**The Resource Alignment Group**

# MAFES Equipment Management System

## System Requirements Specification

**Version 1.0**

**October 20, 2025**


**Project Team**

Bradan Craig

Drew Marecek

McKade Wing

Theodore Morin

**MAFES Customer Representatives**

Tyler Messerschmidt

Lee Hecker

**The Resource Alignment Group**

# MAFES Equipment Management System

## System Requirements Specification

### Table of Contents

# 1. Introduction

This is a capstone project for the Maine Agricultural and Forest Experimentation Station (MAFES) at the University of Maine. This project will fulfill the capstone requirement for a Computer Science Bachelor's degree for Bradan Craig, Drew Marecek, McKade Wing, and Theodore Morin. This project involves developing a web application for an equipment management and tracking system to replace MAFES's current Excel process. The new system will allow all MAFES staff and students to easily view, reserve, and manage agricultural and forestry equipment across their six research farms.

## 1.1. Purpose of This Document

The purpose of this document is to outline the goals and scope of the MAFES Equipment Management System project. It includes an overview of the software's intended functionality and a framework to be used throughout the development process, for design, implementation, and testing purposes. The intended audience for this document includes MAFES customer representatives, MAFES research faculty, the capstone development team, and University of Maine faculty reviewing the document's completeness.

## 1.2. References


*END-To-END Testing Tutorial: What is E2E Testing with Example. (2019, September 18). Guru99.com. https://www.guru99.com/end-to-end-testing.html*

*GeeksforGeeks. (2022, April 25). How to Write Test Cases Software Testing. GeeksforGeeks. https://www.geeksforgeeks.org/software-testing/test-case/#*

MAFES Equipment Inventory_2025_tcm, MAFES. (2025). *MAFES' current equipment inventory management spreadsheet.*

SPR, The Resource Alignment Group. (2025). *Client Q/A session and requirement gathering.*

User Stories, The Resource Alignment Group. (2025). *A list of relevant user stories.*

Williams, L. & NCSU CSC326 Course Pack. (2010). *A (Partial) introduction to software engineering practices and methods* (Seventh).


## 1.3. Purpose of the Product

MAFES currently uses a Google spreadsheet shared among researchers, superintendents, and farm staff to track equipment details, but this has become hard to manage. It's slow, inconsistent, and difficult to update, making it challenging for staff to locate equipment, verify its availability, and maintain accurate usage records.

The new application will address these issues by creating a web-based inventory management system. Users will be able to search for equipment, reserve items, and track their current status, reducing confusion and saving time when trying to find a specific piece of equipment. Role-based authentication will separate admins and user permissions, streamlining equipment requests and data management. Additionally, it will retain the same tracking features as the current spreadsheet, including equipment condition, maintenance history, and location, presenting them in a simplified format.

## 1.4. Product Scope

The following diagrams illustrate how users and admins will interact with the primary functions of the MAFES Equipment Management System. The product will provide a centralized platform for tracking, requesting, and managing equipment across all MAFES locations. It will streamline equipment checkouts, returns, and maintenance tracking while allowing admins to manage users, generate reports, and oversee all equipment activity.

**Use Case Diagram 1: User Interactions & Approval**

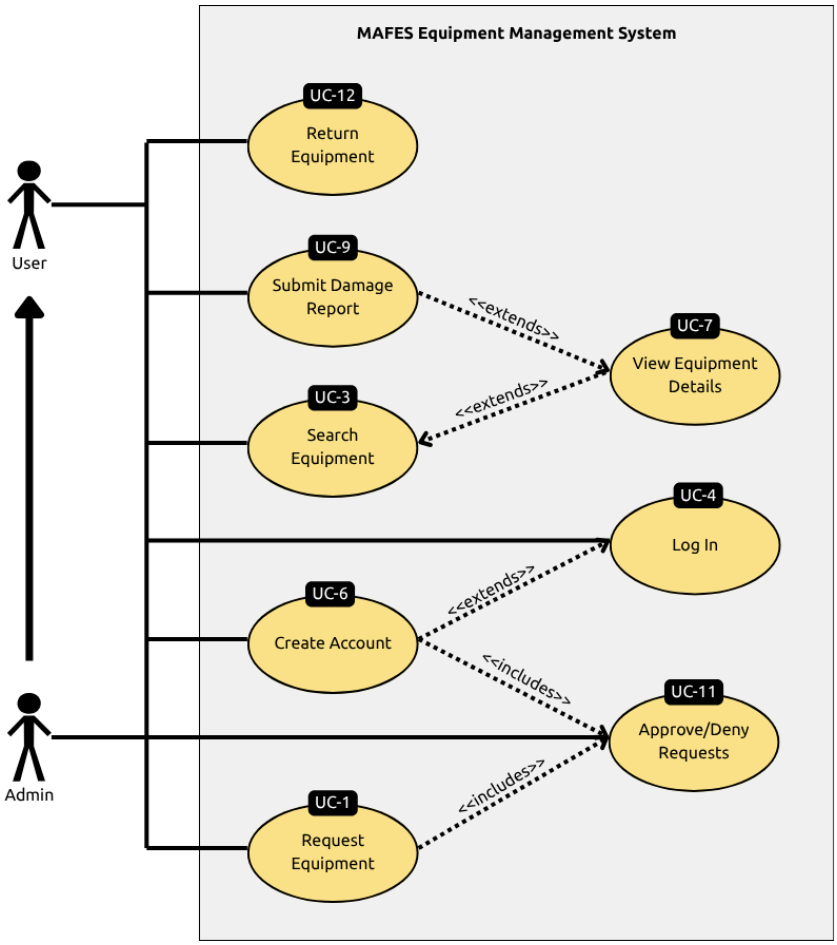

**Figure 1:** This diagram illustrates how users and administrators interact with the system's primary functions, such as account creation, login, equipment search, requests, damage reporting, and returns. It also depicts admin approval processes for certain user actions.

