The Arizona Trail [2018]
CSC 335 – Final Project

Image Source:
http://www.robinsonlibrary.com/america/uslocal/west/northwest/graphics/oregontrail2.gif

Table of Contents

Overview

Your job is to implement a game that is vastly similar to the classic game, “The Oregon Trail,” however in your implementation, the user will be travelling along the Arizona Trail from the Mexican border to Utah. If you have played this game before, then you will know that your character will be travelling via wagon in a party of three other people. During this journey, the user has the opportunity to buy supplies, and then they must simply just make it from point A to point B through a series of multi choice options that drive the gameplay.

This description might seem easy enough, however there are a few things that might occur along the way, which should allow the user to interact with the environment in the following ways:

1) Hunting for food along the trail
2) Purchase goods from pop up shops along the way
3) Managing inventory of supplies to prevent them from running out
4) Rate of travel depends on current weather (raining, snowing, etc…)
5) Encounter some calamity along the way
    a. Wheel breaks
    b. Bison dies
    c. Party member dies along the way

The user will continue on their odyssey through Arizona until they either get to Utah, or they die along the way. At this point, the game should announce to the user, and should display a score that is decided on by your team. With this in mind, the top four scores should appear on the main loadout screen.

Along with the core travelling during the meat of the gameplay, the user should be able to prepare for their journey in an initial town from the general, where they can spend a predetermined budget on supplies such as bullets and food. The user should also be able to get hints about the journey the are about to embark on from the town in whatever creative way you come up with.

Your Task

You are expected to create a fully functional version of the Oregon Trail, that is modified to have somewhat of an Arizona theme to it. The user must be able to go through an engaging journey between the two locations, with multiple stops along the way, alongside the ability to hunt along the way.

With all of this being said, feel free to draw from childhood recollections of the game, or through watching walkthroughs on the web of the number of iterations of the game that have been released. Creativity through the stopping points along the way are heavily encouraged, and at the end of the development cycle, there should be some sort of wow factor (worth 10% of final grading) implemented into your game. However, it is crucial that the core gameplay of moving from town to town is implemented.

Base Requirements

General

- Game should have scoring system to allow for the player to climb a leader board displayed on the loading screen.
- Player begins in a town and is allowed to purchase items for the journey. The user should be able to designate when they want to begin, upon which the overall view will change and transition into a different view.
- If the user dies on the trail, or makes it to Utah, the user should be presented with their score, as well as the option to include a name associated with the score for the leaderboard.
- The user should have the ability to close the application whenever they want.
- The user should have the option to stop their game and return to the main menu.
- The user should be prompted to save the game upon attempting to close the application
- The game should feel fluid, and none of the design features should feel out of place (trust us, you will know when this is the case).
- The primary view of the game should be animated, such as wagon wheels moving, and bison feet moving with landmarks in the background

Map

- Map should consist of key locations to stop off on throughout the trail journey.
- This map of stops should be displayed just before the user begins their travel from the initial location, displaying:
  - Stops to buy goods from
  - Landmarks
  - The overall route from start to finish
  - Any significant land features such as mountains or rivers
- There should be different views for the various locations along the map
- The main view of the player moving along the trail is dynamic, with land features moving in the background.
- Number of stops and landmarks is up to user implementation, but should feel thorough
- Hunting: The user should have the ability to stop moving, and enter a new view that will lead them into a hunting mode, as a means of collecting food resources to keep their character and party alive. This view should be reactive to the user via mouse or keyboard, and feel like a mini game within the actual gameplay.
- The user must be able to select whether to stop off, and what to do along their journey through each road between stops, land marks, and the primary stopping and ending points.

Hunting

- The user must be able to stop off and enter a hunting view to shoot down animals for resources.
- How these animals appear on screen, and what kind of animals they are is up to team implementation.
- Various animals should include different resources/different number of resources.

- It is up to team implementation as to whether this is a third person, or first person view, as well as how the actual hunting mechanic works. However, the animals should be able to run away, and should not just statically sit on screen
- The user should be given a stat sheet of what sort of resources they collected from hunting screen
- The resources used during hunting should be limited, and be affected by each hunting sequence (ex: user used 10 bullets to shoot a buffalo).

Party Dynamics

- The party you travel in should be affected by different calamities that could ultimately result in them dying in the middle of the game
  - Starvation
  - Dehydration
  - Pneumonia
  - Dysentery
  - And much more!
- Party members you have left remaining should impact the overall score the user can achieve

Resources

- The game must implement key resources that affect the overall gameplay
  - Bullets
  - Food
  - Water
  - Blankets
- Feel free to include more resources than just these
- Losing or depleting any of these resources should be the driving factor that results in one of the dynamics outlined above

Traveler

- The trainer should begin without any of the described dynamic items in their inventory.
- The trainer should also begin with a designated budget that they can spend in the initial town on the resources they will use throughout their gameplay
- Once the game begins, the user will keep the leftover budget they have to be used on other resource stops along the way.
- During normal gameplay, the user should have different options they can select that will drive the game forward.
- The user’s character should be prone to the different calamities that can happen to their party, that will ultimately result in their gameplay ending early.

Animation

- The main view when the user and their party is travelling should be animated. This will include things such as wagon wheels to turn, bison legs to move, and land features or clouds to move in the background as well
- The hunting area should also be animated, and will allow for animals to move across screen.

Scoring System

- Scoring should be based off a multitude of factors from the gameplay
- This scoring mechanic should be decided on by the team, and implemented as they see best fit.

Persistence

- The user should be able to save the game, and select an option to the continue the game after starting it up again

Sound

- Game should have background music, and sound effects for any major events that occur in gameplay.

User-Guide

- The game is required to have an unorthodox user guide that works in the form of giving the user hints as to the best strategy to win the game
- This can be done via pop up window from the main menu, or through interactions that occur within the town at the start of the game.
- These hints should be clear enough that a new user to the game would be able to pick up on them well enough to at least make it from the starting point ot the ending point of the game.

Wow Factor

Wow factors are a touch you would like to put into your project that exceed the base requirements, and provide an example of some aspect of mastery from one of the concepts within the course. Ideas for these can be brainstormed during initial scrum meetings with your scrum master, to ensure that all ideas are on the right track with course material.

Scrum Meeting Schedule

This project will utilize weekly scrum meetings to simulate a real software development environment, where certain requirements must be met by the designated meeting.

1) Week 1: Initial planning and assignment of tasks. This is where you can communicate loosely who will take ownership over which aspects of the project.
2) Week 2: Initial layout. This scrum meeting you should have already started your project, and you should have a UML diagram of your project, alongside a way of modelling the expected decision behavior, and random event generator. By this point, you should also

have a very strong idea of what sort of “wow” factor you want to implement into your project.

3) Week 3: “Alpha” Version. By this point, your project should be complete, with some aspects being okay to be left out. Specifically, everything should be functional, GUI and all, without the expectation of sound, animation, leaderboards, or consistency being implemented yet.
4) Week 4: Final Version. The project should be completely done, meeting all requirements including sound, animation, leaderboards, and consistency. Comments should be complete, menus should be functional, and a demo should be given during this meeting, displaying all expectations being met to your scrum master.