****

***Software Engineering***

***Test Plan Document:***

***Iteration 2***

*James Dadson, Jonathan Miller, Colin Newton, Michael Yilma*  
*RUFit*  
*teammapletree.weebly.com*  
*4/1/2019*

**Table of Contents**

**1** **Introduction 3**

1.1 Iteration Overview 3

1.2 Items Not Covered by These Test Cases 3

1.3 Bug Tracking 3

1.4 Quality Control 3

1.5 Adequacy Criterion 3

**2** **Test Cases 4**

**3** **Iteration 0 – Bug Reports 5**

**4** **Customer Acceptance Testing 6**

# 

# **Introduction**

## 

## **Iteration Overview**

In Iteration 2 we are developing the Goals page, introducing a calendar for selecting different days, adding a notes and filters section to the add an exercise pages. The goals page allows users to create, update, complete, and delete their workout goals. The calendar allows for users to select a day and view their workout for their selected day. This allows the user to view their past and future workouts. The note component of the add an exercise page allows for users to add a specific description of their workout. The filter component of the add an exercise page allows for users to filter the exercises and narrow down the results, allowing for easier selection.

***Guidelines:***

* Goals page: has two different sections
  + Creating a goal
    - Assumption:
      * The start date for the goal is the day that the goal is created.
      * The goals title is required.
      * The goals description is optional.
      * The goals end date is optional.
    - Tests:
      * Will run regex expression test on the goal name field.
      * Will run regex expression test on the goal description field.
      * Will run regex expression test on the end date field.
      * Will test the day selected on the calendar for the goals end date is accurate.
  + Viewing the goals page
    - Assumption:
      * Goals have already been added to the database to be viewed by the user.
    - Tests:
      * Will make sure that all uncompleted goals appear in the goals section.
      * Will make sure that all completed goals appear in the completed goals section.
      * Will test that the goals can be edited.
      * Will test that the goals can be deleted.
      * Will test that the goals can be completed.
      * Will test that completed or expired goals can be renewed to current goals.
      * Will test the create goal button to make sure it takes the user to the create a goal page.
* Calendar:
  + Assumptions:
    - We assume the calendar we are implementing has 7 days a week.
  + Tests:
    - We will test to make sure the calendar has all 12 months.
    - We will test to make sure the calendar take leap year into account.
    - We will test to make sure that the calendar automatically selects the current day.
    - We will test the calendar to make sure we can select previous days other than the current day.
    - We will test the calendar to make sure we can select future days other than the current day.
    - We will test to make sure the calendar to make sure that the day updates to the selected day.
* Notes Section
  + Assumptions:
    - We are assuming that the notes section is optional for each exercise
  + Tests:
    - We will run regex expression tests to ensure that the information being submitted is not harmful.
* Filter exercises section:
  + Assumptions:
    - There is not filter selected when the user lands on the page.
  + Tests:
    - We will test that the filter selected only shows the workouts that are relevant to the filter.
    - We will test that all the filters show up in the drop down for the filters.

## **Items Not Covered by These Test Cases**

No, currently we are trying to test all functions and processes that we can think of.

## **Bug Tracking**

We will be using a database management system to track bugs in the database, namely, PHPmyAdmin. The DBMS will make it easier to identify where these bugs are occuring while practically testing features will be our main method of reporting the bugs*.*

## **Quality Control**

We will follow our quality standards, that every member of the group will adhere to during the testing process. We will confirm that we have a good test plan by ensuring that we cover every function of the iteration.

## **Adequacy Criterion**

As a team, we decided that if 95% of the tests pass, then testing will be complete for this iteration. We plan to discuss this number with our client to decide if they agree.