**Use Case Diagram 2: Admin Interactions**

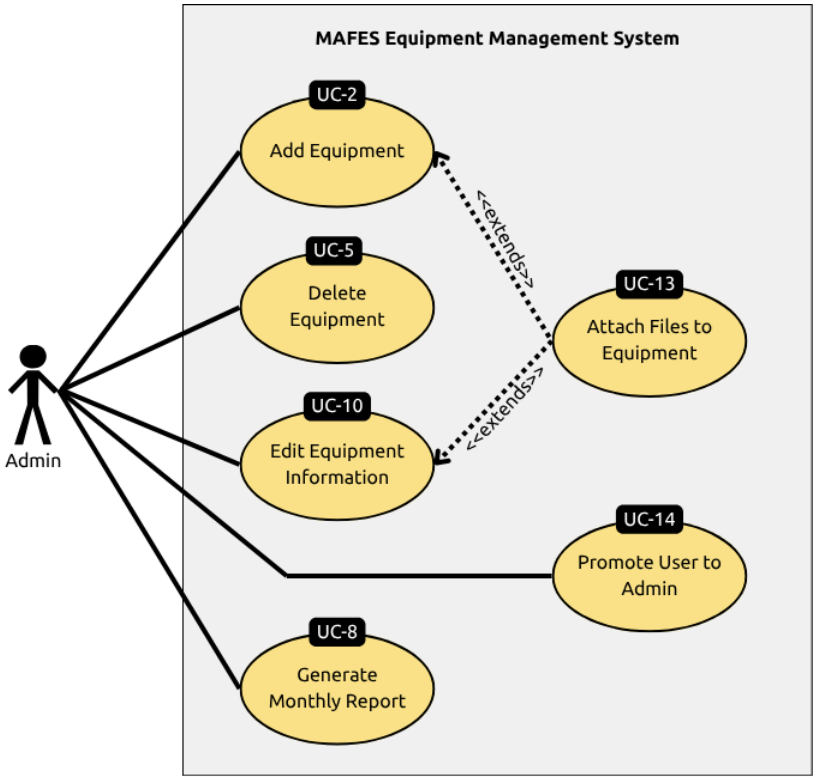


**Figure 2:** This diagram presents the administrative functions of the system, including adding, editing, and deleting equipment, generating reports, attaching files to equipment records, and assigning users admin-level permissions.

## 2. Functional Requirements

This section lists the key functions the system must perform to meet user and admin needs. Each requirement defines specific actions or behaviors necessary for successful system operation.

**FR-0 Equipment Use**

**FR-0.1 Submit Request**

**Use Case 1**

Users will be able to request any equipment from the database. Requests will be logged in the system and sent to the admin.

**FR-0.2 Return Equipment**

**Use Case 12**

Users will be able to mark equipment as returned once they are done using it. The system will update its availability status and log the return date.

**FR-1 Managing Equipment**

**FR-1.1 Add Equipment**

**Use Case 2**

Admins can add new equipment to the database by completing a form that includes key details such as the equipment's name, class, year, make, model, images, files, and assigned farm.

**FR-1.2 Delete Equipment**

**Use Case 5**

Admins can delete equipment from the database by navigating to the equipment page and clicking the "Delete" button. The system will remove the item from the database.

**FR-1.3 Editing Equipment Information**

**Use Case 10**

Admins will be able to edit existing equipment entries to update details such as name, description, images, files, and location when equipment information changes. The admin can also restrict the usage of a piece of equipment and provide a reason (e.g., it's being repaired).

**FR-1.4 Attaching Files to Equipment**

**Use Case 13**

Admins will be able to upload and attach files to a specific equipment record, such as maintenance reports, invoices, and images, for reference and documentation.

**FR-2 Finding Equipment Details**

**FR-2.1 Equipment Filtering**

**Use Case 3**

A filtering option will be available to help users find the equipment they need more easily. The filters will include, at a minimum, farm, class, year, and availability.

**FR-2.2 Viewing Equipment Details**

**Use Case 7**

Users will be able to view full details of each piece of equipment, including its description, images, and availability status, through the equipment information page.

Additionally, users can also see who has that equipment currently checked out, along with their contact information, to coordinate usage or plan future reservations.

**FR-3 User Management**

**FR-3.1 User Account Creation**

**Use Case 6**

Users will be able to create a new account by submitting a signup form with their information and the admin ID that they wish to notify of the request. Once submitted, the system will log the request and send a notification to the designated admin for review. The account will remain inactive until approved through the notification system.

**FR-3.2 Logging In**

**Use Case 4**

Users will be able to log in securely via a login page. Upon authentication, users will be directed to the home page and shown features appropriate to their assigned role (admin vs. user).

**FR-4 Generated Reports**

**FR-4.1 Monthly Reports**

**Use Case 8**

Admins will be able to generate monthly reports summarizing all equipment usage during that month. The report will include all the equipment checked out, including their location, significant damages, and any user-added notes.

**FR-4.2 Damage Reports**

**Use Case 9**

Users will be able to create a damage report for any item that is damaged or broken. Damage reports will include descriptive notes of the incident, as well as any attached files such as invoices and images.

**FR-5 Notification System**

**FR-5.1 Receiving Notifications**

**Use Cases 1, 6, 9**

The system will generate in-system notifications for actions such as request approvals, denials, damage reports, and new account creation.

**FR-5.2 Interactive Notifications**

**Use Case 11**

Admins will be able to approve or deny equipment checkout and account creation requests directly through the notification system. Once an action is taken, the system will automatically update the request status and notify the user of the outcome.

**FR-6 Role-Based Permissions**

**FR-6.1 Promoting a User**

**Use Case 14**

Existing admins will be able to promote an existing user by giving them admin level permissions. This will support the changing of permissions and the role-based architecture.

