Test Plan (Wack a Mole)

CISC. 3140

Prepared by:
QA Team
Mehmet Sahin

Version: 1.0
Created: 11/20/2021
Last Updated: 12/07/2021
Status: DRAFT

Revision and Signoff Sheet

Document History - To maintain a list of changes being made

Version	Date	Author	Description of Change
1	02/14/2014	Swati Seela	Draft
2	02/14/2014	Vijay Shinde	Draft - Reviewed

Approvers List - To track who has reviewed and signoff on the Test plan

Name	Role	Approver / Reviewer	Approval / Review Date

Reference Documents - Clearly mark the document used as an input to create the test plan

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

Table of Contents

1. INTRODUCTION

1.1. Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the Wack A Mole Game Version 10. 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.

1.2. Project Overview

The Whack-a-Prof software is a game designed in Javascript for the CISC 3140 course project.

Whack-a-Prof is a standalone software that is a variation on the traditional Whack-a-Mole game. The game is intended to run on any web browser that supports Javascript. The game is in a 3 by 3 grid arrangement where the Professor will pop up on the screens in which the user will then have to click on the Professor to “whack” him. Player has only 10 seconds to get as high of a score as possible. Mallet -- mouse The Professor pops up randomly and the score counter increases by one point when the Professor is hit. If the User hits a Student, the score will decrease by 1.

1.3. Audience

- Project team members perform tasks specified in this document, and provide input and recommendations on this document.
- Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.
- Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.
- Business analysts will provide their inputs on functional changes.

2. TEST STRATEGY

2.1. Test Objectives

The objective of the test is to verify that the functionality of Whack A Mole game 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.

2.2. 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.
- All the defects would come along with a snapshot JPEG format
- The Test Team will be provided with access to Test environment via VPN connectivity
- The Test Team assumes all necessary inputs required during Test design and execution will be supported by Development/BUSINESS ANALYSTs appropriately.
- Test case design activities will be performed by QA Group
- Test environment and preparation activities will be owned by Dev Team
- The project will provide test planning, test design and test execution support
- Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.
- The system will be treated as a black box; if the information shows correctly online and in the reports, it will be assumed that the database is working properly.

Functional Testing

- During Functional testing, testing team will use preloaded data which is available on the system at the time of execution
- The Test Team will be perform Functional testing only on Whack A Mole game

2.3. 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.

2.4. Data Approach

- In functional testing, Whack a Mole game will contain pre-loaded test data and which is used for testing activities.

2.5. Scope and Levels of Testing

2.5.1. Exploratory

PURPOSE: the purpose of this test is to make sure critical defects are removed before the next levels of testing can start.

SCOPE: First level navigation, dealer and admin modules

TESTERS: QA team.

METHOD: this exploratory testing is carried out in the application without any test scripts and documentation

TIMING: at the beginning of each cycle.

2.5.2. Functional Test

PURPOSE: Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

Scope: The below excel sheet details about the scope of Functional test. Note: The scope is high level due to changes in the requirement.

TESTERS: QA Team.

METHOD: The test will be performed according to Functional scripts, which are stored in HP ALM.

TIMING: after Exploratory test is completed.

TEST ACCEPTANCE CRITERIA

1. Approved Functional Specification document, Use case documents must be available prior to start of Test design phase.
2. Test cases approved and signed-off prior to start of Test execution
3. Development completed, unit tested with pass status and results shared to Testing team to avoid duplicate defects
4. Test environment with application installed, configured and ready to use state

Sign-off	Readiness
•Approved Functional Specification Document •Approved Use cases •Approved Test cases	•Development completed & unit tested •Application deployed and system ready for testing on Test environment •Production like data is available to test all functionalities. •Defect fixes planned based on Defect triage (Unit Testing) and evaluation criteria

TEST DELIVERABLES

S.No.	Deliverable Name	Author	Reviewer
1.	Test Plan	Test Lead	Project Manager/ Business Analyst's
2.	Functional Test Cases	Test Team	Business Analyst's Sign off
3.	Logging Defects in HP ALM	Test Team	Test Lead/ Programming Lead(Vijay)
(4.	Daily/weekly status report	Test Team/ Test Lead	Test Lead/ Project Manager
5.	Test Closure report	Test Lead	Project Manager

MILESTONE LIST

The milestone list is tentative and may change due to below reasons

a) Any issues in the System environment readiness
b) Any change in scope/addition in scope
c) Any other dependency that impacts efforts and timelines

Testing generally is not carried out in one cycle. Based on the testing scope, we can estimate how much time it takes and establish the time lines as you can see in the below embedded excel sheet.

DFRT Execution Cycle.xlsx

2.5.3. User Acceptance Test (UAT)

PURPOSE: this test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.

TESTERS: the UAT is performed by the end users (L1, L2 and L3).

METHOD: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts. Test team write the UAT test cases based on the inputs from End user (L1,L2 and L3 users) and Business Analyst's.

TIMING: After all other levels of testing (Exploratory and Functional) are done. Only after this test is completed the product can be released to production.

TEST DELIVERABLES

S.No.	Deliverable Name	Author	Reviewer
1.	UAT Test Cases	Test Team	Business Analyst's Sign off

3. EXECUTION STRATEGY

3.1. 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 and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final "go-no go" decision.
- 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.

Exit Criteria	Test Team	Technical Team	Notes
100% Test Scripts executed			
95% pass rate of Test Scripts			

No open Critical and High severity defects			
95% of Medium severity defects have been closed			
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 HP ALM			

 

3.2. Test Cycles

- There will be two cycles for functional testing. Each cycle will execute all the scripts .
- The objective of the first cycle is to identify any blocking, critical defects, and most of the high defects. It is expected to use some work-around in order to get to all the scripts.
- The objective of the second cycle is to identify remaining high and medium defects, remove the work-around from the first cycle, correct gaps in the scripts and obtain performance results.

- UAT test will consist of one cycle.

3.3. Validation and Defect Management

Defects found during the Testing will be categorized according to the bug-reporting tool “Mercury HP ALM” and the categories are:

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.
