# Team 12 Project Charter

**Team Members:**
Jacob Brown, Raymond Xie, Xiaoyan “Larry” Li, Jack Locascio, Drew Hatfield, Josh Gerber

**Project Title:**
Parchment Engine

**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. Our native Windows game engine 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 game development, our lightweight game engine eliminates the large time investment it takes for new users to become acquainted with other more complex engines and equips them with the necessary resources to create their own 2D bird’s eye view game.

**Project Objectives:**
1. Create a game engine that allows users to interact with a GUI that lets them create & customize a 2D bird’s eye view game (similar to the original Zelda) with audio and network support and an accessible visual debug mode.
2. Provide an art asset editor for users to create and edit their own maps (item placement, art assets, etc.) and sprites (such as enemies).
3. Provide a script editor to design actions of any game element and let users design their own unique game mechanisms.
4. Provide support for user-generated code (C++ support only).

**Stakeholders:**
Users: Our users would primarily include people looking to create their own game, specifically in the 2D top-down style of the original Zelda. As our game engine is simplified, it would probably appeal to less technically-adept users who find our engine’s ease of use attractive.
Developers: Jacob Brown, Raymond Xie, Xiaoyan “Larry” Li, Jack Locascio, Drew Hatfield, Josh Gerber
Project Manager: Ian Ryan
Project Owners: Jacob Brown, Raymond Xie, Xiaoyan “Larry” Li, Jack Locascio, Drew Hatfield, Josh Gerber

**Project Deliverables:**

- Primary platforms/frameworks/etc: Windows, OpenGL, imgui, C++, SDL2.
- Build a native Windows GUI and OpenGL render context to assist users in the game making process.
- Construct a robust game flow control that is optimised specifically for top-down action games.
- Provide support for user-written C++ code and allow the engine to remain extendable.
- Construct a file system to allow our engine to produce executable game files and manage resources.
- Develop an asset editor that allows users to create and edit their own assets, such as sprites, maps, and items.
- Provide a run-time debug mode which allows users to visualize game mechanisms and debug on the fly.
- Develop a keyboard and gamepad controller module that can be integrated and remapped.
- Write logic config widgets that generate scripts for game elements.
- Write a random generator utility that can use media to quickly generate game elements.
- Build an audio management system that supports complex mixes of sound effects and music.
- Support network connections for multiplayer that hides all protocols and system interfaces from users (cleans up UI for ease of access).
- Build a demo game using the engine as a tutorial for users.