Borisova,Yoanna Y.I.

Rostislav Tinchev, Teodor Genov, Vladimir Katrandzhiev, Yoanna Borisova, Monika Kerulyte, Ignas Kyb

Test plan

Airport Traffic

**Revision and Signoff Sheet**

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description of Change |
| 1.0 | 15.03.18 | Yoanna Borisova | Create document, initial structure, fill in information |
|  |  |  |  |
|  |  |  |  |

**Approvers List**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Role | Approver / Reviewer | Approval / Review Date |
| xfdbxd | dthbdftr |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Reference Documents**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Document Name** |
| 1.0 |  | ORANGEHRM VERSION 3.0 – MY INFO MODULE -FSD |
|  |  |  |

Table of Contents

[1. INTRODUCTION 3](#_Toc508839600)

[1.1. Purpose 3](#_Toc508839601)

[1.2. Project Overview 3](#_Toc508839602)

[2. TEST STRATEGY 3](#_Toc508839603)

[2.1. Test Objectives 3](#_Toc508839604)

[2.2. Test Assumptions 3](#_Toc508839605)

[2.3. Test Principles 4](#_Toc508839606)

[3. EXECUTION STRATEGY 4](#_Toc508839607)

[3.1. Entry and Exit Criteria 4](#_Toc508839608)

[3.2. Validation and Defect Management 4](#_Toc508839609)

[3.3. Defect tracking & Reporting 5](#_Toc508839610)

[4. TEST MANAGEMENT PROCESS 5](#_Toc508839611)

[4.1. Test Risks and Mitigation Factors 5](#_Toc508839612)

[5. TEST CASES 7](#_Toc508839611)

[6. APPROVALS 9](#_Toc508839613)

# INTRODUCTION

## Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the Airport Traffic simulation. The document introduces:

* Test Strategy: rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
* Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
* Test Management: process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster)

## Project Overview

The goal of the project is to create a user’s configurable benchmarkable program which simulates probable real air space traffic conditions and its automated handling by the application itself. During continuous improvement phases and testing the project will provide in its final version with the following: a forecast for future events that will help with the operational planning of airplane companies, traffic monitoring and control, possibility to train employees in a safe environment, analysis of how the separate elements interact with each other. Apart from the upper mentioned functionalities, the application will also be abiding a set of rules defined by the client.

# TEST STRATEGY

## Test Objectives

The objective of the test is to verify that the functionality of the Airport Traffic works according to the specifications.

The test will execute and verify the test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing via CR.

The final product of the test is twofold:

* A production-ready software;
* A set of stable test scripts that can be reused for Functional and UAT test execution.

## Test Assumptions

**Key Assumptions**

* Production like data required and be available in the system prior to start of Functional Testing

**General**

* Exploratory Testing would be carried out once the build is ready for testing
* Performance testing is not considered for this estimation.

## Test Principles

* Testing will be focused on meeting the business objectives, cost efficiency, and quality.
* There will be common, consistent procedures for all teams supporting testing activities.
* Testing processes will be well defined, yet flexible, with the ability to change as needed.
* Testing activities will build upon previous stages to avoid redundancy or duplication of effort.
* Testing environment and data will emulate a production environment as much as possible.
* Testing will be a repeatable, quantifiable, and measurable activity.
* Testing will be divided into distinct phases, each with clearly defined objectives and goals.
* There will be entrance and exit criteria.

# EXECUTION STRATEGY

## Entry and Exit Criteria

* The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
* The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
* Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
* Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

## Validation and Defect Management

Defects found during the Testing will be categorized according to the following table

|  |  |
| --- | --- |
| **Severity** | **Impact** |
| 1 (Critical) | * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss * It causes an abnormal return to the operating system (crash or a system failure message appears). * It causes the application to hang and requires re-booting the system. |
| 2 (High) | * It causes a lack of vital program functionality with workaround. |
| 3 (Medium) | * This Bug will degrade the quality of the System. However, there is an intelligent workaround for achieving the desired functionality - for example through another screen. * This bug prevents other areas of the product from being tested. However other areas can be independently tested. |
| 4 (Low) | * There is an insufficient or unclear error message, which has minimum impact on product use. |
| 5(Cosmetic) | * There is an insufficient or unclear error message that has no impact on product use. |

## Defect tracking & Reporting

Following flowchart depicts Defect Tracking Process:

