Sprint 1 Plan Document

COMP20050

HEXOUST PROJECT

By GROUP 13:

Cian Latchford - 23201433

Federica Fucetola - 23922982

Laura González Calleja - 23713479

1. Sprint objective

The objective of this sprint is to create a functioning version of the start of the game called “Hexoust” that respects all the functional and non-functional software requirements for specification 1 (SR1, SR1.1, SR1.2, SR1.3). This is a turn based game that can be played by two people. Each player is represented by a color: blue or red. The goal is to display the layout of the board, ask the players to insert their names and the color they want to be, display the color and the sentence that indicates the current player’s turn, and allow the players to end the game at any point.

2. Technical information

- **Programming languages**: Java, JavaFX
- **Version control**: GitHub
- **Designing tools**: draw.io
- **Additional tools and programs**: IntelliJ and VS Code (Java IDEs), Google Drive and Word/Google Docs (for shared access and editing space to project’s documentation) and Trello (task and time management)

3. Sprint milestones

- Submission Software Architectural Design and Project Plan - **10 February** 2025 **SUBMITTED**
- Submission Sprint 1 - **23 February** 2025

4. Responsibilities

- Chairperson: Laura González Calleja
- Scribe: Federica Fucetola
- Timekeeper: Cian Latchford

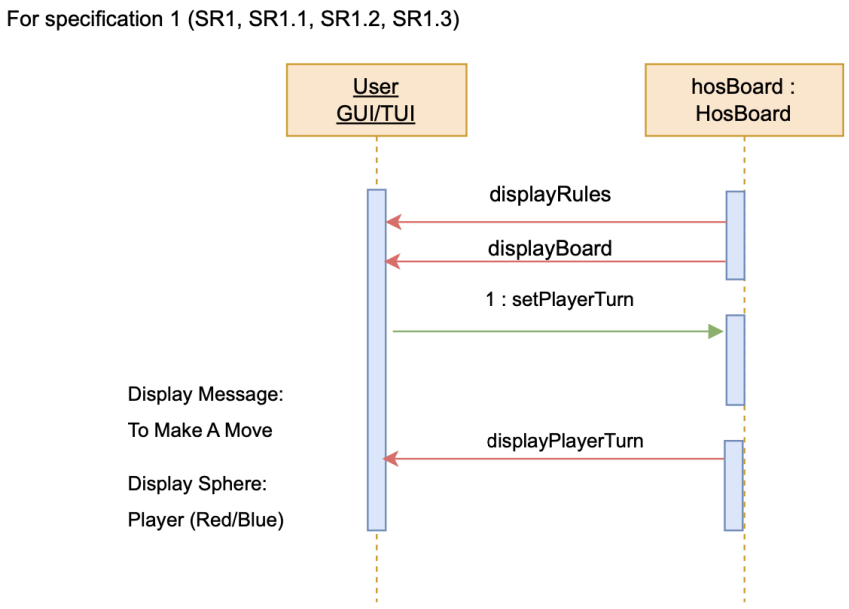
5. Communication Plan

- **Communication tools**: Whatsapp and emails for quick exchanges of messages, as well as Google Meet and in-person meetings.
- **Status reports**: Frequent updates on the project's progress, both online (shared Trello board for general milestones and individual boards for each member and their specific tasks) and in-person.

6. Development plan

1. **Sprint 1**:
    a. **Display board** (**SR1**): display the layout of the board - **Laura González Calleja**
    b. **Display turn** (**SR1.1, SR1.2, SR1.3**): display the color and the sentence that indicates the current player's turn - **Cian Latchford**
    c. **Exit game** (**SR1**): allow player to end the game at any point - **Laura González Calleja**
    d. **Player information** (**SR1.1, SR1.2, SR1.3**): ask the players to insert their names and the color they want to be - **Federica Fucetola**
    e. **Game rules**: display a screen with all the game's rules to the players before the game starts and before they enter their names and colors - **Federica Fucetola**

Following the previous Activity/Sequence Diagram for specification 1:



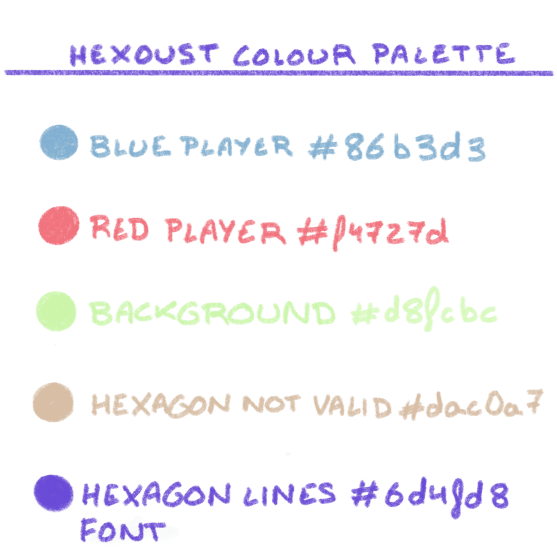
How It Works:

- When the game starts, the HosBoard initialises an **empty board**.
- The **RED player starts first**, indicated by a **RED coloured message** displaying "(PLAYER 1 NAME)'s Turn"
- The board is rendered, showing the available hexagons.

Diagram Implementation:

1. SetPlayerTurn is called in HosBoard to set the starting player (ask the name and colour they want to be).
2. DisplayPlayerTurn updates the GUI with a **RED colour** and a message.
3. The board is displayed to the player (displayBoard).

And the aesthetic guidelines we chose for the project:



7. Changes in the Sprint

- The “display turn” feature is just displaying the sentence that indicates which player has to play at the moment, in the player’s corresponding color. The colored sentences are substituting the colored sphere in the turn’s sentence to make the graphic interface of the Board less cluttered.
- All the features that were specified in the Architectural Sprint were implemented, only the player information display is not linked to the board display yet (as they use different interfaces). This will be done in the next Sprint.

8. Conclusion: success of the sprint

To guarantee the success of this sprint, the following guidelines should be followed:

- The sprint and project should be always completed within established dates
- In case of issues of any kind, communicate promptly with team leader and team members to fix the situation as soon as possible to avoid setbacks and bigger problems in the future
- Meet all the functional and non-functional requirements specified in the “*Software Architectural Design Document*” and in the more general outlines of the project given by the stakeholder
- Get a positive user (stakeholder) feedback