Test Plan for Project Tasks Management System

**1. Test Objectives**

* 1. **Functionality Verification**: To ensure that all features of the system function as expected according to the design specifications and user requirements.
     + User login
     + User Registration
     + Project Creation;
     + Task Creation;
     + Project Update;
     + Task Update;
     + Project Listing;
     + Search Functionality;
     + Project Administration for different roles;
     + Data Storage;
     + Event Logging.
     + CSV export for projects and tasks.
  2. **User Interface Testing**: To ensure that the user interface is intuitive, user-friendly, and consistent across all modules of the system. This includes testing the layout, design, and navigation of the application.

**2. Test Scope**

1. **User Account Management**: This involves testing the user registration and user login. We will verify that users can successfully create accounts, log in, log out.
2. **Project Administration and Management:** This involves testing user roles and permissions management and validate the user interface for assigning project ownership, managing user roles, and permissions, test access control for different user roles (owner, user, administrator).
3. **Task Management:** This involves testing task creation and update functionalities within projects, Validate task creation UI, task editing UI, and feedback mechanisms, test various scenarios such as adding/deleting tasks, updating task details, and task dependencies.
4. **Project Listing Validation:** This involves testing the display of projects, pagination, and sorting options, test responsiveness to changes in project status (etc.).
5. **Event Logging Validation:** This involves testing event logging functionality, validate logging of key system events, including timestamps and relevant data, test log accessibility, search, and filtering capabilities.
6. **Search Functionality Validation:** This involves testing the search feature and its integration, validate UI elements for initiating searches, search filters, and result presentation, test search accuracy.

**3. Test Strategy**

| **Test Level** | **Test Types** | **Testing Technique** | **Tools used** | **Description** |
| --- | --- | --- | --- | --- |
| System Testing | Functional Testing | Experience-based Technique | Manual Testing, Test Cases | Testers will interact with the system as end users would, trying out various functionalities and observing the system's behavior. |
| System Testing | Regression Testing | Automated Testing | Selenium | Involves re-running our test cases from the functional testing phase to ensure that previously developed and tested software still performs after a change. |
| Integration Testing | Functional Testing | Black Box Testing | Postman | Testing the interaction between different components of the software. |
| Acceptance Testing | User Acceptance Testing (UAT) | Black Box Testing | Manual Testing, Test Cases | The final phase of testing, where actual software users test the software to make sure it can handle. |
| System Testing | End-to-End Testing | Black Box Testing | Selenium | Testing the entire software from start to end. |

**4. Test Schedule**

| **Activity** |  | **Start Date** | **End Date** |
| --- | --- | --- | --- |
| Test Case Development and Test Environment Setup |  | 2024-05-06 |  |
| Test Execution and Bug Fixing |  | 2024-05-09 |  |
| **5. Resource Planning**   * Tester 1 (KG): Test cases, Selenium, Postman * Tester 2 (DN): Test cases, Selenium, Postman * Tester 3 (TS): Test cases, Selenium, Postman |  |  |  |

**6. Risk Management**

| **Risk** | **Potential Impact** | **Likelihood** | **Mitigation Strategy** |
| --- | --- | --- | --- |
| Delays in development | Could push back the start of the testing phase, leaving less time for testing. | Medium | Early involvement in the development process to help identify and address potential delays. |
| Unavailability of test environments | Could prevent testing from taking place as scheduled. | Low | Having a backup test environment ready to use in case the primary environment is unavailable. |
| High defect leakage | Lots of bugs fixing after release. | High | Thorough test case review and execution to catch as many defects as possible during the testing phase. |

**7. Entry Criteria**

* The development team has delivered a stable build for testing.
* The test environment is ready and test data has been prepared.

**8. Exit Criteria**

* All planned test cases have been executed.
* All critical and high priority bugs have been fixed and retested.

**9. Deliverables**

* **Test Case Report Document:** This document, maintained in an Excel file, will contain all the test cases that were executed during the testing phase.
* **Test Scripts:** If automated testing was used, the test scripts will be provided. These scripts can be used for regression testing in future development cycles. The test scripts for this project are located in the e2e folder.
* **Bug Reports:** A detailed report of all the bugs found during testing will be provided and tracked in Jira, and the link to the project's Jira board will be provided.
* **Test Summary Report:**  This report, will be provided in jira where will be linked with project.