Phase 1

Rodentia Inventory Tracking Application (R.I.T.A.)

Team 4

Jacob Hatcher | Ken Mabry | Tiffany Matthew | Casey Sapp | Mili Sharma

University of Maryland University College

Revision: 1.0

Date: 06/23/2019

Class: CMSC 495 6380

[Phase 1 0](#_Toc12209089)

[Revision History 3](#_Toc12209090)

[Project Plan 4](#_Toc12209091)

[Overview 4](#_Toc12209092)

[Goals 4](#_Toc12209093)

[Users Characteristics 4](#_Toc12209094)

[Requirement Specifications 4](#_Toc12209095)

[Security 6](#_Toc12209096)

[System Specification 7](#_Toc12209097)

[Configuration Management 7](#_Toc12209098)

[Project Schedule 8](#_Toc12209099)

[Test Plan 10](#_Toc12209100)

[Introduction 10](#_Toc12209101)

[Approach 10](#_Toc12209102)

[Pass/Fail Criteria 10](#_Toc12209103)

[Required Items and Data 11](#_Toc12209104)

[Test Scenarios 11](#_Toc12209105)

[Test Deliverables 14](#_Toc12209106)

[Test Environment 14](#_Toc12209107)

[User Guide 15](#_Toc12209108)

[Purpose 15](#_Toc12209109)

[Scope 15](#_Toc12209110)

[Users 15](#_Toc12209111)

[Authentication 15](#_Toc12209112)

[Main Screen Options 15](#_Toc12209113)

[Create New User: 16](#_Toc12209114)

[Create Record Entries: 16](#_Toc12209115)

[Modify/Update/Delete Records 17](#_Toc12209116)

[Report Details 17](#_Toc12209117)

[Error Messages 18](#_Toc12209118)

[Design 19](#_Toc12209119)

[Web Application Design 19](#_Toc12209120)

[Backend Database 26](#_Toc12209121)

[Phase 1 28](#_Toc12209122)

[Overview 28](#_Toc12209123)

[Testing 28](#_Toc12209124)

[Screenshots 29](#_Toc12209125)

[Next Steps 33](#_Toc12209126)

[References 34](#_Toc12209127)

[Team Contributions 35](#_Toc12209128)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Name | Description |
| 1.0 | 6/14/2019 | Jacob Hatcher | Combined Project and Test Plan and added Design |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Project Plan

# Overview

A research facility who studies addiction in humans has a pre-clinical facility where they breed rodents who have specific genetic make-ups. The expression of specific genes may play a role in addiction. In order to produce test subjects with these genes they must specifically breed certain rodents to produce the desired genotypes. A research animal database is needed to track information in regards to all of the animals.

# Goals

1. Provide a web-based interface to allow Principal Investigators, Geneticists, and Breeder Tech’s to log into the research animal database.
2. Allow access to all existing data for each animal in the facility, add new animals, update existing data, and remove animals from the database.
3. Provide readable reports based on information in the database.
4. Shortcuts to make entering large quantities of data quickly.

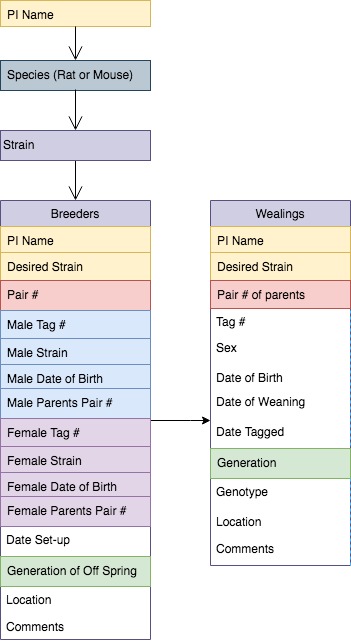
# Users Characteristics

1. Breeder Tech’s: Responsible for most of the data entry, and will use database on a daily basis.
2. Geneticist: Will utilize the readable reports to determine the efficacy of breeders and determine breeders for optimal pairings.
3. Principal Investigator (PI): Primarily will view database to review genotypes produced and the number of animals available to transfer to their laboratories.

# Requirement Specifications

***Database***

The database must contain the pertinent fields for breeders and their weanlings. Records must be made for each Principal Investigator (PI), species (rat or mouse), and strains approved under their protocols (see Figure 1). Only strains approved under PI’s protocols can be added. Tables for each animal must contain the relevant fields and be discernable based at least on the table above it in Figure 1.The data under the breeders and weanlings are shown in Figure 1. These will be used in the User Interface to add, update, remove, and view data and reports.



*Figure 1*: Tables contain fields for PI, species, and strain. Data fields under breeders and weanlings are specific for their classification.

***Visual***

The web-based user interface will provide a simple visual representation and allow the user to interact with the database. Features should allow the user to view, update, remove, and add to the database based on the PI, species, and strain. Additional functionality may be added as time and resources permit.

***Visual Functionality***

Once the three required fields (PI, species, and strain) are selected a limited number (exact number TBD) of breeder pairs should be displayed. All data fields for breeders (See Figure 1: Breeders) should be displayed in readable manner. A specific breeder pair can be searched by breeder pair number. Once a pair number is selected all weanlings for this pair should be displayed, and further narrowing of the search can be done by using other fields within the weanling tables (see Figure 1 for data fields). Just as the breeders, all weanling data fields should be displayed in a readable manner. All of these data fields should be editable, and easily saved for easy updating. Not all fields are required to hold data, as they will be updated as the animals move through the facilities breeding process.

***Visual Reports***

The following reports should be able to be visually populated if user selects to do so:

* 1. Show all breeding pairs who are still active, and have a set up date over 6 months prior to report generation.
  2. A weanling report should generate all weanlings who will turn 21 days old within a given set of dates.
  3. A report showing survivability of pups for a strain, broken down by pair numbers.

# Security

With the application primarily relying on database queries from the web application to view tables, add records, and update fields security, and especially the mitigation of SQL Injection Attacks, needs to be addressed. User passwords will be encrypted using a hashing algorithm before being sent to the database, so plain text passwords will never travel over the network or be stored. We will also be utilizing SQL stored procedures, parameterized queries, PHP Data objects, and escaped HTML where applicable to avoid SQL injection attack vulnerabilities.

# System Specification

Server: Apache Web Server; PHP 7.0 or later; MySQL 5.7 or later

Client: Google Chrome, Firefox, Edge, or Safari on Windows or MacOS

# Configuration Management

Version control will be handled using GitHub

--The project repository will be located at: <https://github>.com/<https://github.com/cmsc495-team4/main>

# Project Schedule

| **Task** | **Duration** | **Start** | **End** | **Team Member** |
| --- | --- | --- | --- | --- |
| Requirements Definition (Week 2) |  |  |  |  |
| Draft Requirements | 6 days | 5/27/19 | 6/1/19 | Tiffany |
| Review/Feedback | 1 day | 6/2/19 | 6/2/19 | All |
| Revise/Finalize | 1 day | 6/2/19 | 6/2/19 | Ken |
| Project Plan (Week 2) |  |  |  |  |
| Draft | 1 day | 6/1/19 | 6/1/19 | Jake |
| Review | 1 day | 6/2/19 | 6/2/19 | All |
| Finalize | 1 day | 6/2/19 | 6/2/19 | Mili |
| Test Plan (Week 3) |  |  |  |  |
| Draft | 6 days | 6/3/19 | 6/8/19 | Ken, Mili, Jake |
| Review | 1 day | 6/9/19 | 6/9/19 | Tiffany & Casey |
| Finalize | 1 day | 6/9/19 | 6/9/19 | Casey |
| User Guide (Week 3) |  |  |  |  |
| Draft | 6 days | 6/3/19 | 6/8/19 | Tiffany & Casey |
| Review | 1 day | 6/9/19 | 6/9/19 | Ken, Mili, Jake |
| Finalize | 1 day | 6/9/19 | 6/9/19 | Jake |
| Web Design (Week 4) |  |  |  |  |
| Draft | 5 days | 6/10/19 | 6/14/19 | Ken, Casey, Tiffany |
| Review | 1 day | 6/15/19 | 6/15/19 | Jake & Mili |
| Finalize | 1 day | 6/16/19 | 6/16/19 | Ken, Casey, Tiffany |
| Database Design (Week 4) |  |  |  |  |
| Draft | 5 days | 6/10/19 | 6/14/19 | Jake & Mili |
| Review | 1 day | 6/15/19 | 6/15/19 | Ken, Casey, Tiffany |
| Finalize | 1 day | 6/16/19 | 6/16/19 | Jake & Mili |
| Phase I (Week 5) |  |  |  |  |
| Create Database | 1 day | 6/17/19 | 6/17/19 | Jake |
| User login screen | 4 days | 6/18/19 | 6/21/19 | Ken & Mili |
| Create new user | 4 days | 6/18/19 | 6/21/19 | Tiffany & Jake |
| Test | 2 days | 6/22/19 | 6/23/19 | Casey |
| Finalize | 1 day | 6/23/19 | 6/23/19 | Casey |
| Phase II (Week 6) |  |  |  |  |
| Add new weanlings (individual and batch) | 3 days | 6/24/19 | 6/26/19 | Casey |
| Update weanling data for *WEANED* and *TAGGED* | 2 days | 6/27/19 | 6/28/19 | Tiffany |
| Add new breeding pairs | 3 days | 6/24/19 | 6/26/19 | Ken |
| Update breeding pairs data | 2 days | 6/27/19 | 6/28/19 | Jake |
| Test | 2 days | 6/29/19 | 6/30/19 | Mili |
| Finalize | 1 day | 6/30/19 | 6/30/19 | Mili |
| Phase III (Week 7) |  |  |  |  |
| Search/Filter weanlings | 5 days | 7/1/19 | 7/5/19 | Ken & Tiffany |
| Search/Filter breeders | 5 days | 7/1/19 | 7/5/19 | Casey & Mili |
| Test | 2 days | 7/6/19 | 7/7/19 | Jake |
| Finalize | 1 day | 7/7/19 | 7/7/19 | Jake |
| Final Delivery (Week 8) |  |  |  |  |
| Report: Currently unweaned | 4 days | 7/8/19 | 7/11/19 | Mili |
| Report: Tagged animals for a breeding pair | 4 days | 7/8/19 | 7/11/19 | Tiffany |
| Report: Breeders to retire | 4 days | 7/8/19 | 7/11/19 | Jake |
| Report: Breeders with increased pup mortality | 4 days | 7/8/19 | 7/11/19 | Casey |
| Test | 3 days | 7/12/19 | 7/14/19 | Ken |
| Finalize | 1 day | 7/14/19 | 7/14/19 | Ken |