### 2.1. Use Cases

The following use cases provide a detailed view of how users and admins will interact with the system. They illustrate the main activities involved in managing equipment and outline the expected system behavior in each scenario.

| | | |
|---|---|---|
| **Number** | 1 | |
| **Name** | Requesting Equipment | |
| **Summary** | Users will be able to request access to equipment via the program. | |
| **Priority** | 5 | |
| **Preconditions** | Equipment must exist and be available. The user must be on the equipment page. | |
| **Postconditions** | Notification within the application will be sent to the admin of the equipment. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Request Equipment button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The Request Equipment button is clicked. |
| | 2 | The system verifies that the equipment isn't already checked out by another user. |

| | | |
|---|---|---|
| | 3 | The system logs the request date and user in the database. |
| | 4 | The system notifies the admin specified in the request form. |
| **Extensions** | **Step** | **Branching Action** |
| | 1a | The equipment has been restricted by the admin:<br>● A message appears, alerting the user that they can't reserve the equipment at this time. |
| | 2a | The equipment is already checked out:<br>● The system prevents the request from being submitted and displays a message to the user that the equipment is currently unavailable. |
| | 3a | Duplicate request for the same equipment:<br>● The system alerts the user that a pending request already exists and does not create a new one. |
| **Open Issues** | Should there be a waitlist for equipment reservation? | |

| | | |
|---|---|---|
| **Number** | 2 | |
| **Name** | Adding Equipment | |
| **Summary** | Admins will be able to add new equipment to the database. This will include all of its relevant information, like name, description, location, and images. | |
| **Priority** | 4 | |
| **Preconditions** | The admin adding the equipment must know the name, class, and farm of said equipment. | |
| **Postconditions** | The equipment data is saved to the database and is viewable to all users. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Add Equipment button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The Add Equipment button is clicked. |

| | | |
|---|---|---|
| | 2 | A form pops up for entering equipment details. |
| | 3 | Admin fills out all required fields. |
| | 4 | Admin submits the form. |
| | 5 | Form is sent to the back end of the program. |
| | 6 | The backend reads the form and adds data to the database. |
| | 7 | New equipment appears in the search results. |
| **Extensions** | **Step** | **Branching Action** |
| | 5a | Admin inputs faulty equipment data:<br>● The backend attempts to validate input.<br>● The system detects errors.<br>● The backend sends the form back to the user to make adjustments. |
| **Open Issues** | N/A | |

| | | |
|---|---|---|
| **Number** | 3 | |
| **Name** | Searching Equipment | |
| **Summary** | Users will be able to search all equipment stored in the database. | |
| **Priority** | 5 | |
| **Preconditions** | Equipment must exist in the database. The user is on the search equipment page. | |
| **Postconditions** | Matching equipment is displayed on the UI. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | Search Button is clicked or Filter Conditions are checked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The user searches the name of the equipment |
| | 2 | UI displays all of the equipment matching the criteria. |

| **Extensions** | **Step** | **Branching Action** |
|---|---|---|
| | 1a | User uses filter(s):<br>● The user uses a specific filter instead of a search parameter.<br>● UI displays all equipment associated with that filter. |
| **Open Issues** | Should there be more filters than type, year, and farm? | |

| **Number** | 4 | |
|---|---|---|
| **Name** | Logging In | |
| **Summ ary** | Users will be able to log into the program securely. | |
| **Priority** | 5 | |
| **Preconditions** | User information must exist in the database. The user must be on the login page. | |
| **Postconditions** | The user is directed to the home page after successful authentication. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Log In button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The user inputs login credentials and clicks “Log In.” |
| | 2 | The program authenticates the credentials. |
| | 3 | The system redirects the user to the home page upon success. |
| **Extensions** | **Step** | **Branching Action** |
| | 1a | User inputs incorrect credentials:<br>● The user inputs incorrect information.<br>● The system denies the login request.<br>● The system returns the user to the login page. |
| **Open Issues** | N/A | |

| | | |
|---|---|---|
| **Number** | 5 | |
| **Name** | Deleting Equipment | |
| **Summary** | Admins will be able to delete equipment from the database. | |
| **Priority** | 4 | |
| **Preconditions** | Equipment must exist in the database. Admin is on the equipment page and has an item selected for deletion. | |
| **Postconditions** | The equipment is removed from the system database. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Delete button gets clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The Delete button gets clicked. |
| | 2 | The system displays a message asking the Admin to confirm the deletion of the selected equipment. |
| | 3 | The Admin clicks “Confirm Deletion.” |
| | 4 | The system removes the record from the database. |
| | 5 | The deleted equipment is no longer shown in the UI. |
| **Extensions** | **Step** | **Branching Action** |
| | 2a | Admin cancels deletion:<br>● The system closes the deletion message and no changes are made to the database. |
| **Open Issues** | N/A | |

| | |
|---|---|
| **Number** | 6 |
| **Name** | User Account Creation |
| **Summary** | Users will be able to create an account by submitting a signup form that |

| | | |
|---|---|---|
| | will send a notification to a specified admin ID for approval. | |
| **Priority** | 5 | |
| **Preconditions** | The system must be active and have at least one admin registered to handle approvals. The user must be on the account creation page. | |
| **Postconditions** | A pending user account request is created in the system and a notification is sent to the specified admin for review. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The user clicks Submit on the account creation form. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The user clicks Submit on the account creation form. |
| | 2 | The system validates the entered information. |
| | 3 | The system logs the request for account creation. |
| | 4 | The system sends a request notification to the designated admin. |
| **Extensions** | **Step** | **Branching Action** |
| | 2a | Missing required information:<br>● The system prompts the user to complete all mandatory fields before submitting. |
| | 3a | Invalid admin ID:<br>● The system alerts the user to enter a valid admin ID. |
| **Open Issues** | N/A | |