# **Test Cases**

*This section details for each test the exact steps necessary to perform the test and the expected results, and provides a section for recording the observed results. This table is the most important section of the document. It comprises both the plan and the results of executing the plan. That is, the testing team fills it in as the testing process proceeds. Failed tests should result in bug reports.*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test #** | **User Story** | **Action / Input** | **Expected Result** | **Actual Result** | **P/F** | **Notes** | **Date test performed** | **Tester** |
| 1.a | Add Note | Add a note to workout | Note correctly attaches to an exercise. | Note correctly attaches to an exercise. | P | Added Correctly | 4/1/2019 | James Dadson |
| 1.b | Add Note | Attempt with harmful code | Note is not added. | Note not added. | P | Note is not added | 4/1/2019 | James Dadson |
| 2.a | Select day | Select day from calendar | The day selected and corresponding “add exercise” page is opened. | The correct day is opened and the add exercise page is open correctly. | P | Opens Correctly | 4/1/2019 | James Dadson |
| 2.b | Select day | Calendar cycles through dates correctly. | The calendar iterates correctly. | The calendar iterates correctly and shows the corresponding workout for that day. | P | Iterates correctly. | 4/1/2019 | James Dadson |
| 3.a | Create a Goal | Add goal name | Data correctly added to database. | Data correctly adds to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 3.b | Create a Goal | Add goal description | Data correctly added to database. | Data correctly adds to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 3.c | Create a Goal | Add goal end date | Data correctly added to database. | Data correctly adds to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 3.d | Create a Goal | Add goal start date | Data correctly added to database. | Data correctly adds to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 4.a | Update a Goal | Change the description of a goal. | Update pushed to the database. | Update pushed to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 4.b | Update a Goal | Change the completion date of the goal. | Update pushed to the database. | Update pushed to database. | P | Added Correctly | 4/1/2019 | Michael Yilma |
| 4.c | Update a Goal | Change the name of the goal. | Update pushed to the database. | Update pushed to database. | P | Added Correctly. | 4/1/2019 | Michael Yilma |
| 5.a | Delete a Goal | Delete a Goal selected. | Delete pushed to the database. | Delete pushed to the database. | P | Deletes Correctly. | 4/1/2019 | Michael Yilma |
| 6.a | Complete a Goal | Complete a Goal selected. | Goal categorized as completed. | Update pushed to database. | P | Updates the Database Correctly. | 4/1/2019 | Michael Yilma |
| 7.a | Filter Exercises | Filter by date selected | Workout filtered by the completion date. | Query not pushed to the database | F | Feature not implemented yet | 4/1/2019 | Michael Yilma |
| 7.b | Filter Exercises | Filter by name selected. | Workout filtered by the goal name. | Query not pushed to the database | F | Feature not implemented yet | 4/1/2019 | Michael Yilma |
| 8.a | Plan a Workout | Select a date and log workouts for future dates. | User redirected to “add exercise”. | User is redirected to the corresponding Add Exercise page. | P | Redirects to the corresponding Add Exercise page. | 4/1/2019 | James Dadson |

***Guidelines:***

* *Although the information in the table above indicates that each test comprises a line, actual descriptions may require multiple lines.*
* *The* ***Test #*** *can represent a category or set of tests related to a user story. In these situations, add a suffix to the number using small Roman letter. For example, use Test 2.a, Test 2.b, …*
* *All tests should have a specific purpose. For example, a test may evaluate the satisfaction of a specific requirement, in which case, the user story number should be included.*
* *A test (****Action****) may comprise a series of steps. Each step should be specific with regard to relevant user input and data files. That is, the information in this document is a prescription for executing the test.*
* *Some steps may be common to more than one Test. Some Tests may take up where others left off. There is no need to duplicate this content. In these cases, refer in the* ***Action*** *section to the* ***Test #*** *and Step Numbers that should be performed.*
* ***Expected Results*** *describe the exact output that should be produced by the test.* ***Actual Output*** *is what happened when the test was run. Note that for non-functional tests, the contents of this field might include elapsed time, memory consumed, etc. For failing tests, the output might include information about exceptions raised.*
* ***P/F*** *denotes Pass / Fail. A test should only pass if its expected output matches its actual output.*
* *The* ***Notes*** *field can provide details so that the team can repeat the test in future iterations.*

