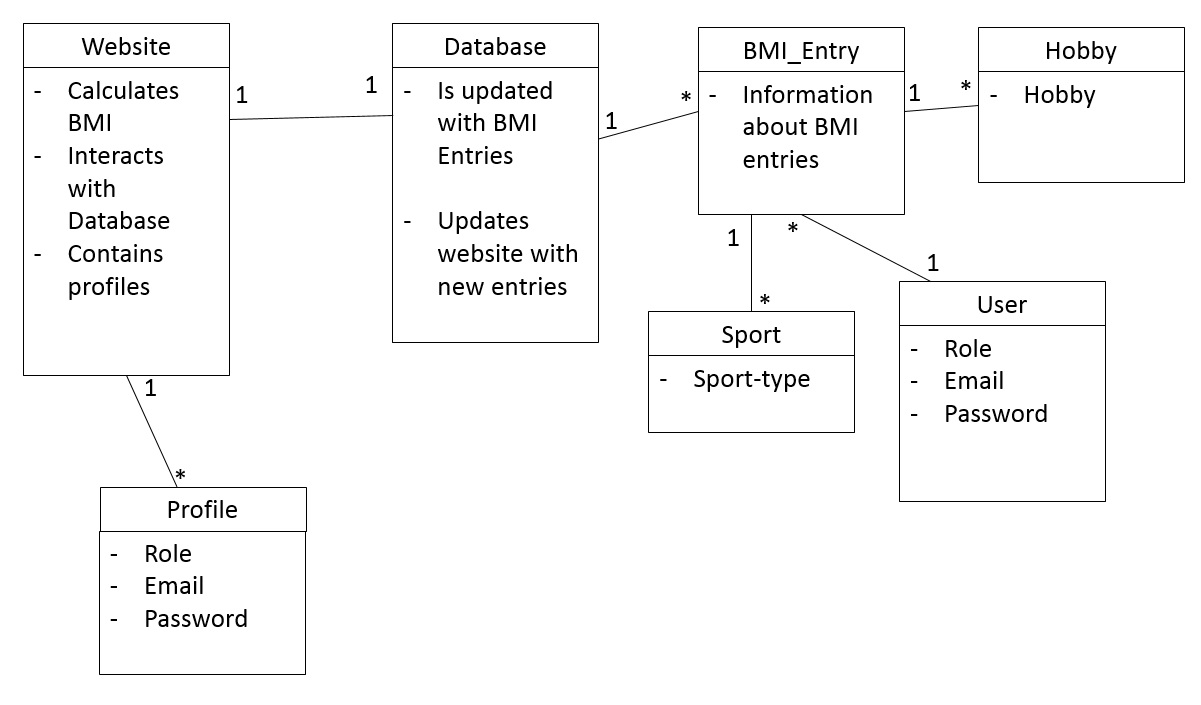
To make the website, we used:

* Digital Ocean and Tomcat9 to host it online and make it available to anyone with the link.
* IntelliJ to write the Java and HTML code.
* MySQL to handle the Database.
* GitBash to access our Digital Ocean droplet and handle the virtual-machine versions of Tomcat, our IntelliJ code, and the MySQL database, as well as monitoring the website’s status and such.