Test Plan

# Introduction

​The purpose of this document is to outline the test plan for the Mighty Mouse Rodentia Inventory Tracking Application (R.I.T.A.).  The application will permit a user to log in with a unique username and password. There will be three categories of users, each assigned a specific role of either Principal Investigator, Breeder Tech, or Geneticist. Upon successful login, users, according to their respective roles, will be able to add new breeders or weanlings to the database, search for and update existing breeders or weanlings, and create reports for use by technicians and investigators alike.

# Approach

​The application development will take place in four phases with additional functionality added at each phase. Testing at the end of each development phase will consist of two levels performed by the QA lead or their designee. The first level will be Smoke Testing to ensure basic functionality (e.g., the application properly loads in browser) and peer review of newly added code for feedback and educational purposes. The second level will consist of functional tests wherein the scenarios for the current phase and all prior phases (Regression Testing) will be performed.

Sample data and scripts for the scenarios will be created and reviewed before testing begins, with expected results for each activity clearly defined.

# Pass/Fail Criteria

Test cases will be evaluated as “Pass” only if the actual test results match the expected results.

# Required Items and Data

All required data will be specified in advance of testing. A pre-defined, realistic set of data will be loaded to the database before testing begins. The required test script will contain additional data for the tests and will direct the actions of the software tester to modify or delete existing database records or to add new records as required.

# Test Scenarios

The following scenarios will be used for full testing of the application. The test activities will be divided among the four phases of development.

| Test Scenarios | | |
| --- | --- | --- |
| Activity | Action | Result |
| User Login | User with existing account logs in with a username and password | Successful login to application with correct Username and Role displayed onscreen. |
| Main Menu Navigation | User will select an action from the main menu. After the result is confirmed, the user will return to the main menu and select the next item. This will be repeated for each item in the menu. | After each menu selection, the correct corresponding page will be displayed |
| Data Entry/Creation |  |  |
| Add new strains, genotypes, generations, and locations to the database | From each respective record creation dialog within the application, enter all information required to create the applicable record.  A list of all input records will be supplied to the tester. | Via phpMyAdmin or similar tool, the tester will confirm that each SQL table is populated with the expected data from the predefined lists. |
| Create new user accounts | From the new user creation dialog, enter all information required to create a new user account.  A diverse list of users will be supplied to the tester. | Via phpMyAdmin or similar tool, the tester will confirm that each SQL table is populated with the expected data from the predefined list. |
| Create new animal entries | From the weanling creation dialog, enter all information required to create a new weanling record.  A diverse list of weanlings will be supplied to the tester. | Via phpMyAdmin or similar tool, the tester will confirm that each SQL table is populated with the expected data from the predefined list. |
| Assign strains to responsible Principal Investigators | From the strains dialog, assign each strain to its corresponding Principal Investigator.  A list of associations will be supplied to the tester. | Via phpMyAdmin or similar tool, the tester will confirm that each SQL table is populated with the expected data from the predefined list. |
| Create new breeding pairs | From the breeding pair creation dialog, enter all information required to create a new breeding pair record.  A list of pairings will be supplied to the tester. | Via phpMyAdmin or similar tool, the tester will confirm that each SQL table is populated with the expected data from the predefined list. |
| Data Modification/Deletion |  |  |
| Edit strains, genotypes, generations, and locations records | The tester will modify all records as directed by a predefined script. | The tester will verify each change by comparing the fields displayed on the resulting confirmation screen to the values supplied by the script. |
| Edit existing user accounts | The tester will modify specified user account records as directed by a predefined script. | The tester will verify each change by comparing the fields displayed on the resulting confirmation screen to the values supplied by the script. |
| Edit existing animal entries | The tester will modify specified animal records as directed by a predefined script. | The tester will verify each change by comparing the fields displayed on the resulting confirmation screen to the values supplied by the script. |
| Edit existing PI-to-Strain associations | The tester will modify specified PI-to-strain associations as directed by a predefined script. | The tester will verify each change by comparing the fields displayed on the resulting confirmation screen to the values supplied by the script. |
| Edit existing breeding pairs | The tester will modify specified breeding pair records as directed by a predefined script. | The tester will verify each change by comparing the fields displayed on the resulting confirmation screen to the values supplied by the script. |
| Delete specified strains, genotypes, generations, and locations records | The tester will delete specific records as directed by a predefined script. | Via phpMyAdmin or similar tool, the tester will confirm that each deleted record is no longer contained in its respective table. |
| Delete a user account | The tester will delete specified user account records as directed by a predefined script. | Via phpMyAdmin or similar tool, the tester will confirm that each deleted record is no longer contained in its respective table. |
| Delete an animal entry | The tester will delete specified animal records as directed by a predefined script. | Via phpMyAdmin or similar tool, the tester will confirm that each deleted record is no longer contained in its respective table. |
| Delete a PI-to-Strain association | The tester will delete specified association records as directed by a predefined script. | Via phpMyAdmin or similar tool, the tester will confirm that each deleted record is no longer contained in its respective table. |
| Delete a breeding pair | The tester will delete specified breeding pair records as directed by a predefined script. | Via phpMyAdmin or similar tool, the tester will confirm that each deleted record is no longer contained in its respective table. |
| Data Validation/Error Checking |  |  |
| A script will be provided to the software tester which directs the tester to perform certain activities such as (but not limited to):   * Create an account with a duplicate username * Create a breeding pair with two animals of the same sex * Enter a wean date prior to an animal’s birth date * Assign the same litter ID to animals with differing birth dates * Specify an animal’s parent’s ID as that of itself, sibling, or same/later generation. * etc… | The tester will follow the provided script to perform these activities. | The tester will verify the resulting message or error matches the activity’s expected result as defined by the script. |
| Data Reporting |  |  |
| A script will be provided to the software tester which directs the tester to generate specific reports such as (but not limited to):   * Report of aging breeder pairs * Report of weanlings to be weaned during the current week, next week, etc. * Report of pup survivability statistics * Report of a weanlings of a specific strain * etc… | The tester will follow the provided script to perform these activities. | The tester will confirm that the returned results matches the expected results specified in the test script. |

# Test Deliverables

|  |  |
| --- | --- |
| Deliverable | Description |
| Test Plan | Document to provide overall guidance of the testing efforts |
| End-of-Phase Test Scripts (4) | Test cases and test script to exercise the functionality of features implemented during the applicable phase. The test cases and scripts will be cumulative. |
| End-of-Phase Testing Reports (3) | Report to convey testing results at the end of each phase |
| Final Test Report | Report to convey cumulative test results from all phases |

# Test Environment

The test environment shall meet the following specifications:

|  |  |  |
| --- | --- | --- |
| **Requirements** | **Client** | **Server** |
| Operating System | Windows/MacOS/Linux | Linux |
| Web Browser | Chrome, Firefox, or Safari | N/A |
| Software | Adobe Reader, PDF print driver | Apache or LightSpeed |
| Hardware | PC/Mac desktop or laptop | Virtual x64 web hosting platform |
| Network | Internet access | Internet access |
| Database | N/A | MySQL 5.7 or higher |

User Guide

# Purpose

The purpose of the user guide is to provide application users with instructions on how to use the Rodentia Inventory Tracking Application (R.I.T.A) to allow Principal Investigators, Geneticists, and Breeder Tech’s to log into the research animal database.

