

This diagram depicts how I envisioned my program’s GUI to function for the generate workout feature. However, I did end up changing some elements of it. I took out the input for height and weight, as I determined that it was unnecessary to the essential features of the program. I also decided to take out the options for endurance/strength/mix as the number of sets, reps, and weights should be left up to the user’s current physical ability. I also expanded the areas of the body that the user could work out to include every muscle in the body, which gives the user more flexibility.



The FinishedExercise class will extend the Exercise class, and will include the sets, reps, weight, and time the user finishes the exercise. The FinishedExercise object will be created to record exercises to a .TXT file and to read data from the .TXT file.



The FinishedExerciseModel is the model component of the MVC design pattern, and will be manipulated by two controllers, one for the record workout function, and one for the view stats function.

Test Plan Outline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Feature | Trigger | Expected Outcome | Result | Notes |
| Generate Workout | Generate Workout option with preferences | List of exercises | Success! | Works well, limited options when user does not have access to gym |
| Record Workout | Record Workout option with save button | Saves data to .TXT file | Success! | Save button works, confirmation also works |
| View Stats | View Stats option with exercise | Graph with last five workouts | Success! | Graph looks great, however, no option to change which stats appear. |