| | |
|---|---|
| **Number** | 7 |
| **Name** | Viewing Equipment Details |
| **Summary** | Users will be able to view complete information about each piece of equipment. If it's checked out, the page will display the name and email of the current holder. |
| **Priority** | 5 |

| | | |
|---|---|---|
| **Preconditions** | The equipment must exist in the database. The user is on the equipment page and has searched/filtered equipment to display. | |
| **Postconditions** | The user can view all relevant information about the selected equipment. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The user selects an equipment item from the search results. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The user selects a piece of equipment. |
| | 2 | The system retrieves its data from the database. |
| | 3 | Equipment details are displayed in the UI. |
| **Extensions** | N/A | |
| **Open Issues** | N/A | |

| | | |
|---|---|---|
| **Number** | 8 | |
| **Name** | Getting Monthly Report | |
| **Summary** | Admin can request a summary report of equipment usage for the month. | |
| **Priority** | 2 | |
| **Preconditions** | The admin has entered parameters for if they want a usage report for a specific farm vs all farms. | |
| **Postconditions** | A generated monthly usage report is provided to the admin. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Generate Report button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | Admin clicks the Generate Report button. |

| | | |
|---|---|---|
| | 2 | The system compiles equipment usage and maintenance data for the past month. |
| | 3 | Admin receives the report. |
| **Extensions** | N/A | |
| **Open Issues** | Should the reports be downloadable and/or exportable? In what formats? Should there also be a report that is for all farms/equipment? | |

| | | |
|---|---|---|
| **Number** | 9 | |
| **Name** | Adding Damage Reports | |
| **Summary** | Users will be able to submit damage reports, which will be stored in the database and are viewable by admins. | |
| **Priority** | 2 | |
| **Preconditions** | Equipment must exist in the database. The user is on the equipment page. | |
| **Postconditions** | A damage report will be attached to the equipment profile. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Add Damage Report button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The user clicks the “Add Damage Report” button. |
| | 2 | The user fills out the damage report form. |
| | 3 | The user submits a damage report. |
| | 4 | Damage report is added to the equipment profile. |
| | 5 | The report is logged in the database. |
| | 6 | Admin is notified of the report. |
| **Extensions** | **Step** | **Branching Action** |
| | 3a | Required fields left blank: |

| | | |
|---|---|---|
| | | ● The system prompts the user to complete all required information before submitting. |
| **Open Issues** | N/A | |

| | | |
|---|---|---|
| **Number** | 10 | |
| **Name** | Editing Equipment Information | |
| **Summary** | Admins will be able to edit existing equipment entries to update relevant equipment details, upload and attach files like invoices or images, and optionally, restrict users from requesting that equipment (e.g. in repair). | |
| **Priority** | 4 | |
| **Preconditions** | The equipment record must exist in the database. The admin is on the equipment page. | |
| **Postconditions** | The database will be updated with the new information and any attached files, which will be viewable to authorized users. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Edit Equipment button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | Admin clicks the Edit Equipment button. |
| | 2 | A form appears showing the current equipment information. |
| | 3 | Admin updates fields such as name, description, image, location, restricts availability, or attaches files. |
| | 4 | Admin submits form. |
| | 5 | The system saves changes to the database. |
| | 6 | Updated information is visible on the equipment page. |
| **Extensions** | **Step** | **Branching Action** |
| | 3a | Admin makes no changes:<br>● The system closes the edit form without changes without |

| | | |
|---|---|---|
| | | updating the equipment record. |
| **Open Issues** | N/A | |

| | | |
|---|---|---|
| **Number** | 11 | |
| **Name** | Approving or Denying Requests | |
| **Summary** | Admins will be able to approve or deny equipment checkout and account creation requests through the notifications received for these actions. | |
| **Priority** | 3 | |
| **Preconditions** | A pending equipment or account request must exist in the system. | |
| **Postconditions** | The system updates the request status (approved or denied) and notifies the requesting user of the decision. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | User | |
| **Trigger** | Admin receives a system notification for a new request. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | Admin views the notification details. |
| | 2 | Admin optionally writes a brief response message. |
| | 3 | Admin selects “Approve” or “Deny.” |
| | 4 | The system updates the request status. |
| | 5 | The requesting user is notified of the admin’s decision. |
| **Extensions** | N/A | |
| **Open Issues** | N/A | |

| | |
|---|---|
| **Number** | 12 |
| **Name** | Returning Equipment |

| | | |
|---|---|---|
| **Summary** | Users will be able to return equipment, updating its availability. | |
| **Priority** | 3 | |
| **Preconditions** | The equipment must exist and be checked out to the user. The user must be on the equipment page. | |
| **Postconditions** | The equipment is marked as available, and the return is logged in the system. | |
| **Primary Actors** | User | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Return Equipment button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The Return Equipment button is clicked. |
| | 2 | The system validates that the user returning the equipment is the same one who has it checked out. |
| | 3 | The system logs the return date and user into the database. |
| | 4 | The system updates the equipment's status to "Available." |
| **Extensions** | **Step** | **Branching Action** |
| | 1a | User tries to return equipment already "Available":<br>● The system prevents duplicate return submissions and notifies the user that the equipment is already available. |
| | 2a | User attempts to return equipment they didn't check out:<br>● The system denies the return request and displays an error message stating that only the user who checked out the equipment can return it. |
| **Open Issues** | N/A | |

| | |
|---|---|
| **Number** | 13 |
| **Name** | Attaching Files to Equipment |
| **Summary** | Admins will be able to upload and attach files, such as invoices, maintenance records, or images, to specific equipment records when |

