Selected Issues for Sprint 2

Frontend_003 - Home Page
Assigned Team Member: Cole Logan
Priority Level: High
Acceptance Criteria: The main page in the system. This page must give access to all other parts of the system.

Frontend_004 - Viewing Equipment Database
Assigned Team Member: Cole Logan
Priority Level: High
Acceptance Criteria: The equipment list from the database. This page must get all equipment details from the database and display them for users to see.

Frontend_005 - Account Page
Assigned Team Member: Cole Logan
Priority Level: Medium
Acceptance Criteria: Where users can view their account information. This page must get user information from the database and display them for users to see.

Backend_001 - User Registration
Assigned Team Member: Shaylin Smith
Priority Level: High
Acceptance Criteria: A backend Flask route to go along with the existing Register page. The route must be able to communicate with the Register page in order to allow users to create new accounts.

Backend_002 - User Verification
Assigned Team Member: Shaylin Smith
Priority Level: High
Acceptance Criteria: Backend functionality to verify users upon login.

Backend_003 - Login Functionality
Assigned Team Member: Shaylin Smith
Priority Level: High
Acceptance Criteria: A backend Flask route to go along with the existing Login page. The route must be able to communicate with the Login page in order to authenticate users.

Backend_004 - Logout Functionality
Assigned Team Member: Cole Logan
Priority Level: Medium
Acceptance Criteria: A method for users to log out of their accounts. There must be a button that can be used to log users out of their session.

Backend_005 - View Equipment Details
Assigned Team Member: Asa Fischer
Priority Level: Medium
Acceptance Criteria: A way for users to view a specific piece of equipment's details. In the equipment list created early in the sprint, there must be a button to view each individual piece of equipment's details.

Backend_007 - Reserve Equipment
Assigned Team Member: Shaylin Smith
Priority Level: High
Acceptance Criteria: A backend Flask method to allow users to reserve equipment. The route must be able to get information from the user and create a reservation which it will enter into the database.

Backend_008 - Cancel Reservations
Assigned Team Member: Johnjacobjames Erdman
Priority Level: High
Acceptance Criteria: Backend functionality to cancel user reservations. Users should be able to manage their reservations, including the ability to cancel any reservations.

Backend_009 - Usage Logs
Assigned Team Member: Johnjacobjames Erdman
Priority Level: Medium
Acceptance Criteria: Backend functionality to track and log reservation information. Admins should be able to access this for later analysis.

Backend_013 - User Profile Management
Assigned Team Member: Cole Logan
Priority Level: Medium
Acceptance Criteria: Users must be able to update their account information. In the user account page, there must be an edit button that users can use to change account information.