

Assessment due date: 12/01/26

Record of testing

23/11/25

Rosie:

- Downloaded all required software for testing and any additional libraries and plugins needed so far.
- Made detailed notes on testing based on lecture slides and readings.
- Created traceability matrix.
- Found tutorials for JUnit and Gradle and Mockito

24/11/25

Rosie:

- TimerSystem test created.

25/11/25

Rosie:

- Received errors on Visual Studio relation to JUnit, and cannot get VS to recognise the TimerTest file as a test.

29/11/25

Rosie:

- Drafted code for testing Collision System. Still cannot use JUnit.

30/11/25

Jude:

- Done System Timer Tests, Passes Github actions tests. Junit package reference added to gradle properties.
- Test for score drafted, build fails. Added new build.gradle in the core folder and referenced JUnit dependency
-

02/12/25

Jude:

Assessment due date: 12/01/26

- “Main-Score-Tests” branch added to GitHub includes full game and tests
- Written up in “Test Plans” document about each test for scores in main.
- Written up in “Test Plans” document about each tests relating to TimerSystem

Ishraan:

- Did some manual testing on the game to check the FR_RESUME functionality
- Started doing some manual testing on the game to check the FR_RESET functionality
-

03/12/25

Rosie:

- Updated tests for CollisionSystem to be more readable and functional - haven't uploaded yet as my codespace isn't compatible with the new alterations.
- Drafted tests for the map.
- Manual testing **to be done** for the movement and map.

06/12/25

Rosie:

- Completed manual test for map.
- Written up in 'Test Plans' about this test.
- Changed the requirement met for the collision testing to FR_MAP rather than FR_MOVEMENT.
- Changed the Traceability Matrix so that it is easier to read, and added some additional requirements that may be added.

07/12/25

Rosie:

- Got JUnit working and can now actually test any code I've written!!
- Created a branch on GitHub for the collision tests.
- Established the Mockito framework.
- Completed collision tests.

Sam:

- Implemented Toast System tests.

08/12/25

Rosie:

- Drafted Movement tests on a new GitHub branch
- Completed manual test for resume functionality

Assessment due date: 12/01/26

09/12/25

Sam:

- Doing some manual testing (in runtime) e.g. trying to find bugs through different ways the player could approach going through the game.
- Discovered that the time that the dean lectures you for has been updated to 30s instead of 50 but the message displayed on the screen.
- Interacting with several events in a short space of time can cause the messages displayed on the screen.
 - > This seems to be an issue with the bob message specifically.
 - > Fixed by having each line as a separate toast instead of using newline characters.

22/12/2025

Jude:

- At this point I have done testing for the events that have been updated/upgraded after taking on the project from the group we chose.
- All events tested have their own set of unit tests, SoAndSoTest.java.
- Events that have been tested now are:
 - Bookshelf moving event
 - Door out event
 - Lever for spikes event
- After creating the tests they'd all fail because there is, for some reason, a project nested inside the project within the project folder. I think this might be something to do with how it has been opened in VSCode maybe? Edited the batch file to change where gradlew.bat is expected to be and it works.
- Uploading to github, I didn't know how to upload in a specific place when vscode got itself in a muddle. Created a new branch instead and added it as a zip file. Will comment on the branch so people understand.
- Done unit tests for each event apart from the dean.

30/12/2025

Rosie:

- Created and now finalising movement tests
- Struggling with GitHub crashing and keeps removing the 'testing' tab

03/01/2026

Assessment due date: 12/01/26

05/01/2026

Sam:

- Testing while combining different versions of the game, I found that the event tests would crash when integrated into the version of the game with the audio system.
- This is because JUnit assumes that LibGDX is always running while JUnit tests are running, which is not true in this case.
- I have fixed this by editing the audio system class to add checks for null in its methods to prevent the tests from crashing due to this.

06/01/2026

Rosie:

- Amended the movement tests again to try and make VS code run the tests.
- Edited movement tests for more clarity, separating the movement case tests into individual up, down, left and right.
- Added helper methods.
- Tests failing due to Mockito not effectively mocking the player. After adding mocked players to each test, the tests ran and all passed (tests run using gradle commands rather than vs code's testing as it would not show up).
- Committed PlayerTest code changes to the repository.

10/01/2026

Jude:

- Wrote up the events tests descriptions in the testing phase 2 document
- Tested key and exit door event
- Tested level and spikes event
- Tested bookshelf and book event

11/01/2026

Rosie:

- Edited the collision test to include a helper method, and improved the clarity of the code.
- Added a few more tests to test class so that the overall coverage is higher

Jude:

- Finished off dean test, wrote up on "testing phase 2".
- Checked over invariance testing.
- Tested dean pathfinding. How the dean's path is reset after a cycle.
- Tested if the dean can catch the player when in close vicinity.
 - When player is invisible
 - When they are visible
- Finished of invariance testing, game state testing.
- Score system and bonus