# Scope

The user guide will provide detailed instruction related to creating and modifying data for each animal in the facility as well as creating and modifying user access. The user will also be able to run reports incorporating the necessary data needed for specific animal breeding’s and genotypes.

# Users

RITA Staff

* Principal Investigator
* Breeder Tech
* Geneticists

IT Support Staff

# Authentication

Access the RITA login via the link

Enter the username in the box labeled username

Enter the password in the box labeled password

Click the “Login” button

# Main Screen Options

After successful authentication the user is directed to the Welcome Screen with options to select from the following tasks:

* Add Pups
* Add/Update Animals
* Add/Update Pair
* Reports
* Sign Out

# Create New User:

To create a new user:

* + From the login page click the Create New User.
  + A login screen will appear for to login as an Administrator to continue
  + Upon logging in create a new user and password

# Create Record Entries:

Creating New Weanlings: click on the Add Pups link, the user will be directed to a new page to create a new entry.

* + Enter the PI\*
  + Enter the Parent Pair Number\*
  + Enter the Weanlings Date of Birth\*
  + Enter the number of Weanlings to add\*
  + Enter comments
  + Click the Add Weanling Button to save data

*\* required fields cannot be left blank*

Creating New Animals: click on the Add Animals link, the user will be directed to a new page to create a new entry.

* + Enter the PI\*, Species\*, Strain\*
  + Enter the Tag #, DOB, Wean Date, Tag Date
  + Enter the Location\* and Sex\*
  + Enter the Genotype and Litter ID
  + Select the Classification\* (pup, weanling, breeder)
  + Enter comments
  + Click the Add Weanling Button to save data

*\* required fields cannot be left blank*

Creating New Breeding Pair: click on the Add Breeder Pair link, the user will be directed to a new page to create a new entry.

* + Enter the PI\*
  + Enter the Species\*
  + Enter the Desired Strain\*
  + Enter Date Set Up for Breeding\*
  + Enter the male tag number\*
  + Enter the female tag number\*
  + Enter the generation\*

*\* required fields cannot be left blank*

# Modify/Update/Delete Records

To update Animals: click on the Update Animals link.

* + The weanling will need to be searched prior to making changes
  + Input the search data: Date of Birth, Tag ID, or Genotype
  + Hit the search button
  + Select the desired animal to update
  + The animal record will populate
  + Update the fields with the data for the selected weanling
  + Click save to update animal information

To view/update Breeders: click on the View/Update Breeders link.

* + The breeder will need to be searched prior to making changes
  + Input the breeding pair number
  + Hit the search button
  + Select the desired breeding pair to update
  + The breeding record will populate including all weanlings from selected breeding pair
  + Update the fields with the data for the selected breeders
  + Click save to update breeding information

# Report Details

The following reports will be included in the RITA

Report Breeding Pairs:

* Show all breeding pairs who are still active, and have a set update over 6 months prior to report generation

Report Weanling Report:

* Show all weanlings who will turn 21 days old within a given set of dates.

Report Survivability:

* Show survivability of pups for a strain, broken down by pair numbers.

Additional reports tbd depending on project timeline

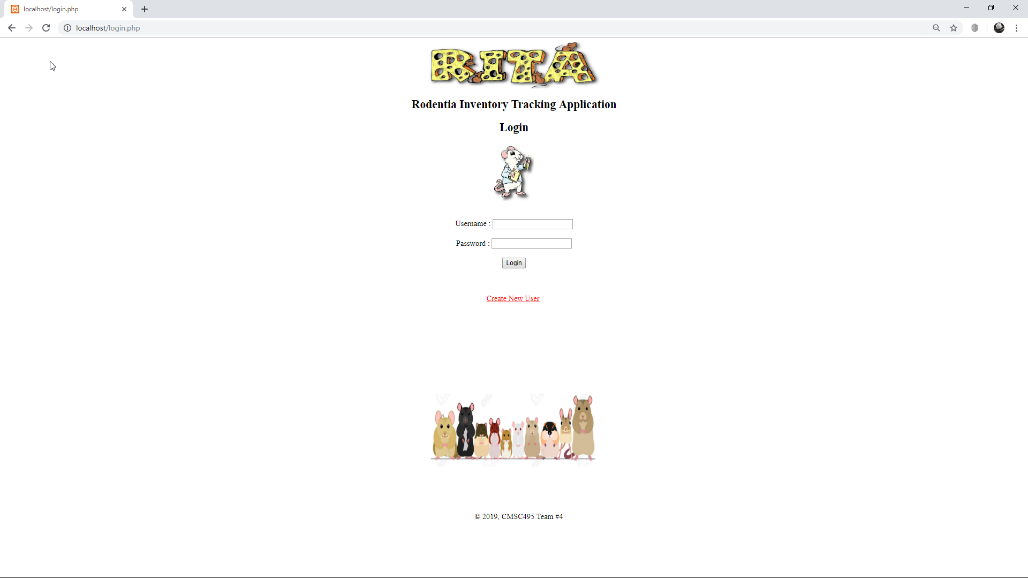
# Error Messages

|  |  |  |
| --- | --- | --- |
| Task | **Error Message** | **Resolution** |
| Access web application | “Unable to connect” | Requires internet connectivity *or*  System is down (contact admin) |
| User Creation | “Username already taken” | Enter a new username *or*  Follow procedures for resetting username and/or password. |
| User Authentication | “Invalid username or password”  “Username already taken” | Enter valid username and password *or*  Follow procedures for resetting username and/or password. |
| Breeding Pair Authentication | “Same Sex invalid Breeding Pair” | Update input record to male and female breeding pair. |
| Litter ID Authentication | “Invalid Litter ID”  “Litter ID has multiple live birth dates” | Update birth record of litter mates prior to adding to correct litter ID *or*  Create litter ID for litter with each birth date. |
| Parent ID Authentication | “Parent ID Invalid” | Confirm Parent ID and modify the necessary records |
| Print inspection | “Unable to print” | Requires available printer |
| Required Fields | “Must complete Required Fields” | All required data must be completed to save and continue. |
| View report | “Unable to generate report” | Requires internet connectivity *or*  System is down (contact admin) |
| Wean Date Authentication | “Invalid Wean Date” | Update record with correct details.   Wean date must be (x-time) after birth date |

