Functional Features

1. Input of squat/bench/deadlift 1 rep maxes
2. Desired Program Type
   1. Upper/Lower split
   2. Full body
   3. Brosplit
   4. PPL
3. Extra Equipment
   1. Bar Types
      1. Cambered Bar
      2. Spider Bar
      3. Safety Squat Bar
      4. Buffalo Bar
      5. Swiss Bar
      6. Ez Curl Bar
      7. Tsunami Bar
   2. Bands
   3. Chains
   4. Boards
4. Input of weaknesses for each compound
   1. Bench Press
      1. Off the chest
         1. Spoto press
         2. Pin press
         3. 3 count pause
      2. Mid lift
         1. Pin press
         2. Slingshot
         3. 2 Board Press
      3. Lockout
         1. Banded bp
         2. Close grip bench
         3. Slingshot bench
   2. Squat
      1. In the hole
         1. Box squats
         2. Pin squats
      2. Right above parallel
         1. Box squats
         2. Pin squats
      3. Lock out
         1. Banded squats
         2. Reverse banded squats
   3. Deadlift
      1. Off the floor
         1. Paused deadlifts right off of floor
         2. Deficit deadlifts
      2. At knees
         1. ½ + 1 deadlifts
         2. Paused Deadlifts right below knees
      3. Lock out
         1. Block pulls
         2. Rack pulls
         3. Paused Deadlifts above knees
5. Injury checker
   1. Eliminates certain movements based on questionnaire answers
   2. Is x painful for you?

* Knee extension
* Hip extension
* Knee flexion
* Hip flexion
* Lower back pain
* Elbow extension
* Elbow flexion
* Overhead Movement
* Benching press

1. Follow-up questionnaire after each block
   1. Difficulty
   2. New Accessory work/variants
   3. Test maxes or do another block?
2. Lb ←→ Kg conversion
3. Macronutrient Calculator
4. Social Media aspect - add friends and automatically share PRs
   1. Every time you achieve a goal it automatically posts
   2. User isn’t allowed to make posts
   3. User can like when a friend achieves a goal
5. Desired Block Length
6. Beginner, Intermediate, Advanced Lifter selection
7. Raw vs. Equipped Lifting
8. Peaking Cycle (1-4 weeks)
   1. For testing maxes

Nonfunctional Requirements

1. Dictionary for Terms Used/Explanations
   1. Volume
   2. Intensity
   3. Reps
   4. Sets
   5. Programming Types
      1. PPL
      2. Brosplit
      3. Upper/Lower
      4. Full Body
   6. Macronutrients
      1. Carbs, Fats, Protein
   7. Micronutrients
   8. Types of Lifters
   9. Resources For Beginners (form videos, etc)
2. Remembers User Input
3. Users will be anyone that is physically capable of moving a barbell
4. Documentation behind Lifter Types
5. Takes under 5 seconds to generate program
6. Security
7. Oracle Database
8. System checks and updates server every time it is opened and the device is connected to the internet.

Due Monday Sept 17:

1. Entire feature list complete by Monday

2. Identify a least 10 use cases, flesh 1 out completely and send it to prof, it’s a deliverable.

Between Monday and Wednesday complete all 10 use cases…could have like 30, just have 10. He wants a table format…most important

