Engineering 1
Group Assessment 2
Software Testing Report Document

Cohort 2 Team 17
Gnocchi Games

Testing Methods

We used two different types of testing methodologies, white box testing and black box testing. These allowed us to cover all of the new features we added to the game, as objects and their interactions were covered by white box tests, while the different screens and features were covered by the white box tests.

 The white box tests were performed in the form of several unit tests. This allowed us to specifically test certain units of code, such as a method, separately in order to verify that they were working as intended by passing in a predefined input. Unit testing was appropriate because it helped discover errors in complex pieces of code by splitting them into smaller, more manageable pieces. These tests are fast and easy to write which is appropriate for our agile development method. Extensive unit testing ensured we had a high test coverage of our added code.

 The Black box tests were a series of manual tests as they had a wider scope than any other test. They allowed us to test different scenarios that could be encountered by the player, such as saving and loading, or interactions with different power ups. Black box testing was also the only way we could test things like whether the player won or lost the game.

Tests

The tests cover all of the additional features of the game; everything but the old features from the previous team have had a test created for it. These tests involved both a black box test, and a white box test.

 Those tests that didn't have a white box alternative were those who were testing for interactions between multiple systems of the game, such as tests for the displaying of the correct scenes for different scenarios. Or they were tests where the difference was impossible to perceive as they were relatively minute, such as features for some of the power ups (PowerUpRotation, and PowerUpDrag).

 Most tests passed, but a few failed. Most failures were related to unimplemented features, which were low priority and so it's not a major issue that they failed, as the only way to make them pass would be to implement the feature.

The specific test that failed was the manual test "Test for top 3".
> The reason it failed is explained in section 3's (The Change Report) Implementation section.

The Results of all tests are stored on the website at:
Unit Tests - https://frinksy.github.io/PixelBoatWebsite/reports/tests/test/index.html
Black Box Tests - https://frinksy.github.io/PixelBoatWebsite/test_videos.html
The unit tests are stored on the website at:

https://frinksy.github.io/PixelBoatWebsite/reports/tests/test/index.html

The black box tests are as follows:

The black box tests for UR_Collisions and UR_Awards:

Test for game over 1:
Click into game
Run into obstacles till you lose all your health
Check if game over screen appears

Test for game over 2:
Click into game
Finish game in 2nd place or lower
Check if game over screen appears

Test for game over 3:
Click into game
Don't qualify for final
Check if game over screen appears

Test for Top 3:
Click into game
Finish game
See what place you get

Test for Award Screen:
Click into game
Finish game in 1st place
Check if award screen appears

Black box tests for the UR_DIFFICULTY_LEVEL involves following the

Test for easy difficulty:
Click on "easy" button
Click on "Ready" button
Move forward until stamina is depleted and record time
-if takes more time for the bar to deplete than the Medium and hard difficulties test one passed
Wait until the stamina bar is full again and record time
-if takes less time for the bar to fill than the Medium and hard difficulties test two passed
Collide with an object and record approx percentage
-if takes less damage than the Medium and hard difficulties test three passed

Once both medium and hard tests have been ran, we compare these values with the values collected in the easy test to confirm that easy test has been passed

Test for medium difficulty:
Click "Medium" button at difficulty screen
Click "Ready" button on difficulty screen
Move forward until stamina is depleted and record time

-if takes less time to deplete the stamina bar that the easy difficulty test one passed

Wait until the stamina bar is full again and record time

-if takes more time for the bar to fill the easy difficulty test two passed

Collide with an object and record approx percentage

-if takes more damage than the easy difficulty test three passed

If all three tests have passed then Medium difficulty test passed

Test for Hard difficulty:

Click "Hard" button at difficulty screen

Click "Ready" button on difficulty screen

Move forward until stamina is depleted and record time

-if takes less time to deplete the stamina bar than Medium test one passed

Wait until the stamina bar is full again and record time

-if takes more time for the bar to fill than Medium difficulty test two passed

Collide with an object and record approx percentage

-if takes more damage than the Medium difficulty test three passed

If all three tests have passed then Hard difficulty test passed

Black box tests for the requirement UR_POWER_UP involves:

Test for Speed Power-Up:

Move into Speed Power-Up

- if boats maxspeed increases or is the same (if already max) test one passes

- if boats durability stays the same test two passes

- if both tests pass the speed Power-Up test passes

Test for Health Power-Up:

Move into Durability Power-Up

- if boats Durability or is the same (if already max) increases test one passes

- if boats Max Speed stays the same test two passes

- if both tests pass the Durability Power-Up test passes

Test for Energy Power-Up:

Move into Energy Power-Up

- if boats stamina increases or is the same (if already max) test one passes

- if boats durability stays the same test two passes

- if boats max speed stays the same test three passes

- if all tests pass the Energy Power-Up test passes

Black box tests for the requirement UR_SAVE involve:

Test Saving:

- Click into game

- Play for a bit
- Press escape
- Click Cancel
- Play for a bit
- Press escape
- Enter save name
- Click Load
- Select corresponding save
- Click Load
- If all steps performed correctly, and state is restored properly, test passes

Test Deleting a Save:
- Click into game
- Save game (press escape, enter name and save)
- Repeat above 2 steps 3 times
- Click Load (on main menu screen)
- Select save to delete
- Click delete
- Click back to menu
- Click back to Load
- Verify that save has been deleted
- If all steps performed correctly, test passes

The Results of all tests are stored on the website at:
Unit Tests - https://frinksy.github.io/PixelBoatWebsite/reports/tests/test/index.html
Black Box Tests - https://frinksy.github.io/PixelBoatWebsite/test_videos.html
The unit tests are stored on the website at:
https://frinksy.github.io/PixelBoatWebsite/reports/tests/test/index.html

Test Results - https://frinksy.github.io/PixelBoatWebsite/reports/tests/test/index.html
Test Coverage - https://frinksy.github.io/PixelBoatWebsite/reports/jacoco/test/html/index.html

Source code and jar file can be found through this link:
https://github.com/Frinksy/PixelBoat/releases/tag/v1.2