Zach Thomas

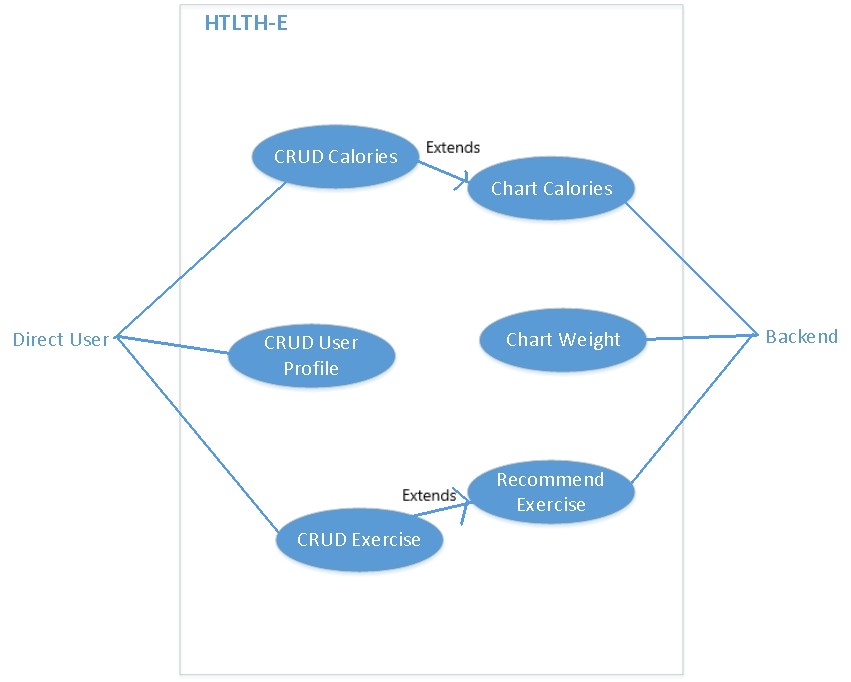
IST 361

Conceptual Design

Scenario

One day Zach, a college student who lives in State College, decided that he wanted to live a healthier lifestyle. He made a plan keep a journal of his activities, as well as a list of what he has eaten and his goals for the future. After considering all of the things to keep track of, Zach knew that he could use HLTH-E to keep track of everything he would need, as well as gain some recommendations for things to consider in the future. Zach entered the things he had eaten in the past day, as well as a 2000 calorie limit he wants to impose on himself for the future. He then sees that in order to be on track for the future he needs to start exercising. Zach selects an option to see recommended exercises for his future plan. Since he isn’t interested in gaining muscle right now, he selects cardio exercises. The next day, Zach gets up, eats breakfast, and writes what he had in his calorie log. He heads to the gym to start some exercises recommended by the application. After he gets home from the gym, he writes down what exercises he completed, and what he had for lunch. After reviewing the convenient chart showing his calorie gains and deficits for the day, Zach sees that to stay on track for his calorie limit, he can eat no more than 500 calories for dinner. After a few weeks, Zach wanted to see an overall graph of his progress so far. He selected the last three weeks of calorie tracking, and added his exercise log. He could see that he is starting to gain a better understanding of patterns that helped him stay on track, and those that didn’t. In the end, Zach saw a benefit in his lifestyle by keeping track of these certain aspects of life, even though it is costly to remember to track these metrics whenever possible. Some risks associated with programs such as these are users not “being honest with themselves,” and not committing enough time to this application.