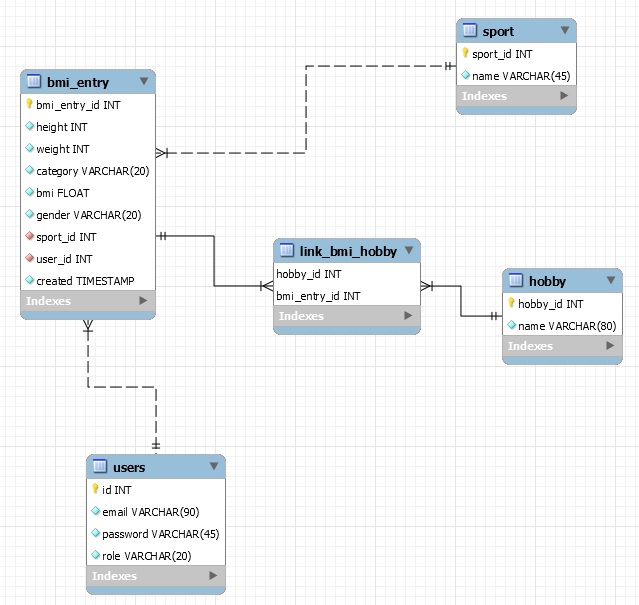
**Activity Diagram:**



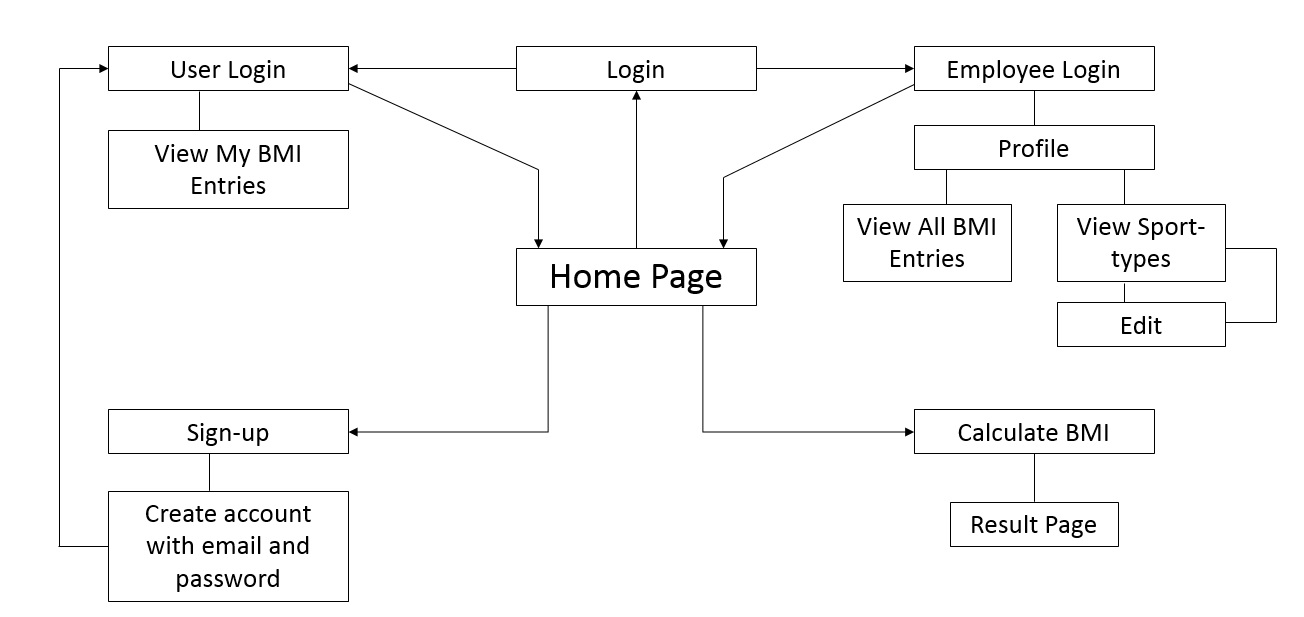
**Domain Model:**



**EER Diagram:**



**Navigation Diagram:**



These UML models/diagrams are based on our limited understanding of UML and were made after the fact. Neither of us really understand the necessity of these models, particularly when we have to make them ourselves based on a project mostly made by someone else, hence it is difficult for us to fully comprehend the flow of the code and the true navigation, and such.

Since most of the project is based on a framework we were given beforehand, there are a lot of things we don’t truly know about, such as the validation of user-input and how security is handled. We know that certain elements, such as prepared statements, prevents attacks where bad actors attempt to inject strings of code to get to our database’s contents, and we are also aware that the way our droplet is handled via SSH prevents certain attacks, such as man-in-the-middle attacks.

We encountered some issues in regards to the deployment part of our projects, which we both worked individually on, but in parallel. The main issue was that our only tomcat tutorial was outdated and referred to an obsolete (and inaccessible version) and the deployment tutorial referred to folders on our droplets that simply did not exist, hence we were forced to hardcode our setenv.sh (set environment) code into our Java code, which deviated from the tutorial quite a lot. However, we were both ultimately able to establish a connection to tomcat and deploy our “.war” files and create our websites, which are accessible and interactable by anyone with the link.

Our process consisted mostly of following the tutorials provided by our teachers, and some minor troubleshooting at various points.

The main thing we both would have liked was that the deployment part had a video tutorial as well, since this was arguably the most difficult part of the entire process.

We both learnt quite a lot about how websites are set up and how HTML and Java can work in tandem.

As for how we could have improved, it could have been neat to implement more interactability, to make some buttons (like the “profile”, “about”, and such) actually do \*something\*, as well as cleaning up certain elements to improve user-friendliness.

Kristoffer did end up adding a few additional things and fixing a flaw in the tutorial that left the result page with numbers instead of names for the sport-types and hobbies. He also changed the language to be uniformly English.

The following are our two BMI websites:

Kristoffer: <http://46.101.246.201:8080/bmi/>

Nicklas: <http://167.99.247.27:8080/bmi/>

GitHub:

Kristoffer: <https://github.com/Dosei-desu/CPH-Exercises/tree/master/2nd%20Semester/BMI>