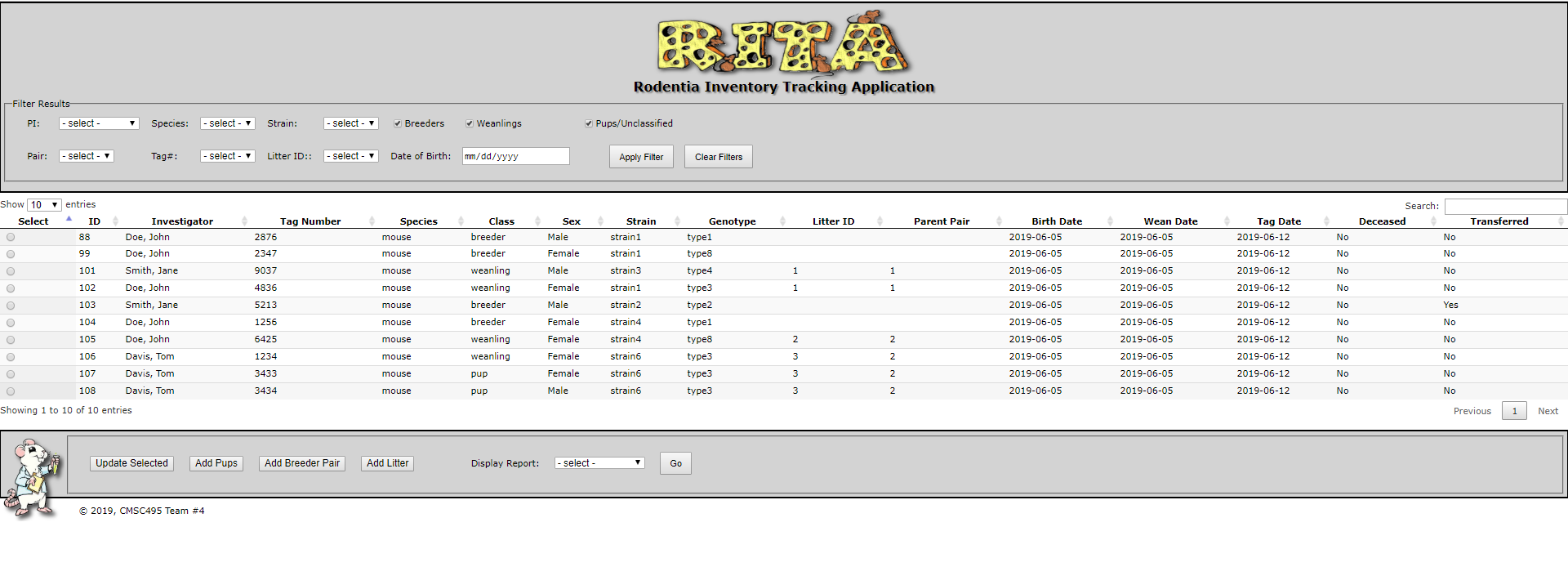
Design

# Web Application Design

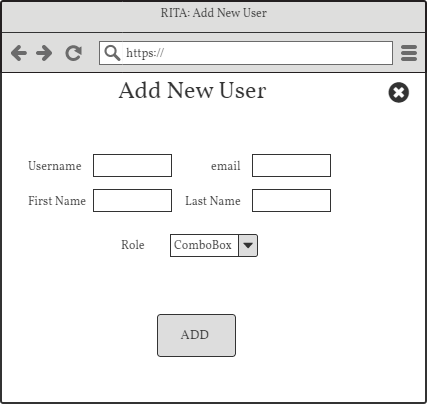
Below are wireframes for the intended user interface.



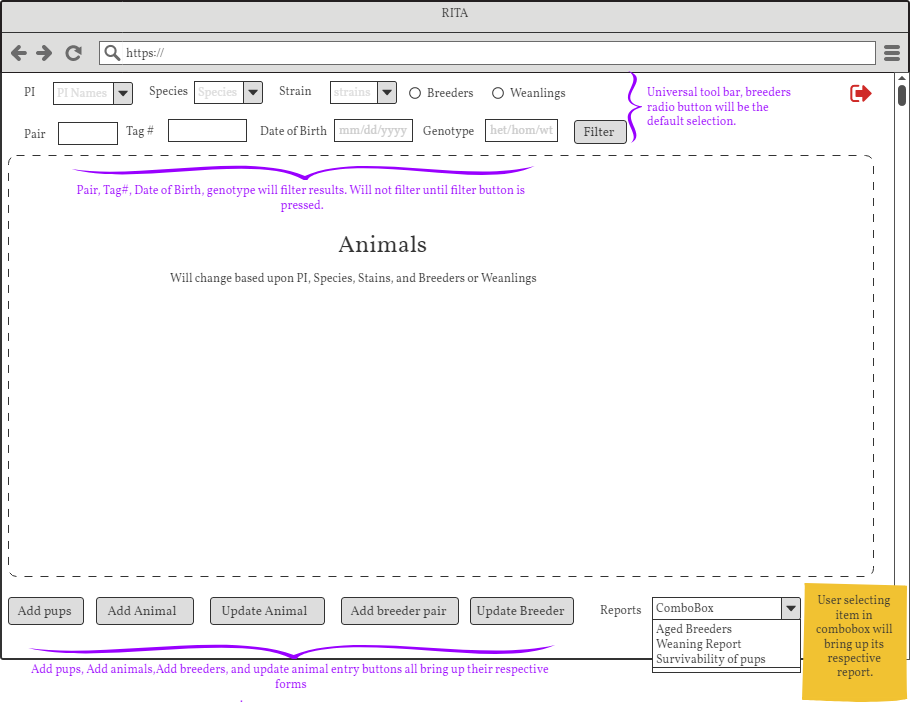
Login Screen. User enters username and password.

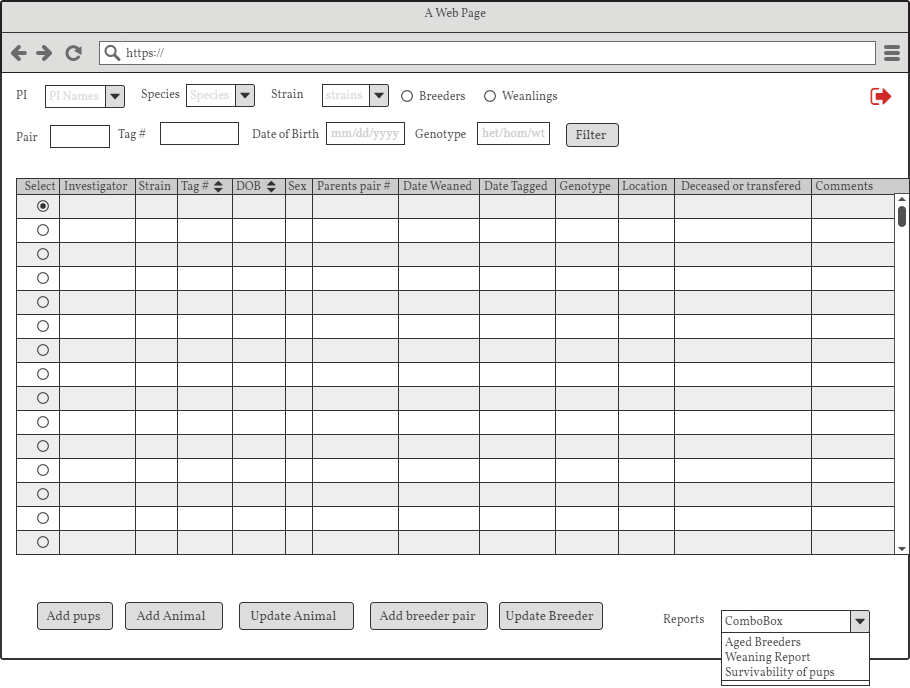


Upon successful login user is directed to Main Screen. From this screen user can filter results and generate reports.

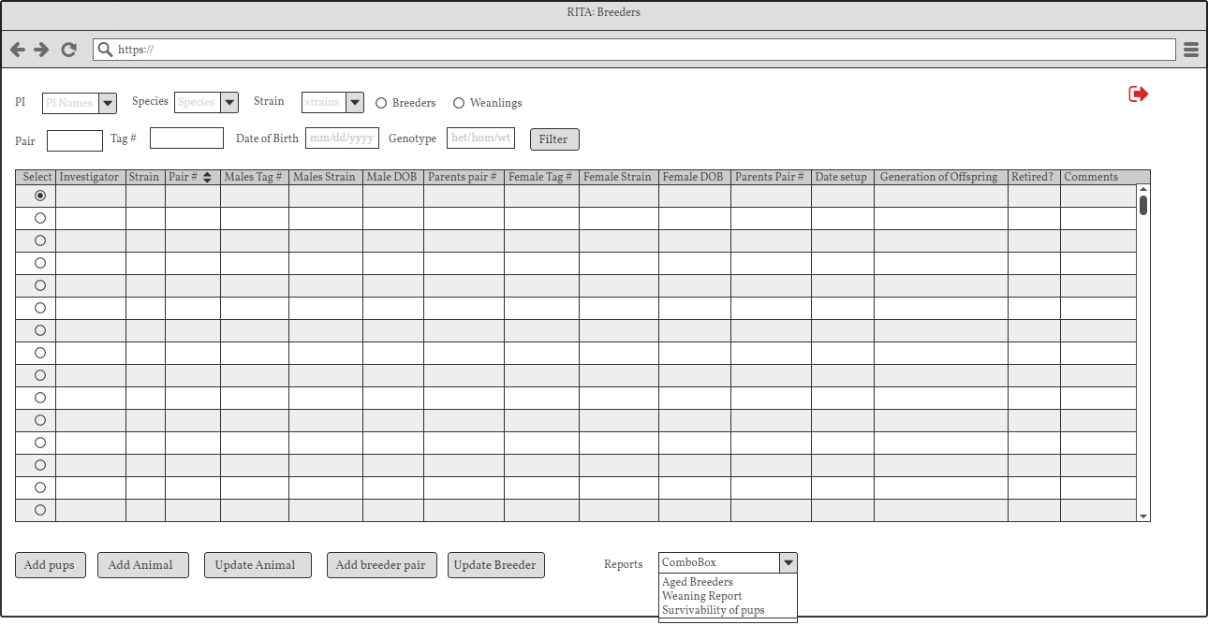
****

Add New User Screen. Once logged in as administrator, user can fill out form to add new user.