| | | |
|---|---|---|
| | adding equipment or editing their details. | |
| **Priority** | 1 | |
| **Preconditions** | The equipment must exist in the system, the admin must have a file they wish to attach, and they are on the equipment page. | |
| **Postconditions** | The attached file will be stored in the system and linked to the selected equipment record. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | N/A | |
| **Trigger** | The Attach File button is clicked. | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | The Attach File button is clicked. |
| | 2 | The system prompts the admin to upload a file. |
| | 3 | Admin selects the file and uploads it. |
| | 4 | The file is linked to the equipment record and stored in the database. |
| | 5 | The attached file appears on that equipment's UI. |
| **Extensions** | **Step** | **Branching Action** |
| | 3a | File upload fails:<br>● The system notifies the admin and allows retrying the upload. |
| **Open Issues** | Should we limit what files are accepted? | |

| | |
|---|---|
| **Number** | 14 |
| **Name** | Promoting User Accounts to Have Admin Permissions |
| **Summary** | Admins will be able to promote user accounts so that they can have admin permissions. |
| **Priority** | 2 |

| | | |
|---|---|---|
| **Preconditions** | An admin exists in the system and there is an existing user account that can be given admin permissions. | |
| **Postconditions** | The original user now has admin privileges. | |
| **Primary Actors** | Admin | |
| **Secondary Actors** | User | |
| **Trigger** | Admin clicks the “Add Admin” Button | |
| **Main Scenario** | **Step** | **Action** |
| | 1 | Original Admin clicks “Add Admin.” |
| | 2 | Admin inputs desired user account email. |
| | 3 | Admin clicks “Confirm.” |
| | 4 | The system validates that the entered user account exists. |
| | 5 | System grants admin permissions to the selected user. |
| **Extensions** | **Step** | **Branching Action** |
| | 4a | The entered user isn’t a valid account:<br>● The system notifies the admin that the email/account details entered are incorrect. |
| **Open Issues** | N/A | |

### 2.2. Functional Requirements Tests

The testing for functional requirements will verify that each feature operates as intended. Each test will confirm that the user and admin interactions are performed correctly and produce the expected results.

| Test Case ID | Use Case(s) | Test Description | Test Steps | Expected Result |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Test 1.1 | 1, 3, 4, 7 | End-to-End Scenario: Successful Equipment Request and Denial | 1. Navigate to the Login screen and log in as a “User”<br>2. Search for an existing piece of equipment<br>3. Click the item to view its details<br>4. Click the “Request Equipment” button<br>5. Log out, and log back in as an “Admin”<br>6. Check notifications and find the new request<br>7. Click “Deny”, and enter a message<br>8. Log out, and log in as a “User”<br>9. Navigate to “Notifications”<br>10. Verify that notification states the correct result of the admin’s decision. | 1. Login is successful (UC 4)<br>2. The equipment and its details are displayed (UC 3, 7)<br>3. After the request, the “Admin” has a pending approval notification (UC 1)<br>4. After denial, the “User” has a notification stating their request was denied (UC 1) |
| Test 1.2 | 1, 3, 4, 7 | Successful Equipment Request and Approval | 1. Navigate to the Login screen and log in as a “User”<br>2. Search for an existing piece of equipment<br>3. Click the item to view its details<br>4. Click the “Request Equipment” button<br>5. Log out, and log back in as an “Admin”<br>6. Check notifications and find the new request<br>7. Click “Approve”, and enter a message<br>8. Log out, and log in as a “User”<br>9. Check that the equipment is under the users name | 1. The “Admin” receives the pending approval notification (UC 1)<br>2. After approval, the “User” has a notification stating their request was approved (UC 1)<br>3. The equipment’s status is now “Unavailable” (UC 7)<br>4. The equipment’s detail page now shows the “User” as the current holder (UC 7) |

| | | | | |
|---|---|---|---|---|
| Test 1.3 | 1, 3, 4, 7 | Successful Equipment Request and Approval | 1. Navigate to the Login screen and log in as a "User"<br>2. Search for an existing piece of equipment<br>3. Click the item to view its details<br>4. Click the "Request Equipment" button<br>5. Log out, and log back in as an "Admin"<br>6. Check notifications and find the new request<br>7. Click "Deny", and enter a message<br>8. Log out, and log in as a "User"<br>9. Check that there is a notification saying that their request was denied | 1. The "Admin" receives the pending approval notification (UC 1)<br>2. After approval, the "User" has a notification stating their request was denied (UC 1)<br>3. The equipment's status is still available (UC 7)<br>4. The equipment's detail page shows "the equipment as available (UC 7) |
| Test 2.1 | 2, 5, 10 | End-to-End Scenario: Full Equipment Lifecycle Management by Admin | 1. Log in as "Admin"<br>2. Navigate to "Add Equipment" and create a new item called "Test Item"<br>3. Confirm that "Test Item" appears in search results<br>4. Go to the "Test Item" page and click "Edit"<br>5. Change its location, name ("Test Item2"), description, image, and attach a file<br>6. Click "Save"<br>7. Verify the fields were updated<br>8. Navigate to the "Test Item2" page and click "Delete"<br>9. Verify "Test Item2" does not appear in search results | 1. The equipment is successfully created and searchable (UC 2)<br>2. The item details are editable, saved, and visible (UC 10)<br>3. The equipment is successfully removed from the system (UC 5) |

