Burning Choices

Project Designers: Ethan Seiber, Jamel Warren, and Hunter Haislip

Abstract

In the course of the next couple of months we will develop and present a 2D game named "Burning Choices". The game will consist of the genre adventure. It will include multiple levels and hopefully branching story depending on the time constraints. The game took inspiration from the Bandersnatch, a fictitious game in the Netflix original series Black Mirror that also included branching story line. The target audience will be those that enjoy classic action arcade style games. Things achieved thus far include the brainstorming. We have put together many ideas of things we would like to achieve with this project.

1. Introduction

The project is a 2D adventure game. The end goal of the project is to tell a story that branches depending on the actions of the player. The audience for this project would be those who enjoy retro style games. The audience will hopefully play the game and leave feeling satisfied/ entertained. The reason why we have chosen this project is thanks to different interests ranging from a love of video games to things we enjoy in our own lives and would like to create ourselves. The story will follow the player as they find their way through our world overcoming any obstacles in their way. In doing so they are given the ability to control their own story.

1.1. Background

Throughout the paper these terms will be used in describing certain aspects or situations in the game. Title screen, 2D, branching storyline, linear storyline are some of the terms. When you watch a movie before you can watch a screen will pop up that says play. This is the title screen and video games have them. They make sure the user is ready to play instead of just throwing them in where they left off. 2D refers to the dimensions the game will be in i.e. 2D is short for 2 dimensions. To easily understand what this means picture a square drawn on a paper and then think of a cube. You can tell they are both completely different because the cube has depth this is a 3D representation that operates on the x,y and z coordinate plane. The square is the 2D representation because it's flat and only operates on the x and y plane. Branching storyline refers to the story structure and how the story of the game is built. The story that is produced depends on the choice of the player. This means the player could end up with a completely different ending or story leading to the ending depending on what they do in the game. Linear storyline also refers to the story structure within the game. A linear storyline is much like reading a book. From beginning to end the story is static and unchanging with only one ending. This definition is applied to games as well. This is a team project so we all have our own motivations for deciding to make a game. For me (Ethan) and Jamel it's the video game aspect of it. Jamel and I plan to take our degrees in computer science to work in the video game industry. For Hunter it's the the idea. Hunter received inspiration from the Black Mirror, a tv series, episode Bandersnatch to make the game have multiple endings based on player choice. Hence the title Burning Choices.

1.2. Challenges

Some of the challenges we suspect to run into include programming the interaction between the player, character and game environment. This is drastically different from any programming projects we have attempted before. We have learned about gathering user input and manipulating it, but we need to manipulate it in a way that interacts with objects or the player. Creating in game objects like enemies and obstacles leads in to the environment interaction challenge. A saved games are a necessity for games which entails gathering the players current place in the world as well as their choices in the story and be able to use this to load their progress when they load up the game. Challenge four is to create our different story lines and make sure they lead to an ending. Finally is the debugging. This is more of a patience challenge. To address these challenges we will need to do some research on useful libraries to help provide pre- defined functionality.

2. Scope

The project will be finished when we have at least three in game levels, a linear storyline, save states, as bug free as possible and the world builds and functions properly. The stretch goals we have set for this project are the branching storylines and more than three in game levels.

Use Case ID	Use Case Name	Primary Actor	Complexity	Priority
1	Add item to cart	Shopper	Med	1
2	Checkout	Shopper	Med	1

TABLE 1. SAMPLE USE CASE TABLE

2.1. Requirements

As part of fleshing out the scope of your requirements, you'll also need to keep in mind both your functional and non-functional requirements. These should be listed, and explained in detail as necessary. Use this area to explain how you gathered these requirements.

2.1.1. Functional.

- The program should load up the game "Burning Choices" and start on the title screen.
- The game will allow for saved states.
- The player can choose from the title menu to do things like load a game or start a new one.
- WANT TO ADD MORE!!!

2.1.2. Non-Functional.

- Usability- The game will give the player an amount of entertainment through gameplay.
- Recoverability- If the game crashes then the user will be able to open the game up and continue from a save. If the save corrupts then a backup of your save will be loaded instead.

2.2. Use Cases

Use Case Number: 1
Use Case Name: Running the game
Description: The user wishes to play the game so they run the executable. This will launch the game to a beginning menu.

1) The user finds their executable
2) The user opens the executable
3) The executable throws the user into the game's beginning menus
4) The user is inside the game viewing the menu options

Alternative: User creates a shortcut to the executable

1) User opens the shortcut they created
2) Then the shortcut will open the game.
3) The user will be put on the title screen

Use Case Number: 2
Use Case Name: Starting the story.
Description: The user is on the title screen and wants to start the story of the game.

1) The user will see multiple menu items on the title screen.
2) The user will click on New Game to start a new story.
3) The game starts at the beginning of the story.

Use Case Number: 3
Use Case Name: Game saving
Description: The user reaches a point in the game where they wish to save their progress and quit.

1) The user will hit the Esc key to pause the game.
2) A list of menu items will pop up.
3) The game environment will pause allowing the user time to read the items.
4) The user will select Save Game in the pause menu.
5) The game will store the player's story data in a file created by the executable.
6) The game will un- pause and begin where the user left off.

Use Case Number: 4
Use Case Name: Load Game
Description: The user is on the title screen and wishes to continue where they left off in the story.

1) The user will see a menu item that reads "Load Game".

2) The user will click ”Load Game”.
3) The game will open the file holding the saved data of where the user last saved.
4) Using the saved data the game will load to the point where the user last saved in the story.

