Technical Design Document (TDD)

[CRC cards](#_Toc97809518)

[Architecture](#_Toc97809519)

[Technology](#_Toc97809520)

[Sprint 1 Report](#_Toc97809521)

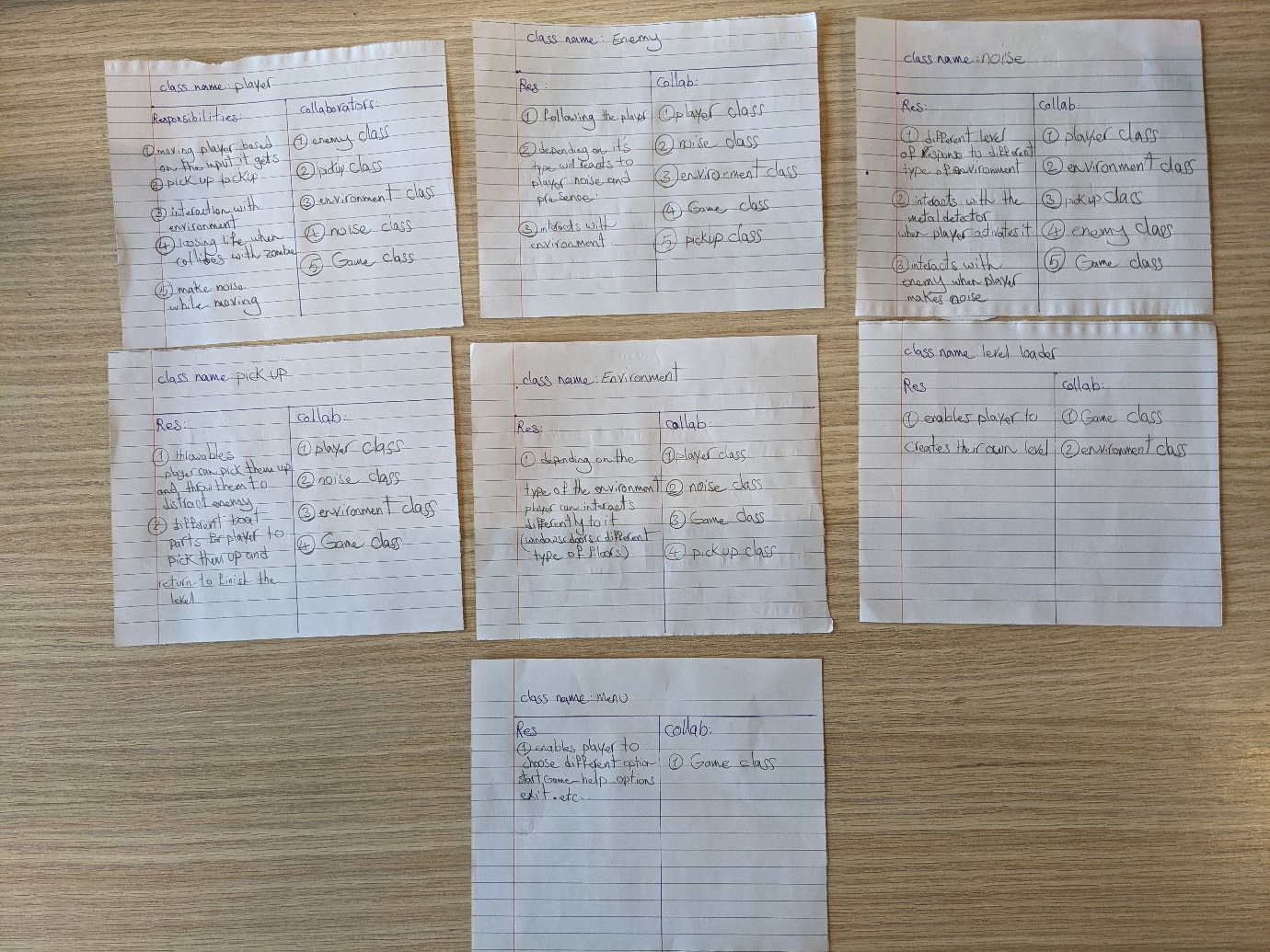
[Sprint 2 Report](#_Toc97809522)

[Sprint 3 Report](#_Toc97809523)

# 

# 

# CRC cards



# Architecture



# Technology

**Research**

**Begin by outlining the rationale for your technology choices (i.e. justify your choices) and any research you did in relation to support libraries or other aspects related to development. For example, if you picked one SFML GUI library over another, justify that decision. We suggest you are economical in your use of time in relation to the research element (e.g. 30 minutes max on any one particular aspect).**

* **JSON for Modern C++**

We needed a library to parse and create JSON files to store our level data (i.e.: tiles position, pre-computed UCS Pathfinding, …). We tried to find the best and easy to use one and we found this one, JSON for Modern C++, made by nlohmann which is the most famous JSON library used for C++.

