Quality Assurance Plan

HR Management Application

4th November 2023

19020341 - Avishka Hettiarachchi

1 Introduction

1.1 Purpose

The purpose of this Quality Assurance Test Plan is to outline the testing strategy and approach for the HR Management application. This document serves as a comprehensive guide to ensure the systematic and effective testing of the project, encompassing the login functionality, employee profile management, leave tracking, attendance reports, and timesheet approval/rejection features.

1.2 Project Overview

The HR Management application is a pivotal project designed to optimize human resources processes within the organization. This application facilitates secure and efficient management of employee profiles, leave tracking, attendance reporting, and the approval or rejection of timesheets. The primary focus of this Quality Assurance Test Plan is to ensure the reliability, accuracy, and security of the HR Management application.

2 Scope

2.1 In-Scope

❖ ***Functionality Testing***

- Authentication and Authorization
- Employee Profile Management
- Leave Tracking
- Attendance Reports
- Timesheet Approval/Rejection

❖ **Security Testing**

- Data Security
- Validation of encryption mechanisms for sensitive data.
- Confirmation that access controls prevent unauthorized access.

❖ ***Performance Testing***

2.2 Out-of-Scope

- Compatibility Testing
- Continuous Integration/Continuous Deployment (CI/CD) Pipeline
- Network and Infrastructure Testing
- Regulatory Compliance

3 Testing Strategy

3.1 Product/Application/Solution Risks

Risks	Criticality	Mitigation Strategy
Authentication Issues	High	Thorough testing of login functionality with valid and invalid credentials. Implement secure password handling and account lockout mechanisms.
Data Security Issues	High	Regularly update security protocols, conduct penetration testing, and implement access controls.
Data Accuracy Issues	Medium	Execute test cases to validate data accuracy, implement data validation checks, and conduct data integrity testing.
Timesheet Approval/Rejection Issues	Medium	Develop and execute test scenarios for approval/rejection functionality, and ensure proper workflow testing.

3.2 Level of Testing

Test Type	Description
Unit Testing	Validate individual components and functions.
Integration Testing	Verify interactions between different modules.
System Testing	Perform end-to-end testing of the entire system.
Performance Testing	Assess system responsiveness and stability under various user loads.

3.2.1 Functional Testing

Functional Testing is a key component of the testing strategy, focusing on validating the correctness and reliability of the HR Management application's functionalities. This testing type involves the verification of critical features such as authentication and authorization, employee profile management, leave tracking, attendance reporting, and timesheet approval/rejection. Functional Testing ensures that each aspect of the system works according to specified requirements, providing confidence in the application's ability to meet user needs and expectations.

3.2.2 Regression Testing

Regression Testing is integral to the testing strategy, aimed at ensuring that new changes, enhancements, or bug fixes do not negatively impact existing functionalities. As the HR Management application evolves, regular updates and modifications are inevitable. Regression Testing involves the re-execution of functional tests to confirm the stability and integrity of existing features. This process prevents unintended side effects that may arise from code changes, ensuring that the application maintains its reliability across various software iterations.

3.3.3 Non-Functional Testing

Non-Functional Testing is a crucial aspect of the testing strategy, focusing on evaluating elements such as performance, security, and usability. Performance Testing involves assessing the system's responsiveness under different user loads, identifying and optimizing performance bottlenecks. Security Testing aims to validate data security measures, preventing unauthorized access and ensuring robust protection against potential threats. Usability Testing focuses on evaluating the user interface for intuitiveness and user-friendliness, gathering feedback on the overall user experience.

4. Test Approach

4.1 Test Design Approach

The test design approach for the HR Management application is a systematic and comprehensive strategy aimed at ensuring the thorough validation of its functionalities. The design of test cases will follow a structured process, incorporating various testing techniques to cover different aspects of the application.

Equivalence Partitioning

- This technique involves dividing the input space into classes or partitions and selecting representative values from each class for testing. For example, testing different categories of user roles during the authentication process.

Boundary Value Analysis

- Focusing on values at the edges or boundaries of input domains, this technique helps identify potential errors. For instance, testing the application's behavior with minimum and maximum values for leave requests or timesheet entries.

Decision Table Testing

- Creating decision tables to represent different combinations of conditions and actions. In the HR Management application, decision table testing can be effectively utilized in the process of determining leave approval or rejection based on various conditions.

User Story Testing

- Aligning test cases with user stories to validate that the application meets user requirements. This technique ensures that the software aligns with the intended user experience and functionality.