| | | | | |
|---|---|---|---|---|
| Test 2.2 | 2 | Negative Path: Add Equipment with Incomplete Form | 1. Log in as “Admin<br>2. Navigate to “Add Equipment”<br>3. Fill out the form, but leave at least one required field blank<br>4. Click “Submit”<br>5. Check that error message appears<br>6. Add input to the empty field<br>7. Verify that equipment is added to the database | 1. The form is not submitted<br>2. An error message is displayed letting the admin know they did not fill out all the required fields (UC 2) |
| Test 3.1 | 3 | Feature Test: Equipment Filtering | 1. Log in as “User”<br>2. Navigate to the equipment search page<br>3. Click “Filters”<br>4. Select “Farm A”<br>5. Select “Large Machinary”<br>6. Select “2014”<br>7. Apply the filter | 1. The search only shows equipment from “Farm A”, created in “2014”, and is a part of the “Large Machinery"<br>class. |
| Test 4.2 | 4 | Negative Path: Incorrect login Credentials | 1. Navigate to the Login page<br>2. Enter a valid username, but an incorrect password<br>3. Click “Log In”<br>4. Show error message for incorrect credentials<br>5. Input correct password<br>6. Click “Log In”<br>7. Verify that the user is redirected to the “Home” page | 1. The Login attempt fails<br>2. An error message is displayed: “Invalid username or password.”<br>3. User is directed to the “Home” page when the correct password is inputted. |
| Test 6.1 | 6, 4 | End-to-End Scenario: New User and Approval | 1. Navigate to the “Sign Up” page<br>2. Fill out the required fields in the “Sign Up” form<br>3. Log in as a user with “Admin” credentials<br>4. Find the new account request and click | 1. The “Admin” receives the new account request (UC 6)<br>2. After approval, the new user can successfully log in (UC 4, 6) |

| | | | | |
|---|---|---|---|---|
| | | | “Approve”<br>5. Log out<br>6. Log in as the newly approved user | |
| Test 6.2 | 6, 4 | End-to-End Scenario: New User and Denial | 1. Navigate to the “Sign Up” page<br>2. Fill out the required fields in the “Sign Up” form<br>3. Log in as a user with “Admin” credentials<br>4. Find the new account request and click “Deny”<br>5. Log out<br>6. Attempt to log in with the new credentials<br>7. Verify that the log in page tells the user that they were denied | 1. The “Admin” receives the new account request (UC 6)<br>2. After approval, the new user can not log in (UC 4) |
| Test 6.3 | 6 | Negative Path: Incomplete Sign Up Form | 1. Navigate to the “Sign Up” page<br>2. Leave at least one required field blank<br>3. Click “Submit”<br>4. Check that error message appears<br>5. Fill in blank field<br>6. Click “Submit”<br>7. Verify that admin are informed of the new “Sign Up” request | 1. The form is not submitted<br>2. An error message appears, letting the user know they have not filled out all the required fields |
| Test 6.4 | 6, 4 | Negative Path: Admin Denies New User | 1. Navigate to the Sign Up page<br>2. Fill out the required fields in the “Sign Up” form<br>3. Log in as “Admin”<br>4. Find the new account request and click “Deny”<br>5. Log out<br>6. Attempt to log in as the newly approved user<br>7. Verify that the flash | 1. The login attempt from the new user fails (UC 4, 6)<br>2. An error message is displayed |

| | | | | |
|---|---|---|---|---|
| | | | message tells user that they have been denied | |
| Test 8.1 | 8 | Feature Test: Generate Monthly Report | 1. Log in as “Admin”<br>2. Navigate to an “active” farm page<br>3. Click “Generate Report”<br>4. Verify that the file was generated<br>5. Verify that the data is from the correct timeline | 1. A report is generated and displayed showing the usage data for the specified farm |
| Test 8.2 | 8 | Feature Test: Generate Report with No Data | 1. Log in as “Admin”<br>2. Navigate to the “inactive” farm page<br>3. Click “Generate Report”<br>4. Verify that file is generated<br>5. Verify that file is not corrupted | 1. A message is displayed stating: “There is nothing to report on for this period” (UC 8) |
| Test 9.1 | 9 | Feature Test: User Submits Damage Report | 1. Log in as “User”<br>2. Navigate to an equipment page<br>3. Click “Add Damage Report”<br>4. Fill out the form<br>5. Click “Submit”<br>6. Log out, and log in as “Admin”<br>7. Navigate to that equipment page<br>8. Check that damage report exists and not corrupted | 1. The “Admin” receives a notification about the damage report<br>2. The damage report is visible on the equipment's profile<br>3. The equipment is now flagged as “Damaged”. |
| Test 9.2 | 7, 9, 1 | Negative Path: Attempt to Request Damaged Equipment | 1. Log in as a “User”<br>2. Navigate to an equipment page flagged as “Damaged”<br>3. Click the “Request Equipment” button<br>4. Verify that there is a pop up that tells the user | 1. The “Request Equipment” button is disabled (“unclickable”) and grayed out.<br>2. The “Request Button” |

| | | | | |
|---|---|---|---|---|
| | | | that the equipment is damaged | |
| Test 9.3 | 7, 9, 1 | Negative Path: Attempt to Request Damaged Equipment | 1. Log in as a “User”<br>2. Navigate to an equipment page flagged as “Damaged”<br>3. Click the “Request Equipment” button | 1. The “Request Equipment” button is disabled (“unclickable”) and grayed out.<br>2. The “Request Button” |
| Test 10.1 | 10, 7, 1 | Feature Test: Admin Restricts Equipment Use | 1. Log in as “Admin”<br>2. Navigate to an “Available” item<br>3. Click “Edit”<br>4. Check the box for “Restrict Requests”<br>5. Click “Save”<br>6. Log out, and log in as “User”<br>7. Search for the item<br>8. Navigate to the equipments page<br>9. Verify that the “Request Equipment” button is grayed out | 1. The item’s page will show “Unavailable” (UC 7)<br>2. The “Request Equipment” button is disabled (grayed out and unclickable), preventing a user from requesting it (UC 1) |
| Test 11.1 | 11 | Admin Denies Equipment Request | 1. Log in as “User.”<br>2. Request an available piece of equipment.<br>3. Log out and log in as “Admin.”<br>4. View notification of new equipment request.<br>5. Click “Approve.”<br>6. Optionally enter a message and confirm.<br>7. Log out and log in as “User.”<br>8. Open “Notifications” to view the decision. | 1. Admin successfully receives the equipment request notification,<br>2. System updates the request status to “Approved.”<br>3. The user receives a notification stating approval.<br>4. Equipment status changes to “Unavailable.” |

