**🛫 Test Plan: Reserves Availability Module for FAs and Pilots**

**1. Test Plan Identifier**

* **Version**: 1.0
* **Module**: Reserves Availability
* **Stakeholders**: QA Team, Development Team, Product Owners

**2. Introduction**

The Reserves Availability module manages the scheduling and availability of Flight Attendants and Pilots. This test plan aims to ensure the module functions correctly under various scenarios, including peak loads and edge cases.

**3. Test Items**

* **Login System**: Authentication and authorization for FAs and Pilots.
* **Availability Dashboard**: Real-time display of available reserves.
* **Scheduling Interface**: Tools for assigning reserves to flights.
* **Notification System**: Alerts for upcoming assignments or changes.
* **Data Integrity**: Ensuring accurate and up-to-date information.

**4. Approach**

**4.1 Unit Testing**

* **Objective**: Verify individual components like login, scheduling, and notifications.
* **Tools**: JUnit, Mockito.

**4.2 Integration Testing**

* **Objective**: Ensure seamless interaction between modules such as login, scheduling, and the notification system.
* **Tools**: Postman, Selenium.

**4.3 Regression Testing**

* **Objective**: Confirm that new changes don't adversely affect existing functionalities.
* **Tools**: Selenium, TestNG.

**4.4 Acceptance Testing**

* **Objective**: Validate the system against business requirements.
* **Tools**: Cucumber, FitNesse.

**4.5 Performance Testing**

* **Objective**: Assess system behavior under load.
* **Tools**: Apache JMeter.

**5. Test Cases**

**5.1 Functional Test Cases**

| **Test Case ID** | **Description** | **Expected Outcome** |
| --- | --- | --- |
| TC\_FA\_001 | Login as FA with valid credentials | Access granted to FA dashboard |
| TC\_FA\_002 | View available reserves | List of available reserves displayed |
| TC\_FA\_003 | Assign reserve to flight | Reserve assigned successfully |
| TC\_FA\_004 | Receive assignment notification | Notification received via email/SMS |
| TC\_FA\_005 | Logout | Successful logout |

**5.2 Performance Test Cases**

| **Test Case ID** | **Description** | **Expected Outcome** |
| --- | --- | --- |
| TC\_PT\_001 | Simulate 1000 concurrent users | System response time < 3 seconds |
| TC\_PT\_002 | Stress test with 5000 concurrent users | System handles load without crashes |

**5.3 Security Test Cases**

| **Test Case ID** | **Description** | **Expected Outcome** |
| --- | --- | --- |
| TC\_SEC\_001 | SQL Injection attempt in login | Access denied, no data leakage |
| TC\_SEC\_002 | Cross-site scripting (XSS) in notification | No script execution, input sanitized |

**5.4 Usability Test Cases**

| **Test Case ID** | **Description** | **Expected Outcome** |
| --- | --- | --- |
| TC\_US\_001 | Navigate through scheduling interface | Intuitive and responsive UI |
| TC\_US\_002 | Access help documentation | Documentation loads without errors |

**6. Pass/Fail Criteria**

* **Pass**: If the system behaves as expected under all test scenarios.
* **Fail**: If the system deviates from expected behavior, including crashes, incorrect data handling, or security vulnerabilities.

**7. Suspension Criteria and Resumption Requirements**

* **Suspension Criteria**: Testing will be suspended if critical defects are found that prevent further testing.
* **Resumption Requirements**: Testing will resume once defects are resolved and the system is stable.

**8. Test Deliverables**

* Test Plan Document
* Test Case Specifications
* Test Execution Logs
* Defect Reports
* Test Summary Report

**9. Testing Tools**

* **Functional Testing**: Selenium, Postman,Oracle SQL Developer,COSMOS(scheduling Sysytem)
* **Performance Testing**: Apache JMeter
* **Security Testing**: OWASP ZAP, Burp Suite
* **Version Control**: Git, GitHub

**10. Schedule**

| **Phase** | **Duration** |
| --- | --- |
| Test Planning | 1 week |
| Test Design | 2 weeks |
| Test Execution | 4 weeks |
| Defect Resolution | 2 weeks |
| Final Reporting | 1 week |

**11. Risks and Mitigations**

| **Risk** | **Mitigation Strategy** |
| --- | --- |
| Delays in defect resolution | Allocate additional resources for bug fixing |
| Incomplete requirements from stakeholders | Regular meetings to clarify requirements |
| Tool compatibility issues | Ensure all tools are updated and compatible |

**12. Exit Criteria:**

* Complete Test Execution
* All High Priority/Show stopper defects are fixed
* Achieving 90% code coverage for the module's core functionalities.
* Performance tests indicating that the system can handle peak loads without degradation.
* Compliance with aviation industry standards and regulations.
* Approval from QA leads and product owners to proceed to deployment.