4.2 Execution Strategy

4.3.1 Entry Criteria

Entry Criteria	Conditions	Comments
Test environment(s) is available	✓	
Test data is available	✓	
Code has been merged successfully	✓	
Development has completed unit testing		
Test cases and scripts are completed, reviewed and approved by the Project Team		

3.2.2 Exit criteria

Exit Criteria	Conditions	Comments
100% Test Scripts executed	✓	
90% pass rate of Test Scripts		
No open Critical and High severity defects	✓	
All remaining defects are either cancelled or documented as Change Requests for a future release		
All expected and actual results are captured and documented with the test script	✓	
All test metrics collected based on reports from daily and Weekly Status reports	✓	
All defects logged in -Defect Tracker/Spreadsheet		
Test environment cleanup completed and a new back up of the environment		

3.3 Defect Management

Severity	Impact
1 (Critical)	▪ *Functionality is blocked and no testing can proceed* ▪ *Application/program/feature is unusable in the current state*
2 (High)	▪ *Functionality is not usable and there is no workaround but testing can proceed*
3 (Medium)	▪ *Functionality issues but there is a workaround for achieving the desired functionality*
4 (Low)	▪ *Unclear error message or cosmetic error which has minimum impact on product use.*

5. Test Team Structure

5.1 Team Structure

#	Role	Resource Count
1	QA Manager	1
2	QA Leads	1
3	Senior QA Engineers	2
4	QA Engineers	2

5.2 Roles and Responsibilities

QA Manager

The QA Manager is responsible for the overall planning, coordination, and execution of the testing process.

-Responsibilities

- Develop the overall testing strategy and approach.
- Define the scope of testing and testing objectives.
- Allocate resources and define timelines for testing phases.
- Oversee and coordinate the efforts of the entire test team.

QA Leads

QA Leads are responsible for leading a specific testing team or a subset of the overall testing effort. They act as a bridge between the QA Manager and the testing team members.

Responsibilities

- Develop detailed test plans and test cases based on project requirements.
- Assign tasks to Senior QA Engineers and QA Engineers.
- Monitor the progress of testing activities and report to the QA Manager.
- Ensure that test cases are executed according to the defined testing strategy.

Senior QA Engineers

Senior QA Engineers are experienced professionals responsible for the detailed planning and execution of test cases.

Responsibilities

- Develop detailed test cases based on functional and non-functional requirements.
- Execute test cases and document results.
- Identify and log defects with detailed information for the development team.
- Mentor and provide guidance to QA Engineers.
- Participate in test planning and strategy discussions.
- Collaborate with QA Leads to address testing challenges.

QA Engineers

QA Engineers are responsible for the hands-on execution of test cases, identifying defects, and contributing to the overall testing effort.

Responsibilities

- Execute test cases according to the defined testing strategy.
- Log defects with detailed information for resolution.
- Collaborate with Senior QA Engineers to improve testing processes.
- Participate in testing meetings and provide updates on progress.

6. Test Schedule

Phase	*start date*	*end date*
Test planning	05 - November - 2023	12 - November - 2023
Test Environment setup	13 - November - 2023	16 - November - 2023
Test case design	17 - November - 2023	1 - December - 2023
Test execution	2 - December - 2023	23 - December - 2023
Defect Reporting and Resolution	23 - December - 2023	23 - February - 2024
Regression testing	1 - January - 2024	14 - January - 2024
Non-functional testing	15 - January - 2024	8 - February - 2024
User Acceptance testing	9 - February - 2024	23 - February - 2024
Test Closure	24 - February - 2024	2- March - 2024

7. Test Reporting

7.1.Test Reporting Approach

#	Report Name	Owner	Audience	Frequency
1	TEST PROGRESS REPORT	QA Leads	QA Manager, Project Manager, Development Team, Stakeholders	Bi-weekly
2	TEST PROGRESS REPORT	Senior QA Engineers	QA Leads, QA Manager	Weekly

7.2. Quality Matrices

- Test Coverage Matrix
- Defect Density Matrix
- Execution Efficiency Matrix
- Performance Matrix
- User Satisfaction Matrix

8. Test Environment Requirements

Hardware Requirements

Server Infrastructure
- Dedicated servers for hosting the HR Management application.

Client Machines
- Workstations for testing, equipped with a variety of browsers and operating systems to ensure cross-browser compatibility.

Software Requirements

Operating Systems
- Support for multiple operating systems

Web Browsers
- Compatibility with web browsers : Google Chrome, Mozilla Firefox, Microsoft Edge, and Safar

Database Management System
- Compatibility with the designated database management system used for storing application data.

Web Server
- A dedicated web server for hosting the HR Management application.

Test Tools

- Test Management Tools

Automation Tools

9. Dependencies and Assumptions

Dependencies

1. Availability of the HR Management application builds and test environments.

2.Availability of skilled testing resources, including QA Leads, Senior QA Engineers, and QA Engineers.

3. Availability of realistic and diverse test data for comprehensive testing.

Assumptions

1. Issues identified during testing will be addressed by the development team in a timely manner.

2. The test environment will remain stable throughout the testing process.

3. Testing activities will adhere to the predefined schedule.