2.3. Interface Mockups

Figure 1. First look at what the gameplay will be like

Figure 2. What can be expected from the environment of the game

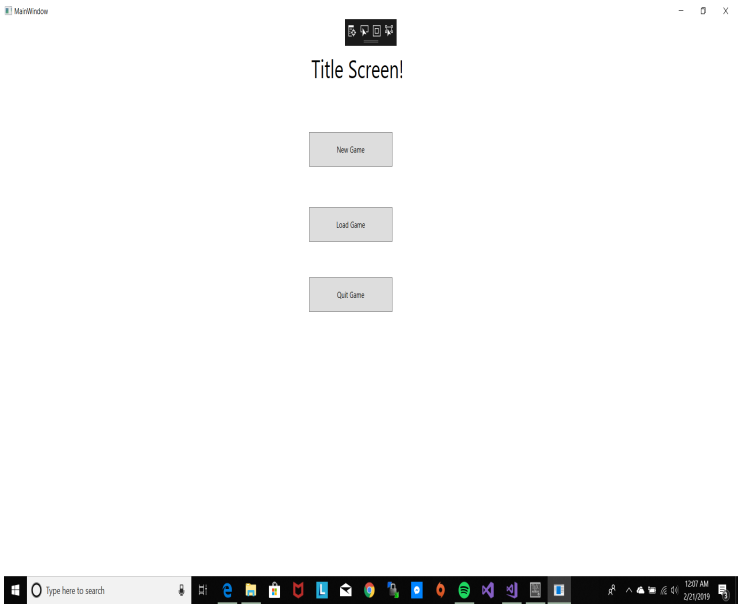

Figure 3. A layout of the Title Screen

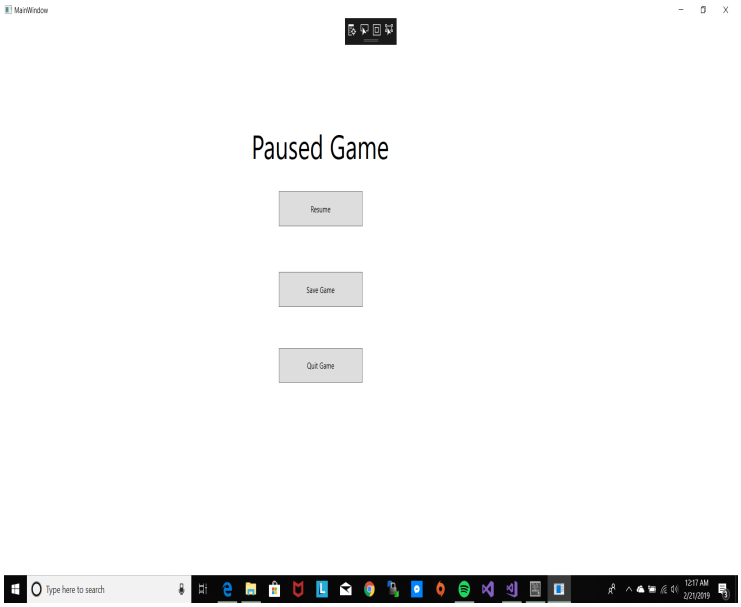

Figure 4. A layout of the pause screen

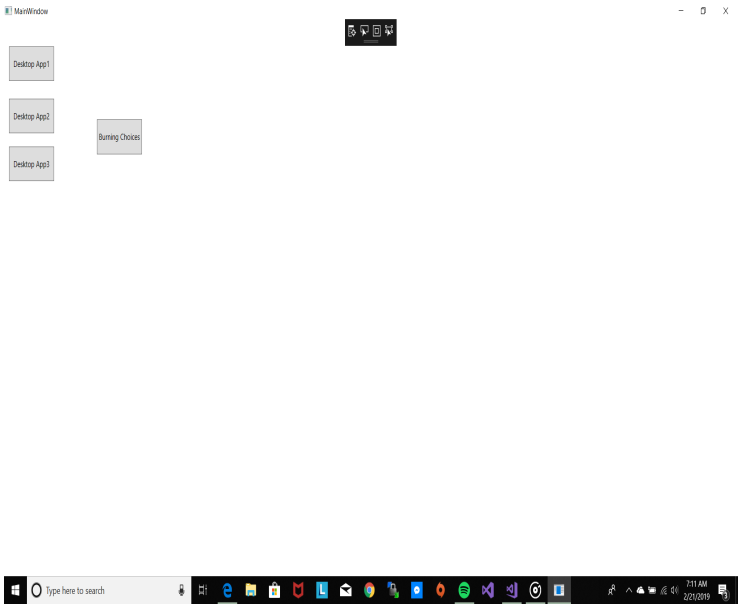

Figure 5. How opening and running the game might look

3. Project Timeline

4. Project Structure

At first, this will be a little empty (it will need to be filled in by the time you turn in your final report). This is your chance to discuss all of your design decisions (consider this the README's big brother).

Figure 6. Your figures should be in the *figure* environment, and have captions. Should also be of diagrams pertaining to your project, not random internet kittens

4.1. UML Outline

4.2. Design Patterns Used

Make sure to actually use at least 2 design patterns from this class. This is not normally part of such documentation, but largely just specific to this class – I want to see you use the patterns!

5. Results

This section will start out a little vague, but it should grow as your project evolves. With each deliverable you hand in, give me a final summary of where your project stands. By the end, this should be a reflective section discussing how many of your original goals you managed to attain/how many desired use cases you implemented/how many extra features you added.

5.1. Future Work

Where are you going next with your project? For early deliverables, what are your next steps? (HINT: you will typically want to look back at your timeline and evaluate: did you meet your expected goals? Are you ahead of schedule? Did you decide to shift gears and implement a new feature?) By the end, what do you plan on doing with this project? Will you try to sell it? Set it on fire? Link to it on your resume and forget it exists?

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.