

**Assignment: Test Plan Creation Objective:** The objective of this assignment is to create a comprehensive test plan for a given software application. The test plan should outline the testing strategy, objectives, scope, resources, and schedule for testing the application.

## 1. Introduction

**Application Name:** Task Manager Pro

**Description:** Task Manager Pro is a web-based application designed to help users manage their tasks efficiently. It allows users to create, update, delete, and categorize tasks. The application also includes features for setting deadlines, assigning tasks to team members, and tracking task progress.

**Version:** 1.0

**Prepared by:** Akula Sheshu Varma

## 2. Testing Objectives

- **Functionality:** Ensure all features work as expected.
- **Reliability:** Verify the application performs consistently under various conditions.
- **Performance:** Assess the application's response time, load handling, and scalability.
- **Security:** Identify and fix vulnerabilities to protect data and user information.
- **Usability:** Evaluate the user interface for ease of use and intuitiveness.

## 3. Test Scope

### In-Scope

- User authentication and authorization
- Task creation, update, deletion, and categorization
- Task assignment and tracking
- Notification system
- User interface responsiveness and accessibility
- Integration with third-party services (e.g., email notifications)

### Out-of-Scope

- Future features planned for version 2.0
- Performance testing under extreme load conditions
- Non-web-based access (e.g., mobile apps)

## 4. Test Strategy

### Types of Testing

- **Functional Testing:** Verify all functions of the application.

- **Regression Testing:** Ensure new code changes do not affect existing functionality.
- **Integration Testing:** Test the interaction between different modules of the application.
- **Performance Testing:** Assess the speed, scalability, and stability.
- **Security Testing:** Identify vulnerabilities and security issues.
- **Usability Testing:** Evaluate the application's user interface and user experience.

5. Test Resources

Personnel

- **Test Manager:** John Doe
- **Test Engineers:** Jane Smith, Bob Johnson
- **Automation Engineer:** Alice Brown
- **Security Analyst:** Charlie Davis

Tools

- **Test Management Tool:** TestRail
- **Automation Tool:** Selenium
- **Performance Testing Tool:** JMeter
- **Bug Tracking Tool:** Jira
- **Security Testing Tool:** OWASP ZAP

Equipment

- Workstations with necessary software installed
- Test servers mimicking the production environment

6. Test Schedule

Activity	Start Date	End Date	Responsible
Test Planning	2024-06-01	2024-06-05	John Doe
Test Case Development	2024-06-06	2024-06-12	Jane Smith
Test Environment Setup	2024-06-06	2024-06-08	Bob Johnson
Test Execution	2024-06-13	2024-06-20	Test Team
Defect Reporting	2024-06-13	2024-06-20	Test Team
Retesting & Regression	2024-06-21	2024-06-25	Test Team
Test Closure	2024-06-26	2024-06-27	John Doe

## 7. Test Environments

### Development Environment

- **OS:** Ubuntu 20.04
- **Web Server:** Apache 2.4
- **Database:** MySQL 8.0
- **Browser:** Latest versions of Chrome, Firefox, Edge

### Staging Environment

- **OS:** Ubuntu 20.04
- **Web Server:** Apache 2.4
- **Database:** MySQL 8.0
- **Browser:** Latest versions of Chrome, Firefox, Edge

## 8. Test Cases

### Example Test Cases

#### 1. Test Case ID: TC001

- **Description:** Verify user can create a task
- **Preconditions:** User is logged in
- **Steps:**
  1. Navigate to the "Create Task" page.
  2. Fill in task details.
  3. Click "Save."
- **Expected Result:** Task is created and displayed in the task list.

#### 2. Test Case ID: TC002

- **Description:** Verify user can update a task
- **Preconditions:** User is logged in and has an existing task
- **Steps:**
  1. Navigate to the task list.
  2. Click on an existing task.
  3. Update task details.
  4. Click "Save."
- **Expected Result:** Task details are updated and displayed.

#### 3. Test Case ID: TC003

- **Description:** Verify user can delete a task
- **Preconditions:** User is logged in and has an existing task
- **Steps:**
  1. Navigate to the task list.
  2. Click on the delete button next to an existing task.
  3. Confirm deletion.
- **Expected Result:** Task is removed from the task list.

## 9. Risk Assessment

### Potential Risks

- **User authentication issues:** Could prevent users from accessing the application.
- **Data loss:** Could occur during CRUD operations.
- **Performance degradation:** Under heavy load, the application may slow down or crash.
- **Security vulnerabilities:** Could expose user data to unauthorized access.

### Mitigation Strategies

- **Authentication Issues:** Implement robust authentication mechanisms and regular testing.
- **Data Loss:** Ensure regular backups and thorough testing of CRUD operations.
- **Performance Issues:** Conduct regular performance testing and optimize code.
- **Security Vulnerabilities:** Perform regular security audits and implement security best practices.

## 10. Appendices

### Glossary

- **CRUD:** Create, Read, Update, Delete
- **Selenium:** A tool for automating web browsers
- **JMeter:** A tool for performance testing
- **OWASP ZAP:** A security testing tool

### References

- [Task Manager Pro User Manual]
- [TestRail Documentation]
- [Selenium Documentation]