

Team Gravity





Project for Scale Focus Sprint

Bletchley Game

### Test plan



Contents

[**Test plan** 0](file:///C:\Users\HP\Desktop\Test%20Plan.docx#_Toc61308016)

[1. Introduction 2](#_Toc61308017)

[2. Entry and exit criteria 2](#_Toc61308018)

[2.1 Entry criteria 2](#_Toc61308019)

[2.2 Exit criteria 2](#_Toc61308020)

[3. Objectives and tasks 2](#_Toc61308021)

[3.1 Objectives 2](#_Toc61308022)

[3.2 Tasks 2](#_Toc61308023)

[4. Scope 3](#_Toc61308024)

[5. Approach 3](#_Toc61308025)

[6. Testing process 3](#_Toc61308026)

[6.1 Test deliverables 3](#_Toc61308027)

[6.2 Resources 3](#_Toc61308028)

# Introduction

This is a document which provides a plan for testing the Bletchley Game for the Scale Focus Sprint Project. Our game is a mini game in the form of a console application. It is suitable for people of all ages and does not require any specific skills to begin with when playing.

# Entry and exit criteria

## Entry criteria

* Access to the application is provided
* Code development has been paused until QA testing is finished

## Exit criteria

* Most test cases are passed successfully
* There are no critical issues which ruin the playing experience
* There is no more time left

# Objectives and tasks

## Objectives

* The objectives of this testing process are to provide verification that at least 90% of the functionalities of the application are suitable for use and do not have any major bugs or issues.

## Tasks

* The foremost tasks that must be completed to successfully test the program and assure its bug free status will be:
  + Defining the most important elements to be tested
  + Designing the various test cases
  + Performing manual testing
  + Performing automated testing
  + Preparing test and bug reports

# Scope

* Manual testing
* Game interface
* Menu options
* Data input
  + - Verify that user data can be entered
    - Verify that different error messages are displayed for different occasions
    - Verify that correct values by requirements are considered acceptable by the code
* Automated testing
* Basic functions for user input
* Game function algorithms

# Approach

* Here is a description of the techniques used to test the application
  + Functional testing
  + Usability testing
  + Unit testing
* Approach to automation
  + Unit testing for C++ with Native Unit Test Projects from Visual Studio

# Testing process

## Test deliverables

* Excel file with description of the test suits filled with different test cases
* Test plan

## Resources

* Microsoft Unit Testing Framework for C++ for testing some of the functions
* Excel as test case management tool
* GitHub for reporting problems