|  |  |
| --- | --- |
| **Use Case ID:** | 1.1 |
| **Use Case Name:** | ORMInput and Units Used |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 13 |
| **Last Revision Date:** | September 17 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base program percentages off the user input. |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab. |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 1.2 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | 5a. User enters number larger than 9999Lb or 9999Kg  1. Next button disabled, dialog box saying “Invalid input: must be less than or equal to 9999”  2. User changes number  3. User continues normal flow |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change them (due to time off, or testing new maxes) |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.2 |
| **Use Case Name:** | InputLifterLevel |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base program blocks and layout on lifter level |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. Completes user case 1.1 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 1.3 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (Must select one of three levels: Beginner, Intermediate, or Advanced, to continue) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max and unit metric inputs  Lifter experience level input |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change it (due to progressing as a lifter) |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.3 |
| **Use Case Name:** | InputProgramType |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base program percentages off the user input. |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. Complete use case 1.2 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. Use case 1.4 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1-2. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (Must choose a program type) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max inputs  Lifter experience level input |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change it. |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.4 |
| **Use Case Name:** | Input of Limitations/Injuries |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base program percentages off the user input. |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab  4. User completes use case 1.3 |
| **Postconditions:** | 1. Stores user input  2. Question fades.  3. User case 1.5 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1-3. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 4 |
| **Exceptions:** | N/A (give them checkboxes) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max inputs  Lifter experience level input  Program Input |
| **Frequency of Use**: | When the user first logs in or if they want to change |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.5 |
| **Use Case Name:** | Weakness Collection for Compound Lifts |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base programmed lifts off of user’s weak points |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. User completes 1.1-1.4 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 1.6 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire questions 1 - 4. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (User must select a button to proceed) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change them (due to time off, or testing new maxes) |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.6 |
| **Use Case Name:** | Input Equipped or Raw |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base programmed lifts |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. User completes 1.1-1.5 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 1.7 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1-5. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (User must select one to continue) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max and unit metric inputs  Lifter experience level input  Program type input  Limitations/injury input  Weakness input |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change them. |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.7 |
| **Use Case Name:** | Input Desired Lifting Frequency |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base programmed lifts |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. User completes 1.1-1.6 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 1.8 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1-5. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (User must select one to continue) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max and unit metric inputs  Lifter experience level input  Program type input  Limitations/injury input  Weakness input |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change them. |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 1.8 |
| **Use Case Name:** | Input Extra Possible Equipment |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect information on actor and base programmed lifts |
| **Trigger:** | User starts to fill out the questionnaire. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on questionnaire tab.  4. User completes 1.1-1.7 |
| **Postconditions:** | 1. Stores user input.  2. Question fades.  3. User case 2.1 begins |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on questionnaire button.  5. User fills out questionnaire question 1-5. |
| **Alternative Flows:** | 4a. While the user is in the questionnaire, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A (User must select one to continue) |
| **Includes:** | Register new User  Sign in  Questionnaire button pressed  One rep max and unit metric inputs  Lifter experience level input  Program type input  Limitations/injury input  Weakness input |
| **Frequency of Use**: | When the user first logs in, and then any time the user would like to change them. |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They know answers to the question |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 2.1 |
| **Use Case Name:** | Input Weight on Macronutrient Calculator |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect weight information on actor in order to calculate macronutrient recommendations. |
| **Trigger:** | User starts to fill out the calculator. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on on macronutrient calculator tab. |
| **Postconditions:** | 1. Stores user input. |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on macronutrient calculator button.  5. User fills out first input box asking for weight. |
| **Alternative Flows:** | 4a. While the user is in the calculator, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A |
| **Includes:** | Register new User  Sign in |
| **Frequency of Use**: | Whenever the user wants to calculator their macronutrient recommendations. |
| **Special Requirements:** | Requires connection to database to store information. |
| **Assumptions:** | They know the answer to the question. |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 2.2 |
| **Use Case Name:** | Input Age on Macronutrient Calculator |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | October 7 |
| **Last Revision Date:** | October 7 |
| **Actors:** | Workout Enthusiast |
| **Description:** | Collect age information on actor in order to calculate macronutrient recommendations. |
| **Trigger:** | User starts to fill out the calculator. |
| **Preconditions:** | 1. User registers for an account.  2. User logs in account for the first time.  3. User clicks on on macronutrient calculator tab. |
| **Postconditions:** | 1. Stores user input. |
| **Normal Flow:** | 1. Use case begins when user accesses the webapp  2. User registers for an account.  3. User logs in account for the first time.  4. User clicks on macronutrient calculator button.  5. User fills out input box asking for age. |
| **Alternative Flows:** | 4a. While the user is in the calculator, they leave  1.App closes until user returns  2.When user returns continue normal flow from step 3 |
| **Exceptions:** | N/A |
| **Includes:** | Register new User  Sign in |
| **Frequency of Use**: | Whenever the user wants to calculator their macronutrient recommendations. |
| **Special Requirements:** | Requires connection to database to store information. |
| **Assumptions:** | They know the answer to the question. |
| **Notes and Issues:** | N/A |