# **Iteration 0 – Bug Reports**

*This section details bugs (or failed tests) found during the current iteration.*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bug ID** | **Originator** | **Submit Date** | **Summary** | **Severity**  [Critical | major | normal | minor | trivial | enhancement] | **Platform**  [PC | Mac] | **OS**  [Windows | Mac | Linux] | **Browser**  [IE | Firefox | Chrome | Safari] | **Completion Date** |
| 1.a | James Dadson | 3/25/19 | Add a Note has not been implemented. | Normal | All | All | All | 4/1/19 |
| 1.b | James Dadson | 3/25/19 | Add a Note allows code injection. | Major | All | All | All | 4/1/19 |
| 2.a | James Dadson | 3/25/19 | The corresponding “add exercise” page is not opened. | Major | All | All | All | 4/1/19 |
| 3.d | Michael Yilma | 3/25/19 | The goal start day is not correctly added to the database. | Normal | All | All | All | 4/1/19 |
| 4.a | Michael Yilma | 3/25/19 | The change the description of a goal does not add to the database. | Minor | All | All | All | 4/1/19 |
| 4.b | Michael Yilma | 3/25/19 | The change the completion date of a goal does not add to the database. | Minor | All | All | All | 4/1/19 |
| 4.c | Michael Yilma | 3/25/19 | The change the name of a goal does not add to the database. | Minor | All | All | All | 4/1/19 |
| 5.a | Michael Yilma | 3/25/19 | The delete a goal does not add to the database. | Major | All | All | All | 4/1/19 |
| 6.a | Michael Yilma | 3/25/19 | The complete a goal does not add to the database. | Major | All | All | All | 4/1/19 |
| 7.a | Michael Yilma | 3/25/19 | The filter exercises by completion date does not work. | Minor | All | All | All |  |
| 7.b | Michael Yilma | 3/25/19 | The filter exercises by goal name does not work. | Minor | All | All | All |  |
| 8.a | Michael Yilma | 3/25/19 | The user is not redirected to “add exercise”. | Major | All | All | All | 3/25/19 |

Currently the only tests not passing are the ones that do not currently have the feature necessary to properly test.

# **Customer Acceptance Testing**

*This section details for each test the steps necessary for the customer to perform acceptance testing and the expected results, and provides a section for recording the observed results. The customer should fill in this table at then end of the iteration/phase. Failed tests should result in bug reports.*

Customer Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test #** | **User Story** | **Action / Input** | **Expected Result** | **Actual Result** | **P/F** | **Notes** | **Date test performed** | **Tester** |
| 1.a | Add Note | Add a note to workout | Note correctly attaches to an exercise. |  |  |  |  |  |
| 1.b | Add Note | Attempt with harmful code | Note is not added. |  |  |  |  |  |
| 2.a | Select day | Select day from calendar | The day selected and corresponding “add exercise” page is opened. |  |  |  |  |  |
| 2.b | Select day | Calendar cycles through dates correctly. | The calendar iterates correctly. |  |  |  |  |  |
| 3.a | Create a Goal | Add goal name | Data correctly added to database. |  |  |  |  |  |
| 3.b | Create a Goal | Add goal description | Data correctly added to database. |  |  |  |  |  |
| 3.c | Create a Goal | Add goal end date | Data correctly added to database. |  |  |  |  |  |
| 3.d | Create a Goal | Add goal start date | Data correctly added to database. |  |  |  |  |  |
| 4.a | Update a Goal | Change the description of a goal. | Update pushed to the database. |  |  |  |  |  |
| 4.b | Update a Goal | Change the completion date of the goal. | Update pushed to the database. |  |  |  |  |  |
| 4.c | Update a Goal | Change the name of the goal. | Update pushed to the database. |  |  |  |  |  |
| 5.a | Delete a Goal | Delete a Goal selected. | Delete pushed to the database. |  |  |  |  |  |
| 6.a | Complete a Goal | Complete a Goal selected. | Goal categorized as completed. |  |  |  |  |  |
| 7.a | Filter Exercises | Filter by date selected | Workout filtered by the completion date. |  |  |  |  |  |
| 7.b | Filter Exercises | Filter by name selected. | Workout filtered by the goal name. |  |  |  |  |  |
| 8.a | Plan a Workout | Select a date and log workouts for future dates. | User redirected to “add exercise”. |  |  |  |  |  |