**Approved?**

**Start**

**Tester:**

**Report defects**

**Dev Lead**

**Assign defects**

**Developer:**

**Fixes defects**

**Tester:**

**Retests the product**

**No**

**Stop**

**Close defect**

**Yes**

**Test Lead**

**Validate defects**

# TEST MANAGEMENT PROCESS

## Test Risks and Mitigation Factors

| Risk | Prob. | Impact | Mitigation Plan |
| --- | --- | --- | --- |
| **SCHEDULE**  Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the UAT scheduled start date. | High | High | * The testing team can control the preparation tasks (in advance) and the early communication with involved parties. * Some buffer has been added to the schedule for contingencies, although not as much as best practices advise. |
| **RESOURCES**  Not enough resources, resources on boarding too late (process takes around 15 days. | Medium | High | Holidays and vacation have been estimated and built into the schedule; deviations from the estimation could derive in delays in the testing. |
| **DEFECTS**  Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve. | Medium | High | Defect management plan is in place to ensure prompt communication and fixing of issues. |
| **SCOPE**  Scope completely defined | Medium | Medium | Scope is well defined, but the changes are in the functionality are not yet finalized or keep on changing. |
| Natural disasters | Low | Medium | Teams and responsibilities have been spread to two different geographic areas. In a catastrophic event in one of the areas, there will resources in the other areas needed to continue (although at a slower pace) the testing activities. |
| Non-availability of Independent Test environment and accessibility | Medium | High | Due to non-availability of the environment, the schedule gets impacted and will lead to delayed start of Test execution. |
| Delayed Testing Due To New Issues | Medium | High | During testing, there is a good chance that some “new” defects may be identified and may become an issue that will take time to resolve.  There are defects that can be raised during testing because of unclear document specification. These defects can yield to an issue that will need time to be resolved.  If these issues become showstoppers, it will greatly impact on the overall project schedule.  If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution. |

# TEST CASES

**Test case ID**: 1.

**Test case**: check saving data.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Preconditions | Test steps | Test data | Expected result |
| 1.a | High | Check if data is saved to the file based on data from simulation. | Program is launched.  User is logged in.  Simulation is running. | 1. Click Save button. 2. Enter file name. 3. Select location where to save file. 4. Click Save button. | Data from running simulation. | File with data from running simulation. |

**Test case ID:** 2.

**Test case:** check loading data.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Preconditions | Test steps | Test data | Expected result |
| 2.a | High | Check if simulation is running based on data from selected file. | Program is launched.  User is logged in.  Simulation is running. | 1. Click Load button. 2. Select desired file with data from a previous simulation. 3. Click Open. | Data from selected file. | Simulation with data from selected file. |

**Test case ID:** 3.

**Test case:** check weather manipulation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Preconditions | Test steps | Test data | Expected result |
| 3.a | High | Check if weather manipulation influences simulation. | Program is launched.  User is logged in.  Simulation is running. | 1. Slide weather condition bar until desired weather is selected. | Data from running simulation. | File with data from running simulation. |

**Test case ID:** 4.

**Test case:** check checkpoint manipulation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Precondition | Test steps | Test data | Expected result |
| 4.a | Medium | Check if new checkpoint is added. | Program is launched.  User is logged in.  Simulation is running. | 1. Select checkpoint edit mode. 2. Click on the intersection point of the grid overlaying the map. 3. Select exit checkpoint mode. | Intersection point: B2. | A checkpoint is added at the location of the B2 point; the program reroutes the air traffic accordingly. |
| 4.b | Medium | Check if selected checkpoint is deleted. | Program is launched.  User is logged in.  Simulation is running.  A checkpoint exists at location B2. | 1. Select checkpoint edit mode. 2. Click on the already existing checkpoint. 3. Select exit checkpoint mode. | Location: B2. | The checkpoint at location B2 disappears; the simulation reroutes the air traffic accordingly. |

**Test case ID:** 5.

**Test case:** check add airplanes to the air space.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Precondition | Test steps | Test data | Expected result |
| 5.a | Medium | Check if airplanes are added to the airspace. | Program is launched.  User is logged in.  Simulation is running. | 1. Select add new airplanes. 2. Set airplane’s attributes. 3. Click Release. | Incoming direction: 160.  Bearing: 340.  Altitude: x (this will be specified when we find out what altitude will be appropriate for the early stages of the airport approach).  Speed: y (same as with the altitude). | An airplane enters the airspace from direction 160. It is flying at altitude x and moving with speed y. It’s current bearing is 340. |

**Test case ID:** 6.

**Test case:** check login functionality.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Severity | Test scenario | Precondition | Test steps | Test data | Expected result |
| 6.a | Critical | Check response on entering valid credentials. | Program is launched. | 1. Enter credentials. 2. Click Login. | Username: usrnm.  Password: pswrd. | Login is successful, and user is redirected to the simulation screen. |
| 6.b | Critical | Check response on entering invalid credentials. | Program is launched. | 1. Enter wrong credentials. 2. Click Login. | Invalid credentials | Login is unsuccessful, and error message is displayed. |

# APPROVALS

The Names and Roles of all persons who must approve this plan.

|  |  |
| --- | --- |
| **Signature:** |  |
| **Name:** |  |
| **Role:** |  |
| **Date:** |  |

|  |  |
| --- | --- |
| **Signature:** |  |
| **Name:** |  |
| **Role:** |  |
| **Date:** |  |