**Planetarium RTM**

**User Stories**

* US1
  + As a new user I want to open an account with the Planetarium so I can save my celestial findings
* US2
  + As a user I want to securely access my account so I can interact with the Planetarium in a secure environment
* US3
  + As a user I want to see my planets and moons added to the Planetarium so I can track my findings
* US4
  + As a user I want to add new planets and moons to the Planetarium so I can update my findings
* US5
  + As a user I want to remove planets and moons from the Planetarium so I can correct my findings

**Story Requirements**

* **Security Requirements**
  + SR1
    - Passwords should never be visible in plain text
  + SR2
    - Users should only be able to interact with resources they have added to the Planetarium
  + SR3
    - Only logged in Users should be able to access the Planetarium home page
* **User Requirements**
  + UR1
    - Users should have unique usernames
  + UR2
    - Usernames and passwords should be between 5-30 characters
  + UR3
    - Usernames and passwords should start with a letter
  + UR4
    - Usernames should support lower case, upper case, numeric characters, underscores, and dashes
  + UR5
    - Passwords should contain a lower case, upper case, and numeric character
  + UR6
    - Passwords should support underscores and dashes
* **Planet Requirements**
  + PR1
    - Planet names should be between 1-30 characters
  + PR2
    - Planet names should support lower case, upper case, numeric characters, white space, underscores, and dashes
  + PR3
    - Planets should have unique names
  + PR4
    - Planets should be “owned” by the user that added it to the Planetarium
  + PR5
    - Planets should allow adding an associated image, but an image should not be required for the data to be added to the database
  + PR6
    - JPEG & PNG should be supported
* **Moon Requirements**
  + MR1
    - Moon names should be between 1-30 characters
  + MR2
    - Moon names should support lower case, upper case, numeric characters, white space, underscores, and dashes
  + MR3
    - Moons should have unique names
  + MR4
    - Moons should be “owned” by the Planet the User adding the moon associated it with
  + MR5
    - Moons should be deleted if their "owner" planet is deleted from the database
  + MR6
    - Moons should allow adding an associated image, but an image should not be required for the data to be added to the database
  + MR7
    - JPEG & PNG should be supported
* **Requirement Mapping**
  + **US1**
    - **SR1**
    - **SR3**
    - **UR1**
    - **UR2**
    - **UR3**
    - **UR4**
    - **UR5**
    - **UR6**
  + **US2**
    - **SR1**
    - **SR2**
    - **UR1**
    - **UR5**
    - **UR6**
    - **PR4**
    - **MR4**
    - **MR5**
  + **US3**
    - **SR2**
    - **SR3**
    - **UR1**
    - **PR3**
    - **PR4**
    - **PR5**
    - **PR6**
    - **MR3**
    - **MR4**
    - **MR5**
    - **MR6**
    - **MR7**
  + **US4**
    - **SR2**
    - **SR3**
    - **UR1**
    - **PR1**
    - **PR2**
    - **PR3**
    - **PR4**
    - **PR5**
    - **PR6**
    - **MR1**
    - **MR2**
    - **MR3**
    - **MR4**
    - **MR5**
    - **MR6**
    - **MR7**
  + **US5**
    - **SR2**
    - **SR3**
    - **UR1**
    - **PR3**
    - **PR4**
    - **MR3**
    - **MR4**
    - **MR5**
* **Acceptance Criteria**
  + **US1**
    - **Happy Path**
      * **Given** the user is on the login page
      * **When** the user clicks the register link
      * **When** the user provides a valid username
      * **And** the user provides a valid password
      * **Then** an alert should appear saying “Account created successfully”
      * **And** the user should be redirected to the login page
      * \*Note\*: this style of specific AC is closer to what you will use for your tests, but for some user stories being specific with your AC will require creating multiple groupings of AC, which can be time consuming
    - **Sad Path**
      * **Given** the user is on the login page
      * **When** the user clicks the register link
      * **And** the user provides invalid data
      * **Then** the user should be shown an alert with an error message
      * **And** the user should remain on the register page
      * \*Note\*: This broader form of acceptance criteria is fine, just keep in mind for your actual tests you will need to break it down into more atomic steps
  + **US2**
    - **Happy Path**
      * **Given** the user is on the login page
      * **When** the user provides a matching username and password
      * **And** the username and password are case-sensitive to what exists in the DB
      * **Then** an alert should appear saying “Login successful”
      * **And** the user should be redirected to their Planetarium page
    - **Sad Path**
      * **Given** the user is on the login page
      * **When** the user provides invalid data
      * **Then** the user should be shown an alert with an error message
      * **And** the user should remain on the register page
  + **US3**
    - **Happy Path**
      * **Given** the user is logged in
      * **And** the user has added planets/moons
      * **When** the user is accessing their planetarium page
      * **Then** their planets/moons should be displayed in the UI
    - **Sad Path**
      * **Given** the user is logged in
      * **And** the user has added planets/moons
      * **When** the user is accessing their planetarium page
      * **Then** the planets/moons will not display
  + **US4**
    - **Happy Path**
      * **Given** the user is logged in
      * **And** the user is on the planetarium page
      * **When** the user provides valid data in the name field
      * **Then** a planet/moon should appear in the list.
    - **Sad Path**
      * **Given** the user is logged in
      * **And** the user is on the planetarium page
      * **When** the user provides invalid data in the name field