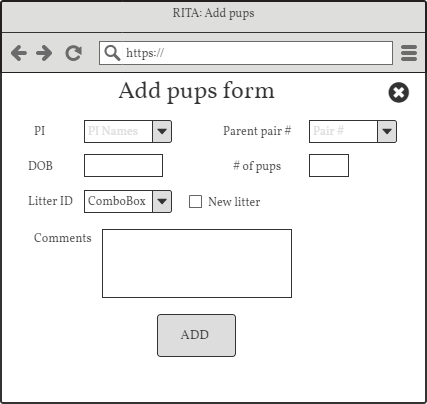
The Main Screen. Toolbar on top allows user to enter animal search criteria. Results will appear in center Animals box. Buttons on bottom allow for activities as marked. Reports dropdown will create selected report.

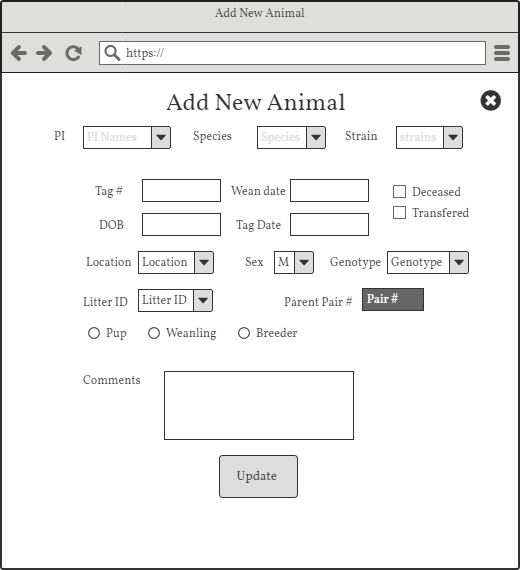


Main Screen view when Weanlings selected.

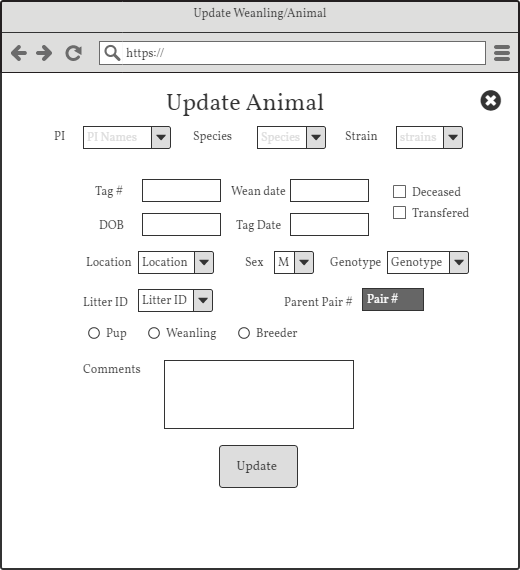


Main Screen view when Breeders selected

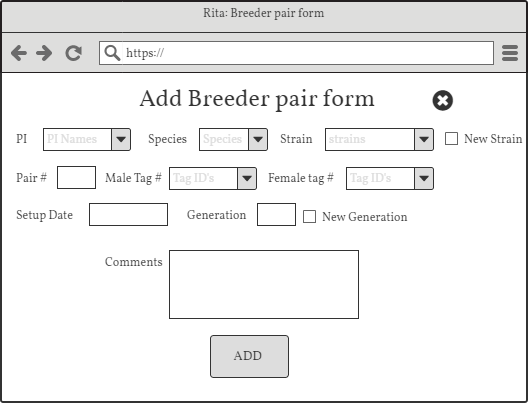
****Add Pups form. User fills out form then clicks ADD. A number of new animals equal to the # of pups are added to the database.

****Add New Animal. User fills out form then clicks ADD. One animal with the filled in fields is added to the database. Used primarily for adding animals born at another location.

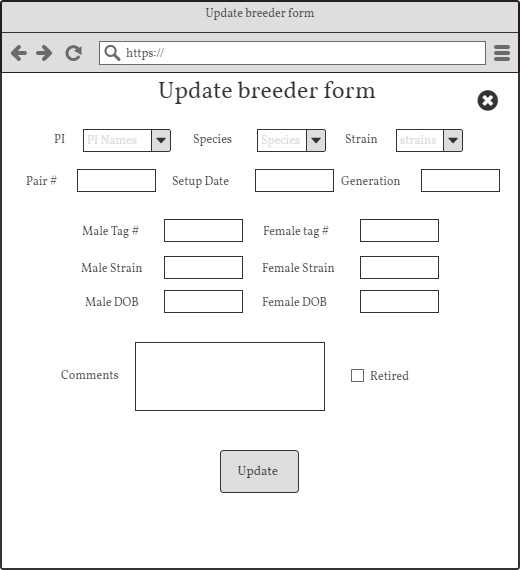
Update Animal Screen. Allows user to enter new data for an animal and update it in the database. Used for weaning, tagging, and error correction.

****

Add Breeder Pair Screen. User can create a new breeding pair. Requires a male and female of the same species. A Wild Tag will be an option for using a Wild as one of the breeding pairs. Generation is the generation of the pair’s offspring.

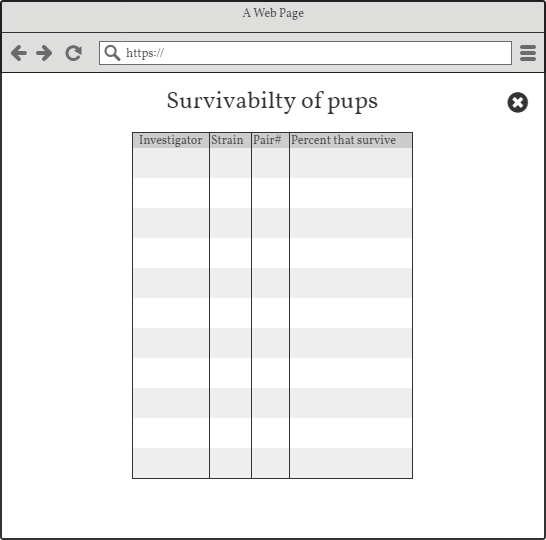
****

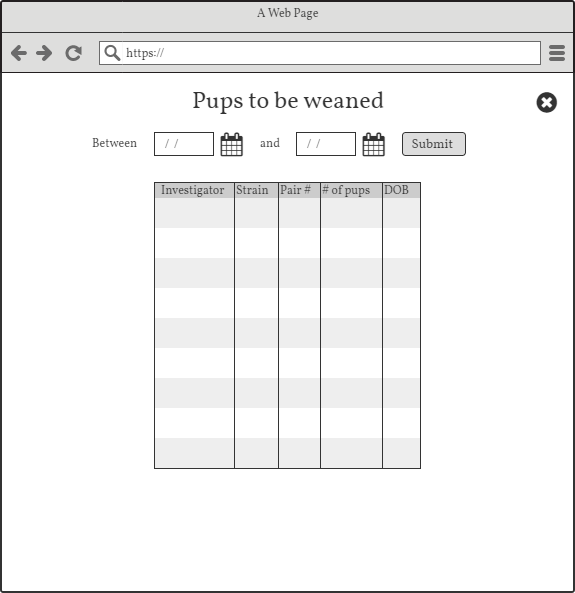
Update Breeder Screen. Similar to Update Animal Screen but has fields for updating breeding pair data.

****

****

Aged Breeder Screen. Shows report for breeders that are due to be retired.

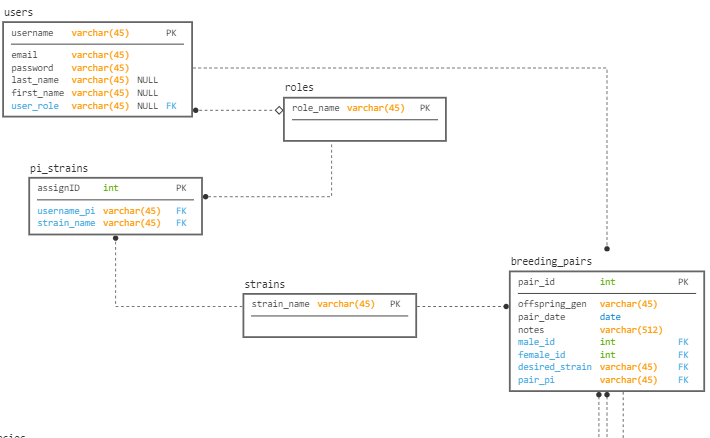
****Survivability Screen. Displays the report of percentage of pups that survive for each breeding pair.