* **tinyc2.h:**

I choose to use tiny2c because it was a pre-made library and it enabled us to easily detect a collision between enemy vision cone and player (circle). Cause we originally did a point to triangle collision check for it and it was not efficient.

**Installation**

* **JSON for Modern C++**

The library is integrated with a single header file which is called json.hpp and contains all the library. This is the simplest way to integrate this library.

Here is the link of the documentation and download:

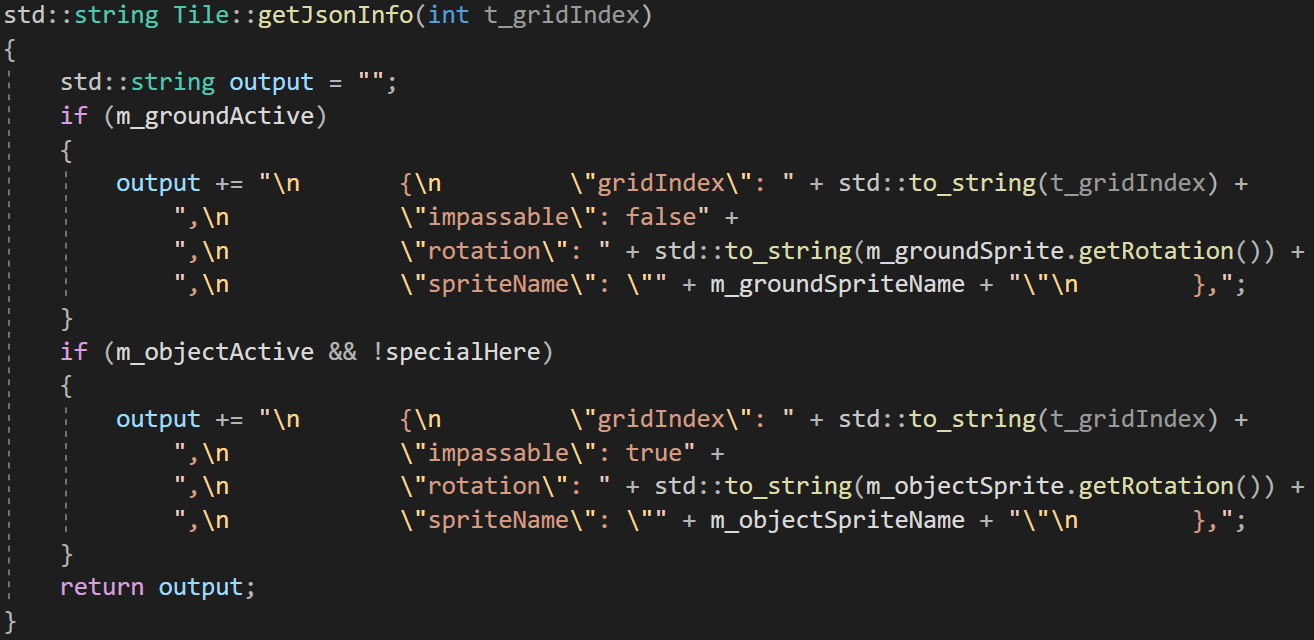
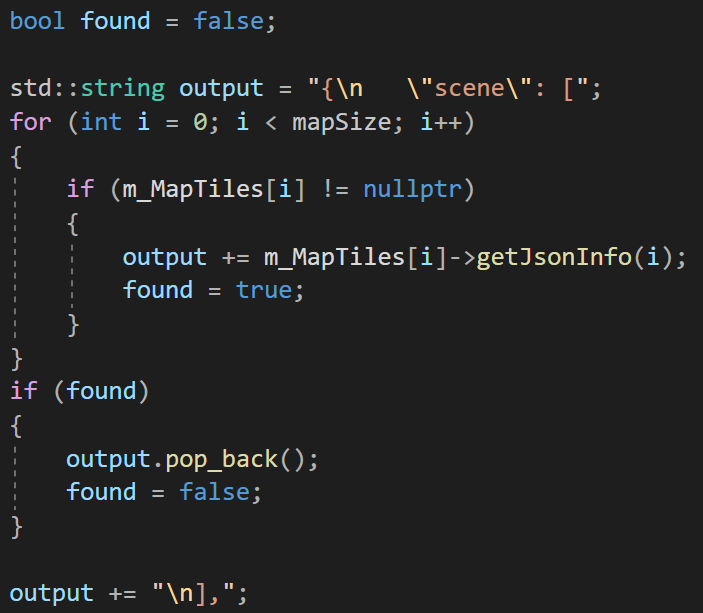
* <https://json.nlohmann.me/>
* <https://github.com/nlohmann/json>