      * **And** provides a non JPEG/PNG image file in the image upload
      * **Then** a planet/moon will not appear in the list
      * **And** the user should be shown an alert saying “Invalid data!”
  + **US5**
    - **Happy Path**
      * **Given** the user is logged in
      * **And** the user is on the planetarium page
      * **When** the user provides an existing planet/moon name in the text field
      * **And** the user clicks the delete button
      * **Then** the existing planet/moon should be deleted from the planetarium
    - **Sad** **Path**
      * **Given** the user is logged in
      * **And** the user is on the planetarium page
      * **When** the user provides invalid data
      * **And** the user clicks the delete button
      * **Then** the user should be shown an alert saying “Planet/Moon does not exist!”
* **Test Data**
  + **US1**
    - **Valid Test Data**
      * Robin
        + Unique username & start w/ letter
      * Bobby
        + 5 characters & start w/ letter
      * Thisshouldbethirtycharactersss
        + 30 characters & start w/ letter
      * Robin\_1-5
        + Supports upper, lower, numeric, dash, underscore
    - **Invalid Test Data**
      * Batman
        + Non-unique username
      * Bobb
        + 4 characters
      * Thisoverthirtycharacterssssssss
        + 31 characters
      * Robin!%?
        + Includes unsupported characters
    - **Valid Password Test Data**
      * Bobb1
        + 5 char w/ required characters
      * Thisshouldbethirtycharacterss3
        + 30 char w/ required characters
      * Bob\_b-1
        + Supports dash & underscore
    - **Invalid Password Test Data**
      * Bob3
        + 4 char pass
      * Thisoverthirtycharactersssssss3
        + 31 char pass
      * Bobby/Bobby/bobb3/BOBB3
        + Not all req charr present
      * #\*(&#(
        + Unsupported characters
  + **US4**
    - **Valid Planet/Moon Test Data**
      * A
        + 1 char w/ letter
      * Planetnamewith30charactersssss
        + 30 characters
      * Ab1 \_-
        + All supported characters
      * Planet.jpeg
        + Supported image type
    - **Invalid Planet/Moon Test Data**
      * Empty text field
        + 0 character name
      * Planetnamewith31characterssssss
        + 31 characters
      * $^&\*
        + Unsupported characters
      * Planet.mp3
        + Unsupported file type
  + **US5**
    - **Valid Planet/Moon Test Data**
      * Existing planet/moon name
        + Planet/moon that can be deleted
    - **Invalid Planet/Moon Test Data**
      * Non-existing planet/moon name
        + Planet/moon that does not exist
* **Defect Report**
  + **Registration**
    - Alert did not appear for “Invalid username” when an existing one was used for registration
      * Low Severity
        + Not app-breaking, but users should know why their username is invalid, especially when it’s within the requirements
      * High Priority
  + **Add**
    - Alert did not appear for “Invalid file type” when uploading a non JPEG/PNG file.
      * High Severity
        + If compromised, users can upload malicious files that are unsupported, such as script files
      * High Priority
    - Form does not clear upon input
      * Low Severity
        + Duplicate inputs are prevented, so potential multiple requests being made are blocked. However, this creates an unnecessary
      * Low Priority
  + **Delete**
    - Alert did not appear for “Invalid moon name” when attempting to delete a non-existent moon
      * Low Severity
      * High Priority
    - Form does not clear upon input
      * Low Severity
      * Low Priority
    - If not fully scrolled up, button will bring you to the top of the page, then you are required to press it again to actually delete.
      * Low Severity
      * High Priority
* **Acceptance Testing Reporting**
  + **US1**
    - **2 – OK**
    - Registering is fairly straightforward and functional, however, the requirements are not explained to the user, which created further confusion when an existing username was attempted with no proper explanation from the alert. The Login link is so tiny it can be very easily missed. It should be a button that is slightly smaller than the input fields, or a slightly larger link than it already is, but centered.
  + **US2**
    - **3 – Good**
    - Logging in is slightly better than registration, as this is where details need to be obscured. Informing the user whether or not the username or password isn’t matching could mean social engineers having an easier time compromising information. Redirection to the planetarium is quick and snappy, with no alert in the way.
  + **US3**
    - **2 - OK**
    - Viewing the planetarium is very straightforward, once you log in, you are instantly brought to the planetarium. However, it is mediocre aesthetically, the font choice is poor, and the table is mostly unstyled outside of global stylings. The text inputs having default stylings are also off-putting.
  + **US4**
    - **1 – Bad**
    - Adding planets and moons is functional, and that is the best thing that can be said about it. Changing the entire addition form when you choose between the type is not just an automation nightmare, but an accessibility one as well. The inputs not clearing after submission also add to this, creating unnecessary work of individually clearing inputs for both the tester and the end user.
  + **US5**
    - **2 - OK**
    - Removing planets is much more functional. You simply add a name, and it deletes, but not when it doesn’t exist. However, an alert not appearing for the latter is poor user experience. Like the add feature, it does not clear upon input either, and has a weird quirk where it scrolls up if you click it and aren’t fully scrolled up already. It won’t delete if this happens, either, you’ll need to click it again.