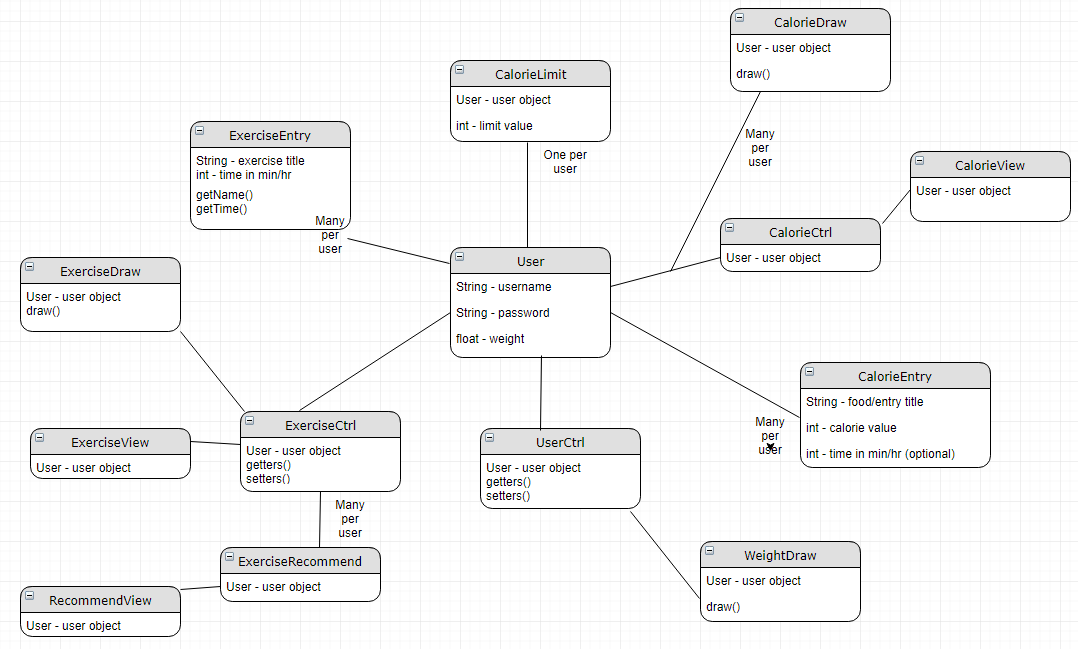
UML Case



Use Case Description

As of now, there are a total of six use cases that this application will contain. As development continues, there probably will be one or two glorified use cases added. The first and most basic use case is CRUD (create, read, update, delete) of the user’s profile information. Mainly, this will just simply be keeping track of login information to handle separate users, as well as keeping track of the user’s current weight. Updating this statistic may not be directly tied to the management of other credentials considering that it could be charted, but it will be part of the user profile, since every user will have a different weight to keep track of. The next use case is CRUD calories. This will entail users keeping a running log of what they have eaten over a specified time period. They can add a new food/calorie entry, see a list of already added information based on the time range, edit current entries, set a total calorie goal, and for whatever reason - delete entries. To go along with this, there will also be a use case to chart said log data in a visually appealing way (Chart calories). After that, there is CRUD Exercise. Just like CRUD calories, CRUD exercise includes a running log of exercise entries to create, edit, set goals, and delete. Again, there will be an accompanying chart function for this data, as well as separate exercise recommendations for future exercises. Finally, the application will encourage and keep track of weight changes if changed by the user, and give the option to draw a graph for a specified time range.

UML Diagram



Class Responsibilities

The classes will be organized in a typical MVC architecture. All of the information being collected will ultimately revolve around the user. Calorie and exercise entries will have basic information such as the title of the entry and data regarding it. CalorieView and ExerciseView will rely on their respective controllers to gather data from the model classes and/or corresponding local data files. UserCtrl will primarily relate to WeightDraw, to give the information the chart will need to render. UserCtrl also may be needed while interfacing with CalorieLimit, since that statistic is part of the basic User information from the beginning. ExerciseRecommend will be a logic file that contains the relations between the desired exercise style and the goals that the user wants to attain. For example, different exercises can burn more/less calories and affect the body differently depending on whether it is weight training or cardio. Given the user’s choices, RecommendView will display a greater showcase of recommendations and options. For the most part, all draw or view classes will take in a user object, and then proceed to us the controller files and data files to display an intuitive view and will allow the user to use the functions available to them, such as the creation, update, and delete of information throughout the application. The controllers and models will have the ability to write to several data files (.csv) for persistence. There will most likely be a separate file different user cases, including exercise, calories, user information, and recommendations.