**Technical achievement**

**This section should be completed at the end of the project by each team member. You should highlight one part of the project that you felt was particularly challenging but managed to complete. This could be a pathfinding algorithm, a rendering technique, a complex game mechanic etc. Include relevant code snippets in your discussion.**

Adrien:

Eoin: I was in charge of making the level builder, this required managing a lot of data and organising it as well as giving clear feedback to the player aswell as the best way to output that data to the json file, It took me a bit but once I had my code layout planned out it all seemed to fall into place



Masih:

Although most of the projects we did up until now were in SFML, I can say I learned a lot during this project. The use of Json file and reading data from it was very interesting and fun. I would say the most challenging part for me was the vision cone and the collision check I ended up using an external library for collision detection.

Code snippet is as follow:







# Sprint 1 Report

**Summary of planned work**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features and tasks** | **Time  Estimate** | **Time Actual** | **Team Member** | **Complete** |
| **Feature 1: Moving Player** |  |  |  |  |
| Task 1: Render a rectangle | 1 hr | 2hr | Masih | Yes |
| Task 2: Setup input handler | 30 min | 30 min | Masih | Yes |
| Task 3: Player movement | 1hr | 2hr | Adrien | Yes |
| Task 4: Improve movement | 1 hr | 1 hr | Eoin | Yes |
| **Feature 2: Enemy** |  |  |  |  |
| Task 1: Render the Enemy | 1 hr | 1hr | Masih | Yes |
| Task 2: Setup Collision with Player | 1 hr | 1 hr | Eoin | Yes |
| Task 3: creating enemy vision cone | 2 hr | 4hr | Masih | Yes |
| Task 4: detecting collision between player and enemy vision cone | 2 hr | 3hr | Masih | Yes |
| Task 5: Improve Movement | 30 min | 1 hr | Eoin | Yes |
| **Feature 3: Environment** |  |  |  |  |
| Task 1: Render the object | 30 min | 30 min | Eoin | Yes |
| Task 2: Setup Collision with player and enemies | 1 hr | 1 hr | Eoin | Yes |
| Task 3: Setup Constructor to for the level loader | 30 min | 30 min | Eoin | Yes |
| **Feature 4: Noise** |  |  |  |  |
| Task 1: Draw the Noise | 1 hr | 1 hr | Masih | Yes |
| Task 2: Allow Noise to be added | 1 hr | 1 hr | Eoin | Yes |
| Task 3: Make the Noise vary based on noise level | 1 hr | 2 hr | Eoin | Yes |
| Task 4: Setup a radius that detects nearby zombies based on noise level | 1 hr | 2 hr | Eoin | Yes |
| **Feature 5: Pickups** |  |  |  |  |
| Task 1: Draw Pickups on the screen | 30 min | 30 min | Eoin | Yes |
| Task 2: Setup Collision with Player | 1 hr | 1 hr | Eoin | Yes |
| Task 3: adding the sprite of pickups | 30 min | 1hr | Masih | Yes |
| Task 4: adding the key input for throwing the pickups | 30 min | 1hr | Masih | Yes |
| Task 5: adding the power bar | 30 min | 30 min | Masih | Yes |
| **Feature 6: Enemy Vision Cone** |  |  |  |  |
| Task 1: setting up the cone | 1 hr | 2 hr | Masih | Yes |
| Task 2: cone state changing | 30 min | 1 hr | Masih | Yes |
| Task 3: cone and player collision | 1 hr | 3 hr | Masih | Yes |
| **Feature 7: View/Camera** |  |  |  |  |
| Task 1: setting up the view | 2hr | 3hr | Adrien | Yes |
| Task 2: render element to the view | 30min | 1hr | Adrien | Yes |
| Task 3: implement sf::Drawable to all the game objects | 30 min | 30 min | Adrien | Yes |
| **Feature 8: Animation** |  |  |  |  |
| Task 1: create the Animation class | 1hr | 3hr | Adrien | Yes |
| Task 2: scale all the sprites | 3h | 3hr | Adrien | Yes |
| Task 3: create sprite sheets | 30min | 1hr | Adrien | Yes |
| Task 4: create the animations | 1hr | 1hr | Adrien | Yes |
| **Feature 10: Menu & UI** | | | | |
| Task 1: setting up the menu buttons | 20 min | 20 min | Masih | Yes |
| Task 2: creating and setting up the menu | 1 hr | 1hr | Masih | Yes |
| Task 3: adding visual improvements | 30 min | 30 min | Masih | Yes |

