November 12, 2023

Dominik Subocz

HND SOFTWARE DEV (YEAR2)

TESTING PLAN, tests & results

COIN FLIP APP

Contents

[Purpose of this Document 2](#_Toc151622799)

[Objective 2](#_Toc151622800)

[Test Scope 2](#_Toc151622801)

[Test Strategy 2](#_Toc151622802)

[Functional Testing 2](#_Toc151622803)

[Usability Testing 3](#_Toc151622804)

[Compatibility Testing 3](#_Toc151622805)

[Unit Testing 3](#_Toc151622806)

[Scenario Testing 4](#_Toc151622807)

[Tests & Results 4](#_Toc151622808)

# Purpose of this Document

The test document serves as a comprehensive guide to evaluate the functionality, usability, and performance of the Coin Flip application.

The document provides a detailed test plan, covering various aspects such as navigation, menu clarity, error handling and overall functionality. It aims to provide a structured approach to testing, ensuring that the application meets user expectations, runs smoothly, and deals effectively with potential problems.

This document only contains the test plan, the actual tests are documented and located in a separate folder in the documents folder.

Through a systematic testing strategy, this document aims to ensure a robust and user-friendly experience for those using the Coin Flip app.

# Objective

The main objective of this test plan is to test and ensure the functionality and reliability of the Coin Flip app so that it works correctly, and the user has a flawless experience.

# Test Scope

The tests will cover the basic functionality of the application, including coin flipping, guessing mode and user interface and responsiveness.

The tests will also cover the scoring system to check that there are no logic errors and that the score is correct.

# Test Strategy

### White-box testing

The testing strategy for the Coin Flip App mostly focus on white-box testing. In this testing approach, the tester has knowledge of the application’s architecture, scrutinizing individual lines of code, methods and pathways. By understanding the intricacies of software’s logic and structure, the main objective is to uncover any hidden bugs, logical errors and other vulnerabilities. This process allows us to ensure that each component performs as intended and that the different parts of the system interact seamlessly.

### Black-box testing

However, we also recognize the importance of black-box testing, where external perspectives come into play. This involves seeking feedback from friends, fellow students, and lecturers who engage with the application without delving into its underlying code. Their insights provide valuable external viewpoints, helping evaluate the user experience, functionality, and overall usability of the application. This dual approach ensures a comprehensive evaluation, combining the meticulous examination of code internals with real-world user perspectives.

The test strategy will include the following types of testing:

## Functional Testing

This test will focus on verifying that the coin toss button works as expected. Confirming that the application displays the correct output and correctly increments the heads and tails score.

The functionality of the menu and navigation will also be tested, as well as the function of the guess mode; to check that the correct message and sound is played when guessing.

Functional testing is crucial to ensure that the core functions of the Coin Flip app work as intended. This includes verifying the operation of the coin flip button, confirming the accurate display of results, and ensuring the correct score increment for heads and tails. Functional testing also includes menu navigation and correct execution of the Guess Mode function, ensuring a reliable and accurate user experience.

## Usability Testing

This test will focus solely on evaluating the user interface. This test will evaluate the user interface for clarity and ease of use and ensure that the application is compliant with the UWP design guidelines.

Usability testing focuses on the user interface, assessing its clarity and ease of use. This is essential to ensure that the Coin Flip app complies with UWP design guidelines, providing users with an intuitive and visually appealing experience. Usability testing identifies areas for improvement in the interface, increasing overall user satisfaction and engagement.

## Compatibility Testing

This test will focus on checking that the app works correctly on different UWP devices. It will check compatibility with different screen resolutions.

Compatibility testing is essential to check that the Coin Flip app runs smoothly on different UWP devices and screen resolutions. This ensures a consistent user experience, regardless of the device being used. By performing compatibility tests, potential issues related to different screen sizes and resolutions can be identified and resolved, ensuring universal usability.

## Unit Testing

The unit testing phase focuses on the systematic evaluation of individual components and functions of the Coin Flip application.

This includes rigorous testing of core functions such as the coin flip mechanism, score calculation and data handling.

Unit tests verify that each method and class work as expected, ensuring accurate results during coin flips, correct recording of historical data and seamless integration of audio and visual elements.

Unit testing focuses on isolating and evaluating individual components and functions within the Coin Flip application. This rigorous testing approach ensures that each method and class work as expected, contributing to the overall stability and quality of the application. Unit tests will cover core functionality, including the coin flip mechanism, score calculation, data handling and audiovisual integration, addressing potential bugs and edge cases.

In addition, the unit tests will cover edge cases and potential error scenarios to ensure the reliability and resilience of the application under different conditions. The aim is to isolate and evaluate each unit of code to verify its correctness, performance and resilience, contributing to the overall stability and quality of the Coin Flip application.

## Scenario Testing

Scenario testing involves systematically evaluating the performance of an application in different user-driven scenarios.

Each scenario represents a specific use case, ensuring that the application meets user expectations and performs correctly in different situations.

For example, scenarios might include simple coin tosses, adjusting coin types, changing the duration of tosses, and trying out the Guess Mode feature.

By systematically executing these scenarios, the testing process aims to identify potential problems such as incorrect coin results, application crashes or unexpected behaviour.

Additionally, scenario testing allows validation of user interactions, ensuring a smooth and intuitive experience.

Scenario testing involves simulating real-world user interactions to evaluate the performance of the Coin Flip application in different situations. Each scenario represents a specific use case, such as simple coin flips, adjusting coin types and testing the Guess Mode feature. This approach identifies potential problems, such as incorrect results or unexpected behaviour, ensuring a robust and user-friendly application. Scenario testing also checks user interactions, contributing to a smooth and intuitive user experience.

# Tests & Results