Parchment Game Engine

Team 12 - Product Backlog

Jacob Brown, Raymond Xie, Xiaoyan "Larry" Li, Jack Locascio, Drew Hatfield, Josh Gerber

Problem Statement

Creating games is a process many people want to get involved with, but it's often difficult to find an engine that both suits the creator's needs and doesn't have a steep learning curve. Parchment recognizes these issues and provides an intuitive GUI that allows users to seamlessly edit various facets of their game and build their own 2D Zelda-like experience from the ground up, offering users a map editor, sprite editor, item editor, support for user code, a robust file system, and various other features to assist in the game making process. By streamlining the game development process, Parchment eliminates the large time investment it takes for new users to become acquainted with utilizing a new engine and equips them with the necessary resources to create their own 2D bird's eye view game with little to no prior experience.

Background Info:

Audience

The Parchment Engine is geared towards developers without much experience in game development. Our engine aims to provide the developer with a toolset that allows them to easily build a game from scratch and serves as an introduction to game development with minimal coding experience required. However, the Parchment Engine also allows for advanced functionality aimed at experienced developers to allow for custom scripts to run within the engine, allowing the engine to be expandable beyond its base usability.

Similar Platforms

There are several similar existing 2D-focused game engines such as Pixelbox.js and GameMaker Studio. Pixelbox uses a JavaScript library to eliminate tedious aspects of HTML5 game development and GameMaker is a highly robust engine that allows users to make 2D games for nearly any platform. Additionally, there are a multitude of other non-2D-focused game engines such as Unity and Unreal Engine, both which are larger and more complex in scope.

Limitations

While many existing game engines are useful, there is still a steep learning curve to many of them, and we believe that aspiring game developers should be able to pick up an existing game engine and dive right into development with the aid of only a few tutorials or, better yet, just using the engine itself. Parchment aims to create a more intuitive game engine with a clean design that has no entry barriers and will equip any aspiring developer with the tools required to create a 2D game. We believe less time should be spent learning how to use a game engine and more time should be spent towards development, and our vision aims to make this concept a reality.

Functional Requirements:

1. As a user, I would like to open and close Parchment natively on a Windows OS so that I can use the game engine.
2. As a user, I would like to resize the Parchment window responsively so that I can use it on any size of monitor.
3. As a user, I would like to rearrange and resize docked views so that Parchment can better suit my workflow.
4. As a user, I would like to view a Parchment splash screen so that I can see startup progress.
5. As a user, I would like to change Parchment settings so that I can configure it to my preference.
6. As a user, I would like to change the UI appearance (including themes and fonts) so that I can meet my accessibility needs.
7. As a user, I would like to view in-application tutorials or documentation so that I can learn the features.
8. As a user, I would like to name/create a new game project so that I can start developing.
9. As a user, I would like to view my existing list of projects so that I can continue developing.
10. As a user, I would like to save a game project so that my work progress is kept.
11. As a user, I would like to delete an existing game project and associated files so that I can manage projects.
12. As a user, I would like to change the name of an existing game project so that I have flexibility.
13. As a user, I would like to initialize a game instance using Parchment so that I can test the game.
14. As a user, I would like to have my game assets loaded and rendered by Parchment so that I can focus on game development.
15. As a user, I would like to be able to view all assets associated with a game project so that I can manage them.
16. As a user, I would like to access a sprite editor view so that I can edit sprites.
17. As a sprite editor, I would like to scroll through or search all sprites I have made so that I can easily manage them.
18. As a sprite editor, I would like to edit existing sprites so that I can iterate on sprite design.
19. As a sprite editor, I would like to create simple animations so that my sprites have motion.
20. As a sprite editor, I would like to place and delete sprites so that I can create game scenes.
21. As a sprite editor, I would like to be able to undo and redo actions so that I can fix mistakes.
22. As a sprite editor, I would like to use basic pixel manipulation tools (pencil, fill, eraser, etc) so that I can edit sprites.