# Sprint 2 Report

**Summary of planned work**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Features and tasks** | | **Time  Estimate** | | **Time Actual** | **Team Member** | | **Complete** |
| **Feature 1: Moving Player** | | | | | | | |
| Task 1: fix player state | | 30min | | 2hr | Adrien | | Yes |
| **Feature 8: Animation** | | | | | | | |
| Task 1: split the throwing animation in two | | 1hr | | 1hr | Adrien | | Yes |
| **Feature 9: Level Builder** | | | | | | | |
| Task 1: basic script to generate the environment | | 1hr | | 2hr | Adrien | | Yes |
| Task 2: Draw Layout | | 1 hr | | 1 hr | Eoin | | Yes |
| Task 3: Make Button Object | | 30 mins | | 30 mins | Eoin | | Yes |
| Task 4: Generate and Draw Buttons | | 30 mins | | 30 mins | Eoin | | Yes |
| Task 5: Setup Menu Interaction | | 30 mins | | 30 mins | Eoin | | Yes |
| Task 6: Setup Scrolling on menu | | 1 hr | | 3 hrs | Eoin | | Yes |
| Task 7: setup interaction with map | | 1 hr | | 2 hrs | Eoin | | Yes |
| **Feature 10: Menu & UI** | | | | | | | |
| Task1: Setting up the HUD | 1hr | | 1hr | | | Masih | Yes |
| Task2: Finding and Setting Sprites | 2hr | | 2hr | | | Masih | Yes |
| Task3: interaction with pickups | 1hr | | 1hr | | | Masih | Yes |
| **Feature 11: Sound** |  | |  | | |  |  |
| Task 1: Setup SoundHolder | 2 hr | | 3hr | | | Adrien | Yes |
| Task 2: Find sounds | 4hr | | 4hr | | | Adrien | Yes |
| Task 3: Edit sounds | 1hr | | 2hr | | | Adrien | Yes |
| Task 4: Add and sync footsteps sounds with animation | 1hr | | 1hr | | | Adrien | Yes |
| Task 5: update footstep sounds volume depending of distance to player | 1hr | | 2hr | | | Adrien |  |
| Task 6: Add ambient sounds/music | 3hr | | 1hr | | | Adrien/Masih | No |

# Sprint 3 Report

**Summary of planned work**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features and tasks** | **Time  Estimate** | **Time Actual** | **Team Member** | **Complete** |
| **Feature 2: Enemy** | | | | |
| Task 1: A\* pathfinding implemented |  |  |  |  |
| Task 2: UCS pathfinding implemented |  |  |  |  |
| Task 3: Enemy reacts to noise |  |  |  |  |
| Task 4: Path is drawn in Debug mode | 1 hr | 1 hr | Eoin | Yes |
| Task 5: Impassable objects are highlighted in Debug mode | 1 hr | 1 hr | Eoin | Yes |
| **Feature 4: Noise** | | | | |
| Task 1: Ground varies in Noise | 1 hr | 1 hr | Eoin | Yes |
| **Feature 5: Pickups** | | | | |
| Task 1: power bar | 2 hr | 2 hr | Masih | Yes |
| Task 2: pick up throw | 2 hr | 5 hr | Masih | Yes |
| Task 3: bottle break (particle system) | 2 hr | 2 hr | Masih | Yes |
| Task 4: noise creation | 1 hr | 1 hr | Eoin/Masih | Yes |
| **Feature 9: Level Builder** | | | | |
| Task 1: Allow Player to rotate objects | 2 hr | 2 hr | Eoin | Yes |
| Task 2: Allow player to delete objects | 30 min | 30 min | Eoin | Yes |
| Task 3: Add special objects | 30 min | 30 min | Eoin | Yes |
| Task 4: Add parsing of special objects in game | 1 hr | 1 hr | Eoin | Yes |
| Task 5: Exporting level information to a json file | 1 hr | 1 hr | Eoin | Yes |
| Task 6: Check Whether level already exists | 30 min | 30 min | Eoin | Yes |
| **Feature 10: Menu & UI** | | | | |
| Task 1: adding pause menu | 2 hr | 3 hr | Masih | Yes |
| Task 2: adding help menu | 3 hr | 5 hr | Masih | Yes |
| Task 3: adding load screen | 30 mins | 30 min | Eoin | Yes |
| Task 4: Dynamically load levels | 1 hr | 1 hr | Eoin | Yes |
| Task 5: When last level is complete trigger win | 30 min | 30 min | Eoin | Yes |
| **Feature 11: Sound** | | | | |
|  |  |  |  |  |