**Welcome to Terrier FitnessGraphical user interface, application

Description automatically generated**

# What is Terrier Fitness?

Terrier Fitness is an iOS developer app based in SWIFT and written with Xcode. The app provides the user with two main features. An organization system to help keep track of their workouts, and a default setting that gives the user a list of various workouts. The app can be used by a variety of people either looking into getting into fitness and are unsure where to start or for people who need a way of organizing their current exercise regimen. Hiring a trainer can be expensive and the internet has an overwhelming amount of information about the best workouts for each muscle group. Our app narrows down the selection to what we thought was the best workouts for each specific muscle group and has many cardio options as well.

# The Program Works? YES.

**Basic Steps of the code**

1. **The program starts by creating the Home View Controller Scene. This View serves as the home page for our app in which it lays out the 7 days of the week where the user can select any of the days to create their regiment.**
2. **Then the program then navigates inside the “ViewController” for whatever day was selected. From there the main home view will use the navigator controller to move to whatever view was selected based on the day.**



1. **Once in the Workout page for a specified day, the program will provide the user with an assortment of options including**

* **Arm**
* **Back**
* **Shoulder**
* **Legs**
* **Core**
* **cardio**

**If one of the default buttons is selected a random workout for that specific muscle group will be added to the users list from an array initialized before hand. A second option available is to manually enter any workouts to the list to make it customizable.**

1. **The program will shuffle the order of the exercises and only increment forwards to make sure that no workout is selected more than once when selecting the default workouts. This process is repeated for each default workout.**

**Text

Description automatically generated**

1. **An additional function the app has is to specify the Sets and Reps (Repetition) for the workout and has it displayed next to the workout**
2. **JUST REALIZE THAT ALREADY THE PLANNER IT HAS BEEN CREATED!**
3. **The User can clear a workout in the list this is done by replacing with zero which effectively deletes the line. There s a clear button that is linked with this functionality.**

**Keywords the program uses:**

**Text Field Controller 2**

**Home View Controller Scene**

**Navigator Controller (How it works)..**

**Workout selection Sunday Custom.**

**Clear**

**Label 2 controller**

**Arm**

**Back**

**Shoulder**

**Legs**

**Core**

**Clear Button**

**Challenges Faced**

Not finding a C++ template to create the iOS app and having to figure out how to work with SWIFT was a major hurdle we had to deal with; being this a new coding language. More specific, on how we manage the interface to make it link with the code by using this new language.

Being able to communicate the ideas between all 5 members was not an easy task, when some students focus on the code, others on documentation and others on the interface.

**Why this was a good project and deserves 15% of your grade**

1. Learning SWIFT and getting real world experience with app development: teaching ourselves from scratch how to make the code.

2. The result was a very useful application: simply a practical planner. Maybe it does not have the fanciest feature a fitness app can have but sometimes we realize that what you really need is an organizer for your ideas and this asset it provides benefits to me and all my friends.

3. Good projects are not only based on results but in efforts and in our case, we are pretty sure we hit both. Effort was involved throughout every process of its making. From thinking about the idea, then finding a way to make an app with no previous experience and thinking about how will look on the interface and playing with front and back-end codes at the same time.

4. Working hard to make it compile and doing constant research for new features to add.

5. We learned a lot.