|  |  |
| --- | --- |
| **Use Case ID:** | 3.1 |
| **Use Case Name:** | Social Media Page |
| **Created By:** | Team |
| **Last Updated By:** | Team |
| **Dated Created:** | September 18 |
| **Last Revision Date:** | September 18 |
| **Actors:** | Workout Enthusiasts |
| **Description:** | Shares workout accomplishments with friends |
| **Trigger:** | User clicks on the Social media tab/button |
| **Preconditions:** | 1. User logs into account |
| **Postconditions:** | 1. User goes back to main page |
| **Normal Flow:** | 1. User Logs in  2. User accesses the social media page |
| **Alternative Flows:** | N/A |
| **Exceptions:** | 2a. User tries to access page but has no connection  1a. User waits to reconnect  1b. User leaves app to reconnect |
| **Includes:** | Register new User  Sign in |
| **Frequency of Use**: | Whenever the User wants to see his friend feed |
| **Special Requirements:** | Requires connection to database |
| **Assumptions:** | They have friends |
| **Notes and Issues:** | N/A |

By end of week (17 sept-24 sept)

Feature cases

All use cases

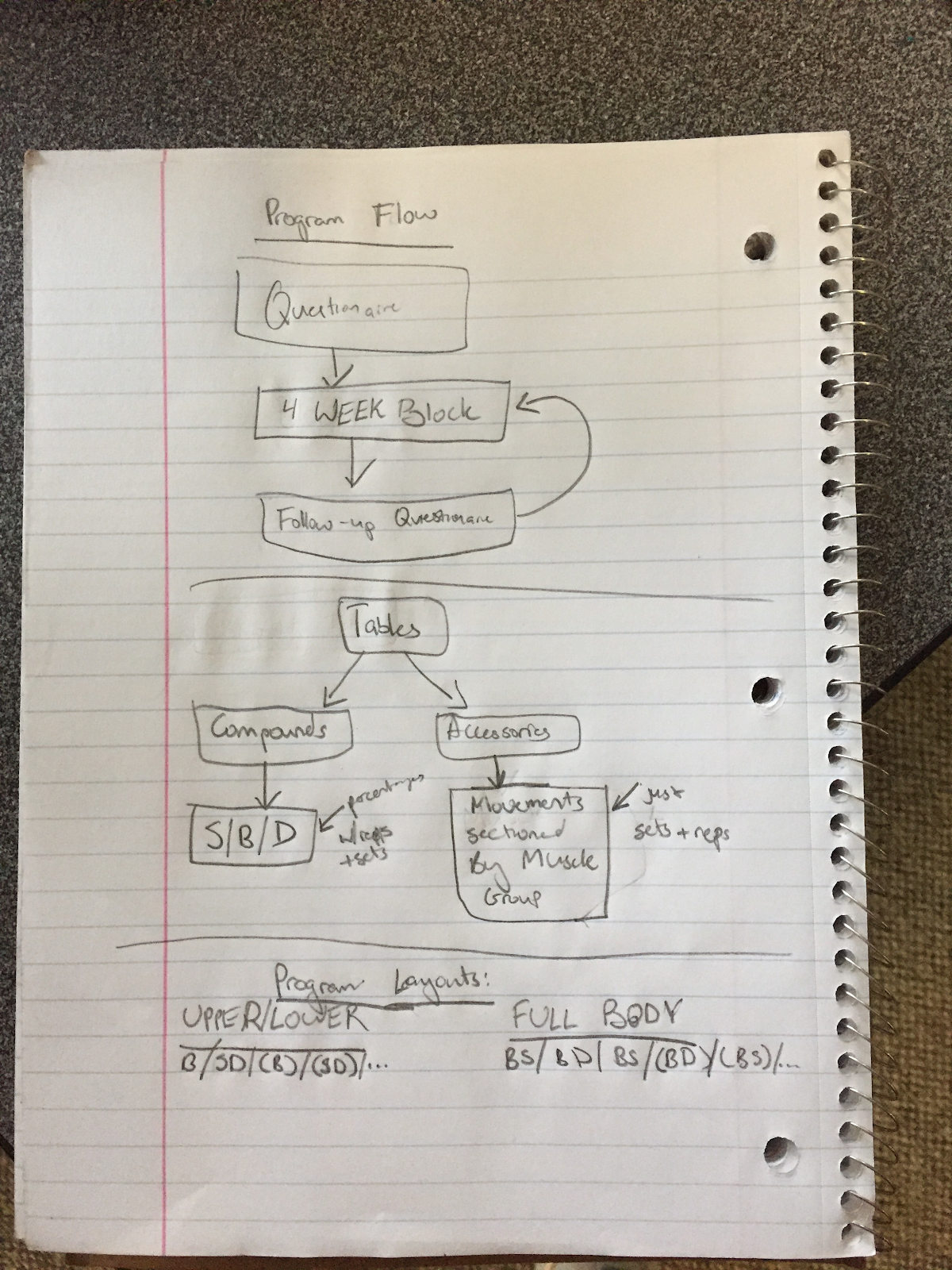
Use case diagrams

Identification of the class candidates

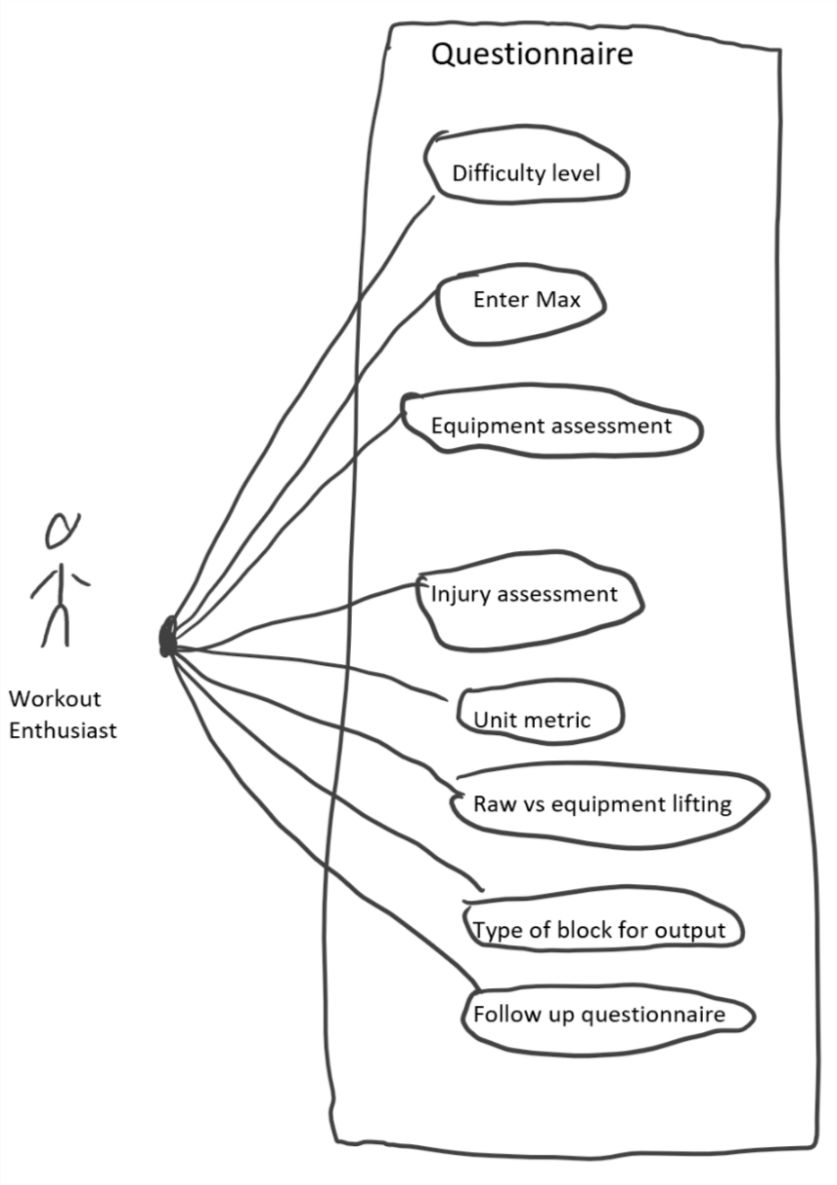
Class List

1. User (Lifter)
2. Program Maker
   1. Beginner Program Maker
   2. Intermediate Program Maker
   3. Unequipped Advanced
   4. Equipped Advanced
3. Questionnaire
4. Social Media
   1. Post
      1. PRs
      2. Finishing a session
      3. Finishing a block
   2. Add Friend (Function)
5. Helpful Info
   1. Dictionary
   2. Helpful Information
6. Macro Counter

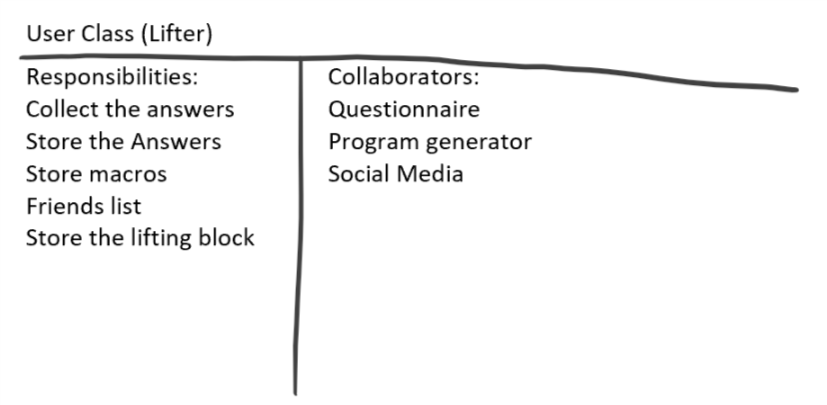
* User
  + Stores
    - Blocks Created
    - Answers to Questionnaire
    - Friend list
  + Methods
    - Retrieve current block
    - React to Post in feed
    - Add/Remove Friend
    - Getters/Setters
* Program Maker
  + Stores
    - Nothing
  + Methods
    - Create Block
* Questionnaire Class
  + Stores
    - Questions
  + Methods
    - Retrieve Questions
    - Save Answers to User Class
* Social Media Class
  + Stores
    - Posts made every week (Sunday -> Saturday)
    - Date Post was made
  + Methods
    - Make new Post
* Post Class
  + `Stores
    - Data within post
    - Reactions to post
  + Methods
    - Like a post
    - Clear Posts
* Helpful Information Class
  + Stores
    - Dictionary Of Terms
    - Links to helpful videos and websites
  + Methods
    - Getters
* Macro Counter Class
  + Stores
    - Macros
  + Methods
    - Getters/Setters



Use Case Diagram



CRC Card



Database outline:

We will run a MySQL Relational Database that will store information on users that will be used to generate their workout block. We will keep the tables to a minimum, we will have three tables. The first will store usernames, passwords and friends, the second will contain the equipment that each user has access to, and the third will contain Personal Records, their current maxes for different workouts and benchmarks they reached in pervious block.

9/19/18 class

State Diagram

Multiple Activity Diagrams (Use cases essential)