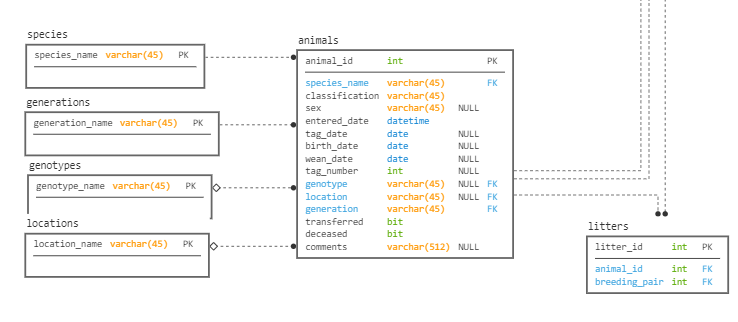
****

Weaning Report Screen. Show report for all pups born between the selected dates. Displayed pups are due to be weaned.

# Backend Database

The database for the application consists of tables for primarily users and animals. Users can have defined roles, and one of those roles, PI (Principal Investigator) has assigned strains they are approved to work on. Another table lists all the strains currently approved for any PI. Animals have a list of traits that are specific for each. Associated tables keep current lists of approved states for some of the traits to allow for simplified updates, such as a location change, or adding different species. A male and female animal (of the same species) can be bred creating a breeding pair. They are done so in an attempt to create a desired strain in their offspring. Each breeding pair can have multiple litters of offspring every year, so a litter id is assigned to the pups (a classification of Animals) for differentiating amongst separate litters. The database is a MySQL database utilizing MyWebSQL ver 3.7 for management. ERD for the database is below.





The ERD for the backend database for R.I.T.A

Phase 1

# Overview

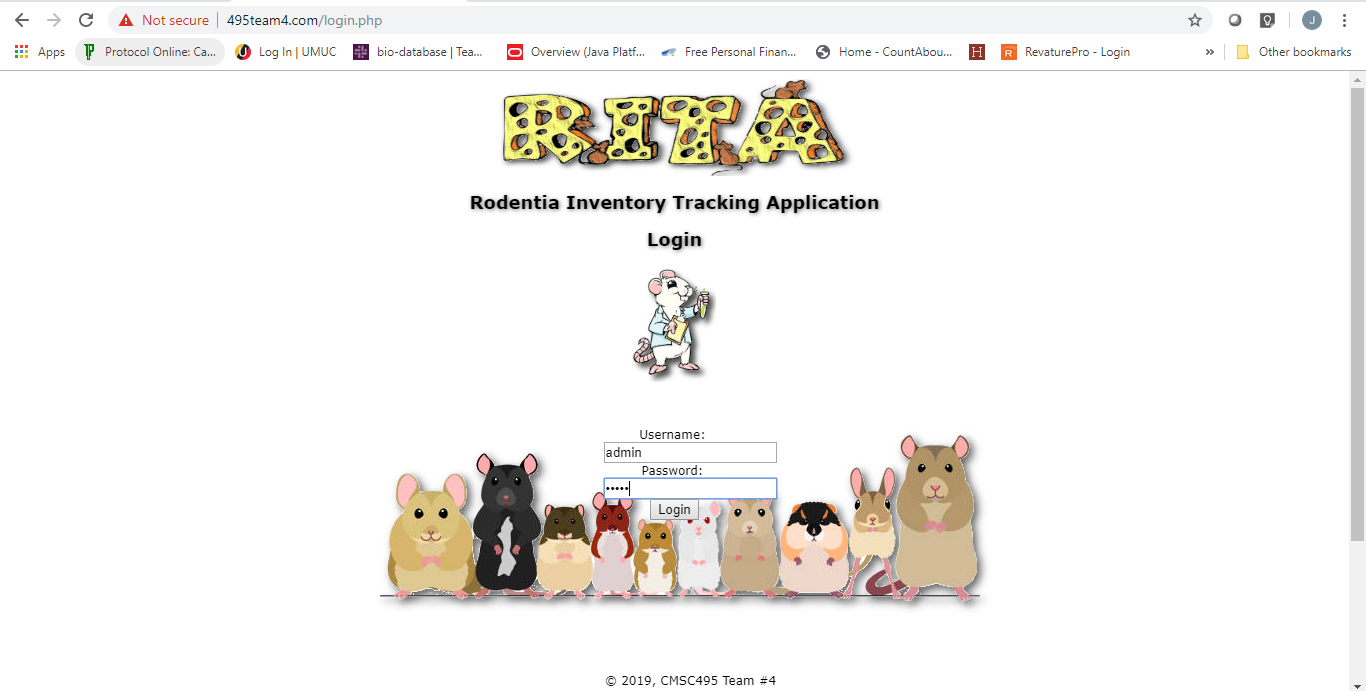
The primary goal of Phase I was to get user login and registration functioning. To accomplish this task a login page was created along with a user registration page and a main page to be directed to after successful login. The login page accepts a user name and password and upon successful validation of the entered credentials, the user is redirected to the main page. The main page consists of a title, filter form, table view, and action buttons. It also has some of the required functionality for future development phases. In the upper righthand corner of the main page is a dropdown menu where all users can select to logout which ends the session and brings them back to the login page, or if they have administrative access they can choose to create a new user. Create new user brings the administrator to the registration page where they can enter a new user’s data for addition to the database. From there, they can continue adding users, return to the main page, or logout.

# Testing

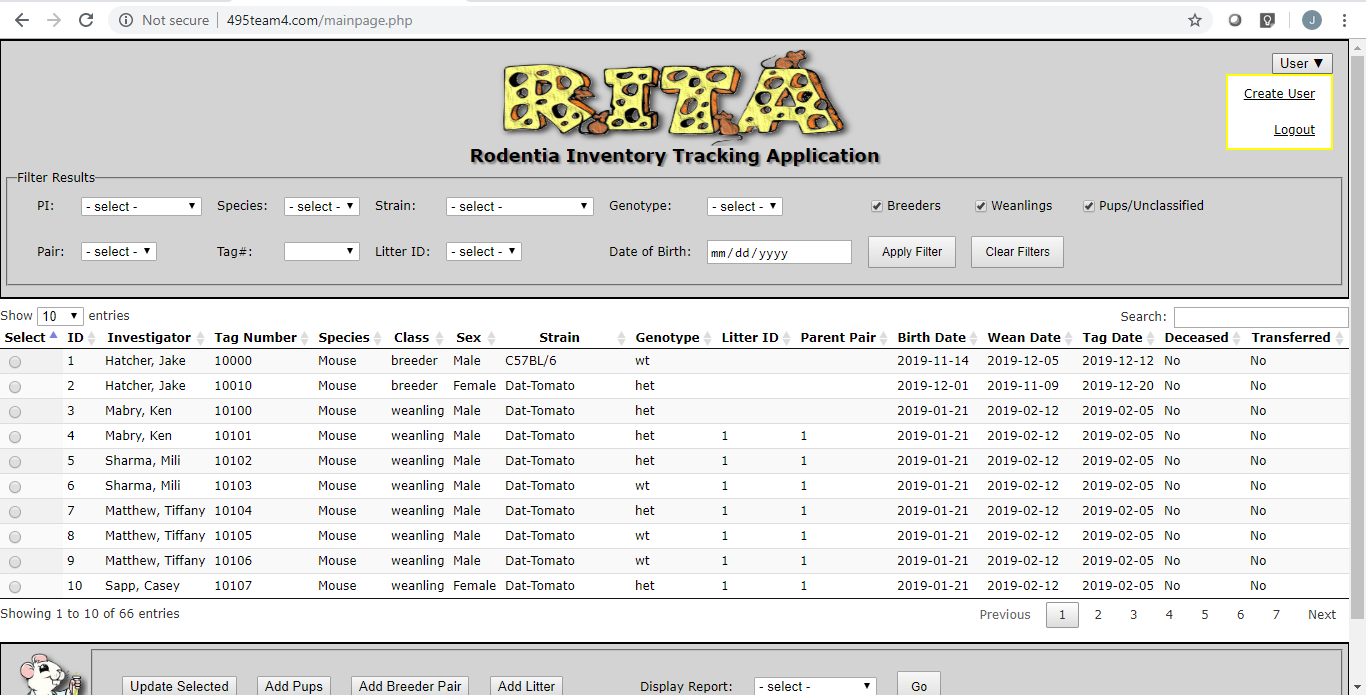
Tested using Google Chrome Version 75.0.3770.100 64-bit on Windows 10 machine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Activity | Action | Expected Result | Actual Result | Pass? |
| User Login/Registration |  |  |  |  |
| Existing Administrator logs in | 1)Enter *admin* for username  2)Enter *admin* for password  3)Click **login** | User successfully logs in and is taken to the main page  495team4.com/ mainpage.php | User successfully logs in and is taken to the main page  495team4.com/ mainpage.php | Yes |
| Navigate to registration page | 1)Go to **user** dropdown menu and click **create user** | User is taken to registration page  495team4.com/ register.php | User is taken to registration page  495team4.com/ register.php | Yes |
| Create new user | 1)Enter values username:*newuser* password:p@ss1  Firstname: new  Lastname: user  Email:nuser@rita.org  Role: Breeder Tech  2)Click **Register** | User is added to the database table *user* | User is added to the database table *user* | Yes |
| Logout | 1)Go to **user** dropdown and click **logout** | Session terminates and user brought to login screen 495team4.com/ login.php | Session terminates and user brought to login screen 495team4.com/ login.php | Yes |
| Existing non-Admin user logs in | 1)Enter *newuser* for username  2)Enter *p@ss1* for password  3)Click **login** | User successfully logs in and is taken to the main page  495team4.com/ mainpage.php | User successfully logs in and is taken to the main page  495team4.com/ mainpage.php | Yes |
| Navigate to registration page | 1)Go to **user** dropdown menu and click **create user** | **Create user** option not listed | **Create user** option not listed | Yes |
| Logout | 1)Go to **user** dropdown and click **logout** | Session terminates and user brought to login screen 495team4.com/ login.php | Session terminates and user brought to login screen 495team4.com/ login.php | Yes |
| Existing user enters incorrect password | 1)Enter *newuser* for username  2)Enter *pass1* for password  3)Click **login** | User stays on login page and error message *‘Username/Password invalid’* displays | User stays on login page and error message *‘Username/Password invalid’* displays | Yes |
| Non-existing user attempts login | 1)Enter *nonuser* for username  2)Enter *24pass* for password  3)Click **login** | User stays on login page and error message *‘username not found’* displays | User stays on login page and error message *‘username not found’* displays | Yes |
| Navigate directly to mainpage without log in | 1)Enter into browser’s address bar: 495team4.com/ mainpage.php (remove spaces) | User is redirected to login page | User is redirected to login page |  |
| Navigate directly to register page | 1)Enter into browser’s address bar: 495team4.com/ register.php (remove spaces) | User is redirected to login page | User is redirected to login page | Yes |

