KASINTU

Web Collection Game

Test Strategy
Semester 3 - Individual Project

Airell Rasendriya Bachtiar
airell.bachtiar@student.fontys.nl

Table of Contents

1. Version

Version	Date	Description
0.1	12-04-2022	Make test strategy document
0.2	14-06-2022	Updated test strategy document
1.0	16-06-2022	Version 1.0 of test strategy document

2. Scope

This document will describe how tests will work for Kasintu, a web application game. Tests will cover the functionality of the application that are describe in project plan, in product backlog. This document will be reviewed and approved by the developer and the stakeholders.

3. Test Approach

This test will have 2 kind of approaches, automatic unit tests and user acceptance test. Unit tests are within the back-end code and will can run automatically to check the back-end functionality, front end functionality, and end to end tests. User acceptance test is conducted to voluntary tester to test the user interface of the web application.

Beside testing from unit tests and user acceptance tests, SonarQube and JaCoCo are used to check the quality of the code and test coverage.

4. Testing Tools

Junit are used for unit testing and fake MySQL database is used for realistic testing environment and it used for backend testing. For the presentation layer or user interface, tests are conducted using user acceptance test, frontend testing, and end to end tests using Cypress

5. Release Control

Test will be conducted based on release version from Git in main branch.

6. Risk Analysis

Risk	Prevention Action	Mitigation Action	Probability
No voluntary test for user acceptance test.	Ask friends or other people to take part for a short test.	Fill in the test by yourself.	60%
Code changes majorly that makes unit test unusable.	Plan on what output or parameters should present. Make a better unit test without many dependencies.	Change the unit test to better suited the code.	70%