

Team 5

Nathan Dodson, Tucker McCulloch, Logan Schmalz, Tristan McCulloch

Initial Architecture Document

1. Project Name:

Rustymon

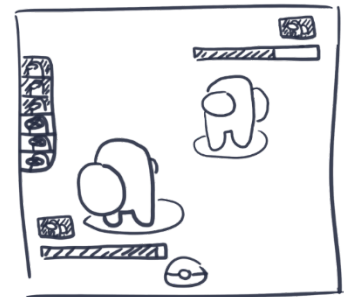
2. Project Synopsis:

An RPG in which you quest to build a team of fighting creatures and become the best there ever was.

3. Architecture:

Written in Rust, Rustymon will be a third-person, top-down role-playing game like Pokémon. The plan is to take the basic elements of Pokémon games, but not to be entirely married to the smaller details of the franchise in general or individual games specifically. According to the official documentation, “SDL2 (Simple DirectMedia Layer) is a cross-platform development library designed to provide low-level access to audio, keyboard, mouse, joystick, and graphics hardware via OpenGL and Direct3D.” There is an SDL2 crate for Rust that we will be utilizing in order to implement this access into Rustymon for window creation, rendering, and player controls. In essence, SDL2 will be the foundation for our gameplay engine. The basic layout of the game will be a set of tile maps that the player moves through one tile at a time. This is preferred over a pixel-based movement/hitbox system for ease of asset development and elegance of gameplay. The tile maps will be of a fixed column/row length (e.g. 15x10 tiles) and will scale up and down with screen resolution as opposed to growing the visible tile area. The overworld tile map will be a seamless and continuous area except for “indoor areas,” where each indoor area tile map will be its own distinct area. The player will move between the overworld and the indoor areas through specific tile triggers (these triggers might correspond with doors, cave entrances, forest entrances, etc.) linked with correct locations to go back and forth. Menus, player movement, and battle screens will be navigable with the arrow keys, certain key binds, and optionally the mouse.

Throughout exploring the world, players will be forced to battle wild creatures and other characters. A simple example of the battle screen layout might be what is depicted here to the right. Health points, user and enemy creatures, and menu items will be present. Players will have access to their personal items and health/creature-catching materials in the bag menu. The attack menu will present the player with available moves to use against the enemy creature. Battlers will take turns to make a move; that could be an attack against the other creature, a self-healing move, or the use of an item to heal or catch other creatures.



Battles will conclude when either battler no longer has any creatures able to fight. In the case the player loses, they will respawn at the nearest healing center.



The overworld comprises two main islands with varied geographic features, from forests to beach-side towns, to mountains and caves. There will be story-based restrictions to certain areas and as the game progresses user access to those areas will increase. As we create assets for the game a detailed tile map of the two islands will be developed. The depiction to the left is simply an artistic generalization of what we want the world to be.

Team 5

Nathan Dodson, Tucker McCulloch, Logan Schmalz, Tristan McCulloch

Pictured on the right is an example of what the start screen would look like. It will feature one of the creatures from the game and be on a simple looped animation. The game will launch on this screen, and when the player presses the 'any' key, it will take them to a menu that allows them to start a new game, start from save, or change settings.

By the project deadline, we hope to have a fully playable game with a story and progression mechanics. Ideally, we want to integrate multiple levels of multiplayer support so multiple players can trade, battle, and even interact in the overworld. Rust should enable us to effectively implement all of the features we want.