| | | | | |
|---|---|---|---|---|
| Test 12.1 | 12 | Successful Equipment Return | 1. Log in as “User.”<br>2. Navigate to the “Equipment” page for an item currently checked out by this user.<br>3. Click the “Return Equipment” button.<br>4. Confirm the return in the prompt.<br>5. Verify the equipment’s status. | 1. The system validates that the user returning the item is the one who checked it out.<br>2. The system logs the return date and user.<br>3. Equipment status updates to “Available.”<br>4. Confirmation message displayed: “Equipment successfully returned.” |
| Test 13.1 | 13 | Attach a File to Equipment | 1. Log in as “Admin.”<br>2. Navigate to the “Equipment” page for an existing item.<br>3. Click the “Attach File” button.<br>4. Select a valid file<br>5. Click “Upload.”<br>6. Verify that the file appears in the equipment’s attachment list. | 1. File upload completes successfully<br>2. The system links the file to the selected equipment record.<br>3. The attached file appears on the equipment’s detail page. |
| Test 14.1 | 14 | Promote a User to Admin | 1. Log in as “Admin.”<br>2. Navigate to the “User Management” page.<br>3. Click “Add Admin.”<br>4. Enter the account you would like to add.<br>5. Click “Confirm.”<br>6. Log out and log in as the newly promoted user. | 1. System validates that the user exists<br>2. The system grants admin privileges to the selected user.<br>3. The new admin has access to admin features. |

## 3. Non-Functional Requirements

This section specifies the performance and quality expectations of the system. It addresses how the application should operate in terms of efficiency, reliability, and overall user experience by defining product, organizational, and external requirements.

**NFR-0 Performance or Efficiency**

NFR-0.1 Real-Time Equipment Availability
Priority: 4
The system will update equipment availability in less than one second to ensure users and admins always see accurate availability information.

NFR-0.2 Filter Load Time
Priority: 2
Each search filter shall return results in less than 2 seconds.

**NFR-1 Reliability and Data Integrity**

NFR-1.1 Daily Database Backup
Priority: 2
The system will automatically perform backups of all data every 24 hours and keep said back ups until replaced to prevent data loss.

NFR-1.2 Notification Delivery Time
Priority: 1
Notifications will be delivered to users within 10 seconds of being triggered.

**NFR-2 Security and Access Control**

NFR-2.1 Password Policy
Priority: 3
All users and admins must log in using their specific account credentials and have a password of at least 10 characters to get in the system.

NFR-2.2 Session Expiration
Priority: 3
User sessions will automatically expire after 30 minutes of inactivity to prevent unauthorized access.

NFR-2.3 Admin Features
Priority: 4
The system shall ensure that all data modification, approval, and deletion functions are inaccessible to any user not explicitly assigned an "Admin" role.

**NFR-3 Usability and Accessibility**

NFR-3.1 Click Efficiency
Priority: 1
Users must be able to locate any piece of equipment within ten user actions or fewer from the home page. User actions include scrolling, clicking, typing, and any other input action that can be used on a standard machine.

NFR-3.2 Navigation Bar
Priority: 2
A navigation bar will be accessible as a column on the left hand side for basic navigation around the application.

**NFR-4 Scalability and Maintainability**

NFR-4.1 Concurrent User Sessions
Priority: 5
The system shall support at least 100 active sessions at once without system failure.

NFR-4.2 Equipment Volume Handling
Priority: 2
The database and search functions shall efficiently handle thousands of equipment records with queries taking less than 3 seconds.

### 3.1. Non-Functional Requirements Tests

The tests for non-functional requirements will measure the system's reliability, responsiveness, and usability under typical operating conditions.

| Test Case ID | Test Description | Test Steps | Expected Result |
|---|---|---|---|
| Test 0.1 | Real-Time Equipment Availability | 1. Two users access the system simultaneously<br>2. User A requests a piece of equipment<br>3. An admin approves User A’s equipment request.<br>4. Click “Log Out”<br>5. Log in as User B<br>6. Navigate to said | The equipment availability updates in real time for User B without delay or manual refresh. |

| | | | |
|---|---|---|---|
| | | equipments page<br>7. Verify that the "Request" button is grayed out | |
| Test 0.2 | Filter Load Time | 1. Navigate to the equipment search page.<br>2. Check multiple filters<br>3. Click "Filter"<br>4. Start timer<br>5. End timer once the page is reloaded<br>6. Verify that the timer is not greater than two seconds. | All filters load and display results within 2 seconds. |
| Test 1.1 | Daily Database Backup | 1. Check the system logs or backup directory after 24 hours of operation. | A backup file is automatically generated every 24 hours without errors. |
| Test 1.2 | Notification Delivery Time | 1. Trigger a notification event<br>2. Observe the timestamp of the event<br>3. Check when the notification is received | The notification is successfully delivered within 10 seconds of the event. |
| Test 2.1 | Password Policy | 1. Log in to the University of Maine system<br>2. Navigate to the system and attempt to create or log in with a password no shorter than 10 characters | Access is denied unless the user is logged in through the University of Maine system, and their password for the system meets the requirements. |
| Test 2.2 | Session Expiration | 1. Log in and remain inactive for 20 minutes.<br>2. A pop-up message appears warning the user they will be logged out in 10 minutes.<br>3. Log user out after 10 minutes<br>2. Attempt to perform any action after inactivity. | The session expires automatically, and the user is redirected to the login page. |
| Test 2.3 | Admin-Only Access | 1. Log in as a regular user.<br>2. Attempt to access or | Regular users are denied access to all modification, approval, deletion, |