23. As a sprite editor, I would like to change the pixel color and use a color picker so that I can create colorful sprites.
24. As a sprite editor, I would like to use selection tools so that I can move, copy, and paste sprites in the editor.
25. As a sprite editor, I would like to import an image for a sprite so that I can bring in outside assets.
26. As a sprite editor, I would like to use multiple layers so that I can create more complex sprites. (time permitting)
27. As a user, I would like to map controls to in-game sprites so that I can control their movement.
28. As a user, I would like to access an item editor view so that I can edit items.
29. As an item editor, I would like to manage item properties so that I can have different item behaviors.
30. As an item editor, I would like to add events to items so that I can have reactive item behavior.
31. As an item editor, I would like to place and delete items on the map so that I can create game scenes.
32. As an item editor, I would like to move an item in the editor.
33. As an item editor, I would like to edit existing items (on the map or otherwise) so that I can iterate on item design.
34. As a user, I would like to be able to define player interactions with objects so that I can implement interactivity.
35. As a user, I would like to access a map editor view so that I can edit maps.
36. As a map editor, I would like to import an existing image or map so that I can bring in outside assets.
37. As a map editor, I would like to delete an existing map so that I can manage maps.
38. As a map editor, I would like to edit an existing map so that I can iterate on map design.
39. As a map editor, I would like to edit map properties such as size so that I can have different map types.
40. As a user, I would like to change the game objective so that I can make different games.
41. As a user, I would like to end the game if an objective is reached so that the game can be beat.
42. As a user, I would like to create a menu screen so that the player can start the game.
43. As a user, I would like to add event listeners to keys so that the player can trigger game events.
44. As a user, I would like to maintain state related to players so that I can implement inventory or score systems.
45. As a user, I would like to change settings for player controls so that I can have custom controls.
46. As a user, I would like to bind different mouse controls so that I can make mouse controlled games.
47. As a user, I would like to import audio for items/sprites so that I can have specific audio tied to game objects.
48. As a user, I would like to import audio for music so that my game can have a soundtrack.

49. As a user, I would like to view and delete audio so that I can manage audio assets.
50. As a user, I would like to have control over the in-game camera so that I can script things like cutscenes.
51. As a user, I would like a system to write and execute my own C++ code so that I can extend Parchment's basic capabilities.
52. As a user, I would like to export my game project as an executable so that I can deploy and release the game.
53. As a user, I would like to view any demo game made using Parchment so that I can collaborate.
54. As a user, I would like to set the desktop icon for my game so that I can add personality.
55. As a user, I would like to use UI hotkeys so that I can more efficiently navigate.
56. As a tester, I would like to record user input segments so that I can troubleshoot glitches.
57. As a tester, I would like a logging system so that I can troubleshoot game crashes.
58. As a user, I would like to enable source control so that I can manage different versions of my game. (time permitting)
59. As a user, I would like the option to implement local co-op so that I can create more types of games. (time permitting)
60. As a user, I would like the option to define controller/gamepad bindings so that the game can be played with a gamepad. (time permitting)

Non-Functional Requirements:

Usability

Parchment aims to be easy to both navigate and use for users of any experience level. Through an intuitive and well-designed GUI, our engine caters to entry-level game developers while also providing additional features and extensibility that may appeal to more experienced developers. Parchment aims to reduce the learning curve found in most modern game engines in an effort to remove entry barriers into game development. To accomplish this, the engine comes pre-loaded with tutorials as well as a demo of a game created using the Parchment engine so users can get an idea of what they can achieve with the tools that Parchment provides. Additionally, the Parchment application itself is lightweight and non-intensive due to its sole focus on 2D games, making the engine accessible to developers with varying degrees of computing power.

Expandability

Parchment's support for user-generated C++ code allows for a degree of expandability that gives more experienced game developers the ability to add features to their game as they see fit. This allows the Parchment engine to not be geared solely towards entry-level developers, but satisfy the needs of experienced developers as well.

Performance

Parchment aims to be as lightweight as possible so that it can easily run on different types of user machines. Games created with Parchment will target the 30-60 fps range, which can be improved with further user optimization. The engine should manage memory effectively so that the application itself, and any game created with it, runs with optimized resource usage.