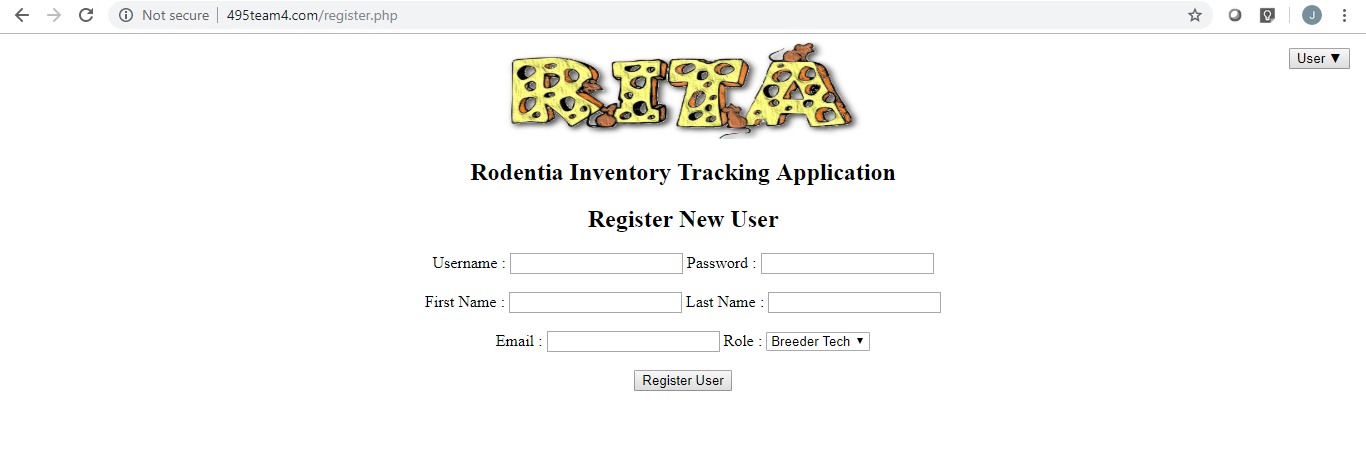
# Screenshots



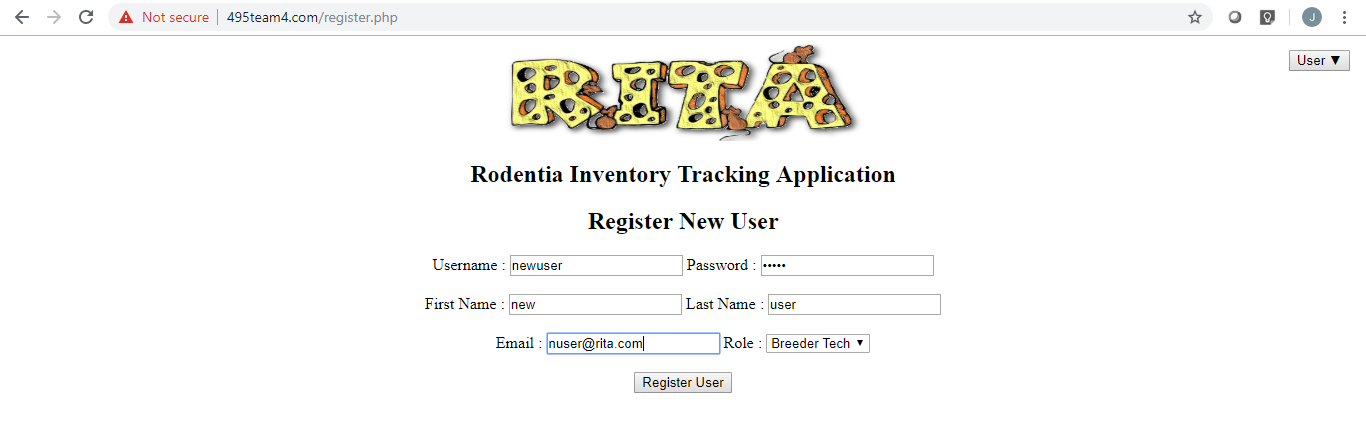
Admin login



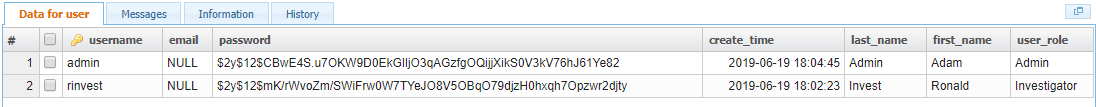
Successful admin login



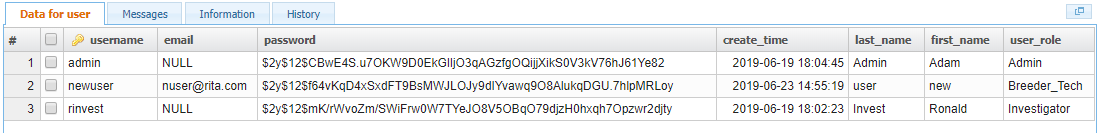
Successful navigation to Register page



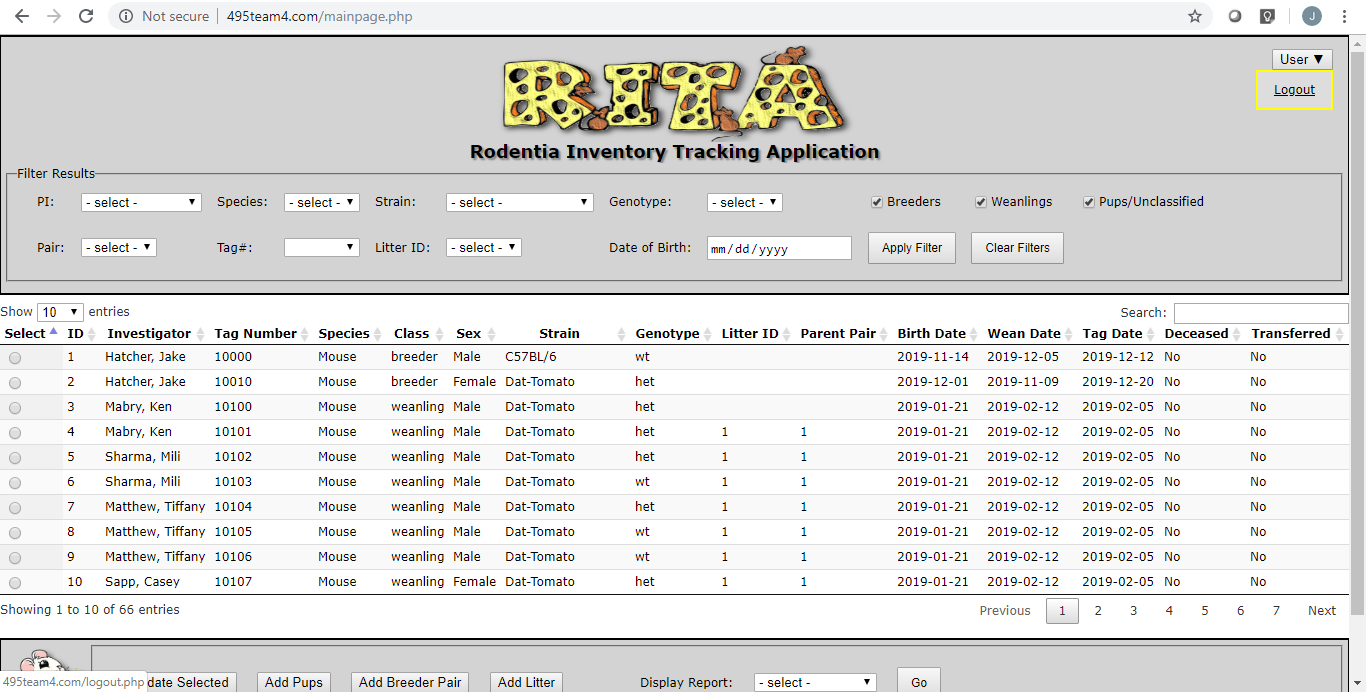
New user registered values



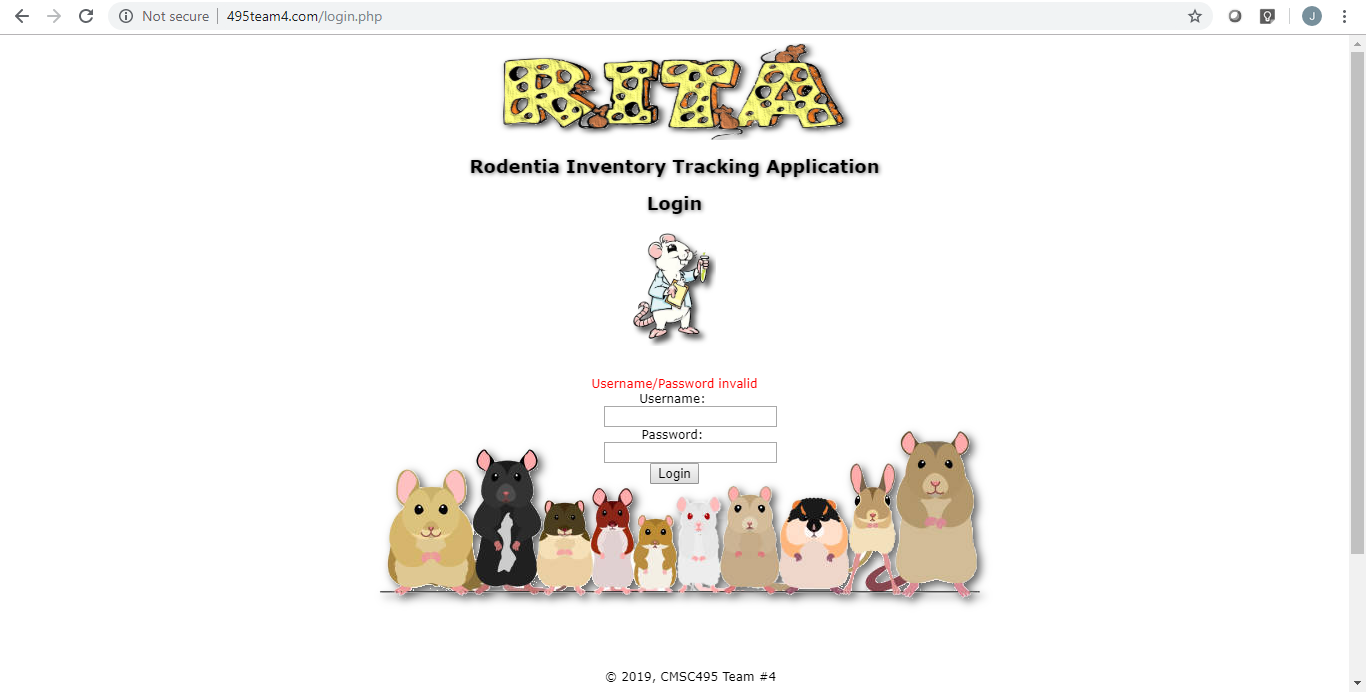
Database *user* table before register



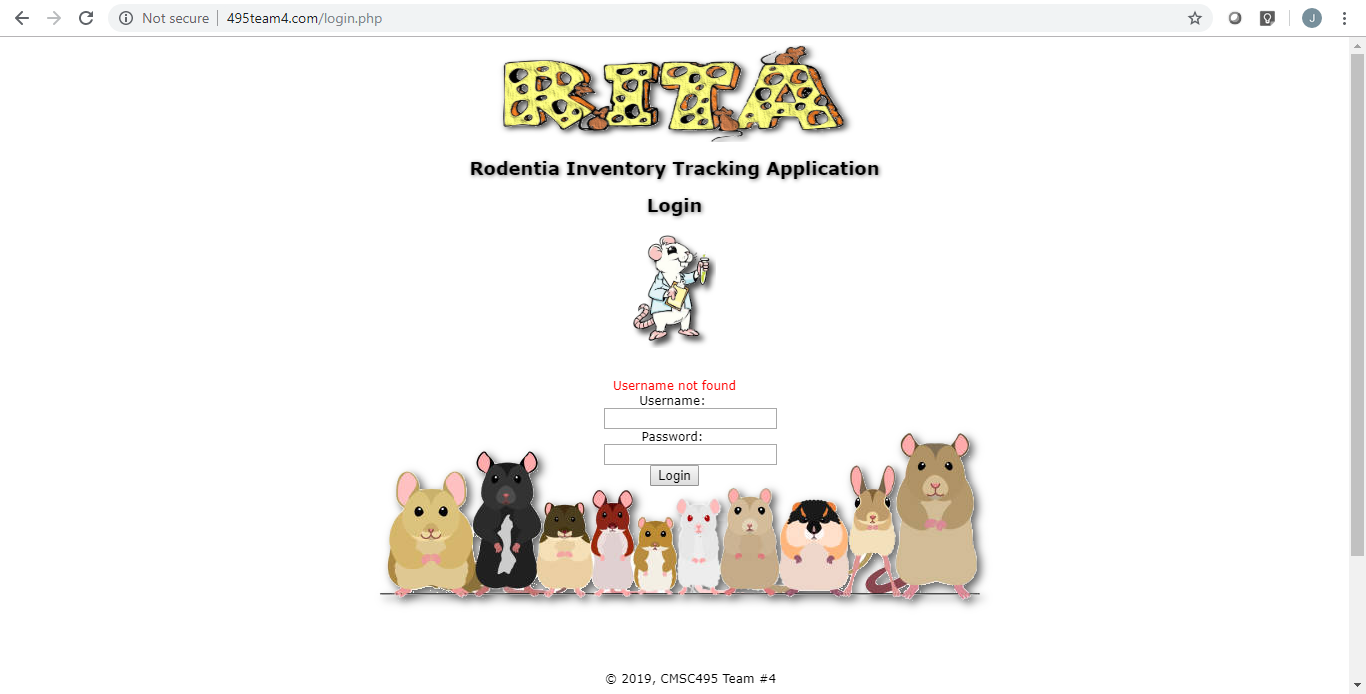
Database *user* table after register



Successful breeder tech login / no **create user** option



Login page after existing username but incorrect password



Login page after non-existent username provided

# Next Steps

Phase II of the development process will begin on 6/24/19 and will be focused on being able to add new animals to the database and update existing ones. Secondary goals include further development of the main page’s table view and adding additional administrator tasks.

References

*Test Plan (a Real Sample).* (2014) Retrieved from

<https://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2014/02/Live_Project_Test_Plan_SoftwareTestingHelp.pdf>

*Test Plan Template: Sample Document with Web Application Example.* (n.d.) Retrieved from

<https://www.guru99.com/test-plan-for-project.html>.

Team Contributions

|  |  |  |  |
| --- | --- | --- | --- |
| Phase I  Week 5 | Creators | Reviewers | Comments |
| Create Database | Ken Mabry  Casey Sapp | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| User Login Screen | Jake Hatcher  Mili Sharma | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| Create New User | Jake Hatcher | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| Update Database | Tiffany Mathew | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| Coding and implementation of web design | Casey Sapp | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| Coding and implementation of database | Ken Mabry | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |
| Documentation Updates | Mili Sharma | Jake Hatcher, Ken Mabry,  Tiffany Mathew, Casey Sapp,  Mili Sharma |  |