| | | | |
|---|---|---|---|
| | | perform restricted actions such as editing, approving requests, deleting equipment, or generating reports.<br>3. Log out and log in as an Admin.<br>4. Attempt the same actions. | and generation functions. The buttons/links for these features are hidden/disabled for regular users. Admins are able to perform these actions successfully. |
| Test 3.1 | Click Efficiency | 1. Log in as a user.<br>2. Attempt to locate a specific piece of equipment using navigation menus or search | The user locates the desired equipment within three clicks or fewer |
| Test 3.2 | Navigation Bar | 1. Log in to the application | The navigation bar should be visible on the left hand side. |
| Test 4.1 | Concurrent User Sessions | 1. Have 100 different users log in simultaneously.<br>2. Attempt standard operations | All 100 sessions remain active and responsive without system slowdowns or errors |
| Test 4.2 | Equipment Volume Handling | 1. Populate the system with several thousand equipment entries.<br>2. Perform searches and filtering operations. | The system maintains performance and responsiveness with a large dataset. |

## 4. User Interface

See “User Interface Design Document for MAFES Equipment Management System.”

## 5. Deliverables

This section outlines the materials that will be provided to the client throughout the course of the project, including the working software and supporting documentation. These deliverables ensure that the client receives a complete and well-documented system.

| Deliverable | Delivery Format | Delivery Date |
|---|---|---|

| | | |
|---|---|---|
| System Requirements Specification | Hard copy & PDF | Oct. 31 |
| System Design Document | Hard copy & PDF | TBD |
| User Interface Design Document | Hard copy & PDF | TBD |
| User Manual | Hard copy & PDF | TBD |
| Administrator Manual | Hard copy & PDF | TBD |
| Source Code | GitHub Repository | Project End |
| Executable Program | GitHub Repository/Web Hosted | Project End |
| Other Required Software | Hard copy & PDF | Project End |

## 6. Open Issues

Any unresolved questions and pending decisions are outlined here. These issues will be addressed later in the development process, as the project progresses.

| Issue ID | Description | Status |
|---|---|---|
| 01 | Determine hosting environment (local server vs. cloud deployment with Docker). | Pending |
| 02 | Choose what type of backend frameworks will be used during development. | Pending |
| 03 | Confirm whether React will be used for the frontend and that all team members are comfortable using it. | Pending |
| 04 | Select a database platform for storing system data. | Pending |
| 05 | Will a waitlist be necessary for equipment reservation? | Pending |
| 06 | Will more filters be needed other than type, year, and farm? | Pending |
| 07 | Should the generated monthly usage reports be downloadable? If so, in what formats? | Pending |
| 08 | Should the generated monthly usage reports be separated by farm, for all equipment, or both? | Pending |

| | | |
|---|---|---|
| 09 | Should we limit what files are accepted? If so, what file types are most beneficial to the client? | Pending |

## Appendix A - Agreement Between Customer and Contractor

This document outlines the agreed-upon system requirements between the customer and our project team. By signing below, both the customer and all team members confirm that they've read and understood the content of this document, and that it accurately reflects the expectations of the system. The project team agrees to build the system based on what's described here, and the customer agrees that these requirements meet their needs.

If any changes need to be made to this document in the future, the customer or project team can submit a request for the change. This process will involve notifying the other party of the suggested change(s), followed by a joint review to understand how the change(s) may affect the project's scope, timeline, or deliverables. No changes will be made until both sides agree and sign off on the updated document.

| | | |
|---|---|---|
| Tyler Messerschmidt | Tyler Messerschmidt | 10/29/25 |
| Customer Name Printed | Customer Signature | Date |
| Lee Hecker | *No longer a signatory on this project* | N/A |
| Customer Name Printed | Customer Signature | Date |
| Bradan Craig | Bradan Craig | 10/29/25 |
| Team Member Name Printed | Team Member Signature | Date |
| Drew Marecek | Drew Marecek | 10/29/25 |
| Team Member Name Printed | Team Member Signature | Date |
| McKade Wing | McKade Wing | 10/29/25 |
| Team Member Name Printed | Team Member Signature | Date |
| Theodore Morin | Theodore Morin | 10/29/25 |
| Team Member Name Printed | Team Member Signature | Date |

Customer Comments:

## Appendix B - Team Review Sign-off

All team members have read through this document and agree with its content and structure. Each team member has had the opportunity to provide feedback and propose revisions during the creation of this document. By signing below, team members confirm their approval of the outlined system requirements. Any minor comments or points of clarification can be noted in the space provided.

| Bradan Craig | Bradan Craig | 10/29/25 |
|---|---|---|
| Team Member Name Printed | Team Member Signature | Date |

Comments:

| Drew Marecek | Drew Marecek | 10/29/25 |
|---|---|---|
| Team Member Name Printed | Team Member Signature | Date |

Comments:

| McKade Wing | McKade Wing | 10/29/25 |
|---|---|---|
| Team Member Name Printed | Team Member Signature | Date |

Comments:

| Theodore Morin | Theodore Morin | 10/29/25 |
|---|---|---|
| Team Member Name Printed | Team Member Signature | Date |

Comments:

## Appendix C - Document Contributions

Each team member contributed to the development of this document to ensure the final version is accurate, precise, and complete. This included researching SRS structure, writing project overviews, creating UML diagrams, and creating functional and non-functional requirements. The table below shows each team member's contributions, along with an estimated percentage of their work on the document.

| Team Member | Contributions | Estimated % |
|---|---|---|
| Bradan Craig | Functional Requirements + Non-Functional Test Cases | 25% |
| Drew Marecek | Non-functional requirements + Test Cases | 25% |
| McKade Wing | Document Write-Up + UML Diagrams + FR Edits | 25% |
| Theodore Morin | Test cases for Functional/Non-Functional Requirements | 25% |