**Individual Assignment Specification**

**Team Mickey**

**Rakshit Choudhary:**

**Devise:**

* **Task Description**: Sign in into the system Sign up and Sign out from the system by the user.
* **Outcome**: Used devise gem to build the user session and to authenticate user.

**Table Associations:**

* **Task description**: Associations from User table to the cage and the mouse tables as per the requirements.
* **Outcome**: Used rails migrations to add references from user to cage and user to mouse.

**Seed Data:**

* **Task Description**: Seeding of valid data to check the functionality of the database.
* **Outcome**: The data that needs to be added to the database on deployment which function accordingly. Added data on seeds.rb file for user table and its associations as well.

**Functionalities**:

**Sign In:**

* **Task description**- The user will be provided with a webpage which would be used for him to login into the website. Whenever he enters his credentials after logging in he would be directed to his webpage which contains information about all the mice’s he has and also can make some changes as required.
* **Outcome**-Since we are using a built in function, the tests should work. The user id and password should be valid.

**Sign Out:**

* **Task description**- The user would be able to complete his session and return back to the login page after the completion of his tasks.
* **Outcome**- Since we are using a built in function, the tests should work. Once the user hits the logout button, s/he should be successfully logged out and re-directed back to the home page for login.

**Edit User:**

* **Task description**- The user would be able to edit his account related information by clicking on the ‘Edit Account’ link on the user home page.
* **Outcome**- Once the user hits the ‘Edit Account’ link, the user shall be redirected to a page where he can change his/her ‘Username’ and ‘Password’. Any of these changes made will be valid and accepted only if the user confirms it with the current password of the user.

**Sign Up:**

* **Task description**- The user should be able to register to the website. In the homepage the user should be able to see a ‘Sign Up’ button which would redirect him to the registration page. It would consist of the fields asking him for a desired email address, username and a password.
* **Outcome** - Since we are using a built in function, the tests should work. Once the user hits the “Sign Up” button, the user shall be redirected to the registration page.

**Team Contribution**: Played the role of TEAM-LEADER and managed all the tasks and distributed the work equally to other members. Devise and use table associations with the cage and the mouse table and its respective seeding was handled and executed successfully. Is also responsible for updating the Demo Video and the Individual Assignment Specification file to the database.

**Faisal Subaei:**

**Transfer Table:**

* **Task Description**: A transfer table for requesting a transfer of mouse from one cage to another cage as per the requirements.
* **Outcome**: Used scaffold to create transfer table with cage id(that the mouse has to be transferred to) and mouse id(which is a link that leads to the mouse web page, so that the mouse can be identified easily) as attributes.

**Table Associations:**

* **Task description**: Added associations from transfer table to the user table and mouse table.
* **Outcome**: Used rails migrations to add references from transfer table to cage and to the mouse.

**Seed Data**:

* **Task Description**: Seeding of valid data to check the functionality of the database.
* **Outcome**: The data that needs to be added to the database on deployment which function accordingly. Added data on seeds.rb file for transfer table and its associations as well.

**Functionalities:**

**Adding a transfer**

* **Task description**: Here the user will be able to request for a transfer of the mouse from one cage to another cage based on his requirements and he also will be able to transfer mice from the parent cage to the child cage(Once the child is more than 21 days old)
* **Outcome:**  Once the user clicks on the “New Transfer” button, the user shall be asked for the “To-Cage id” and the “mouse id” which will make its way as an entry to the transfer table. Once the admin has approved the transfer, the mouse info shall be edited accordingly.(The admins approval functionality shall be implemented in the iteration 2)

**Team Contribution**: Making of the demo video for the purpose of the functionality of the project and its working orderly.

**Suhaib Rousan:**

**Mouse Table**

* **Task Description**: A transfer table for storing mouse data which includes its “Tag Number”, “Genome”, “Parent(Mother) Tag Id”, “Sex”, “Age” and “Date of Birth”
* **Outcome**: Used scaffold to create mouse table with all the above mentioned attributes, for easy access of mouse related data. The user shall be able to store mouse related information for her/is experiments.

**Seed Data:**

* **Task Description**: Seeding of valid data to check the functionality of the database.
* **Outcome**: The data that needs to be added to the database on deployment which function accordingly. Added data on seeds.rb file for mice table and its associations as well.

**Functionalities:**

**Add a mouse-**

* **Task description**- The valid user would be able to make changes in his page i.e. he can add different mice when s/he wants. When adding s/he would also be getting the complete information of the particular mouse/mice he is adding.
* **Outcome**- In the user’s homepage and under the “List all mice” page s/he would be able to see a button which says ‘Add mouse’. After s/he is redirected he would be able to add a particular mouse/mice to the desired cage.

**Edit mouse Info:**

* **Task description**- The user would be able to view information related to a particular mouse/mice. All the details of a particular mouse/mice would be displayed to the user. When the user clicks on “Edit Mouse” s/he shall be prompted with a screen to make changes to the related information.
* **Outcome**- When the user clicks the ‘ View’ button s/he will be able to get all the information related to the mouse. When the user clicks the ‘Edit’ button s/he will be able to edit all the information related to the mouse. Based on this s/he can decide as to what has to be done with that particular mouse.

**Delete mouse**:

* **Task description**- If the user has to delete a mouse s/he should be able to do so by clicking on the ‘Destroy’ button.
* **Outcome**- Once the user clicks the ‘Destroy’ button, s/he will be prompted with a pop-up asking for confirmation. Once the user confirms the operation, the requested mouse shall be successfully deleted from the cage and will not appear under the user anymore.

**Team Contribution**: Is the QA for the team. Takes care of all the promised functionality and checks if there are any issues and notifies the user upfront.

**Adithya K Murthy:**

**Cage Table:**

* **Task Description**: A cage table with attributes such as ‘Cage Name’ and ‘User ID’ of the user that the cage belongs to.
* **Outcome**: Used scaffold to create cage table with all the above mentioned attributes, for easy access of cage related data.

**Cage Associations:**

* **Task description**: Added associations from cage table to the mouse table.
* **Outcome**: Used rails migrations to add references from cage table to the mouse table.

**Seed Data:**

* **Task Description**: Seeding of valid data to check the functionality of the database.
* **Outcome**: The data that needs to be added to the database on deployment which function accordingly. Added data on seeds.rb file for cage table and its associations as well.

**Functionalities:**

**Add a cage:**

* **Task Description**: A valid user will be able to add a new cage under his/her name.
* **Outcome**: Once the user clicks on the ‘New Cage’ button on the User Home Page or on the Cages Page, the user shall be redirected to a webpage asking him to enter the above mentioned attributes in order to create a cage.

**Edit cage**:

* **Task description**- If the user wants to edit the cage name, s/he shall be able to do so by clicking on the ‘Edit’ button.
* **Outcome**-Once the user clicks on ‘Edit’ on the Cages Page, the user shall be redirected to a page where he will be prompted to change the cage name. Once the name is changed and hits on ‘Update’, the name gets updated and the same shows up on the Cages Page.

**Delete cage**-

* **Task description**- If the user wants to delete the cage, s/he shall be able to do so by clicking on the ‘Destroy’ button.
* **Outcome**-Once the user clicks on ‘Destroy’ on the Cages Page, the user will be asked for confirmation by a pop-up window, once the confirmation is given the cage gets destroyed and the same shows up on the Cages Page.

**Team Contribution**: Demo booth operator for the iteration 1 and Individual Assignment specification document.