Matthew Lehmann, Sam Ehrlich, Jordane Coombs, Tri Bui, Paco Ramirez
Team 105-1
02/13/2020

Major Features:

- Game interface: This will include a third person 3-D view of the player model as the player's ball moves along the obstacle course. This GUI will include menus, options, settings, timer, ball object, and obstacle course.
- Online database for the highest scores: This database will host a global leader board for people who play the game.
- Unity game development engine: Ray tracing for high quality rendering of the obstacle course and player object with intuitive game development tools
- Video Recording: Give the player the option to share their run on the obstacle course on the game website
- Social Website: This will give people who play the game the ability to share their best runs
- Account Registration within the website: This will provide players with the opportunity to keep an online account for communicating with other players
- Downloadable on steam or other marketplace: The game will be contained within an application that can be deployed on an application store

Major Requirements:

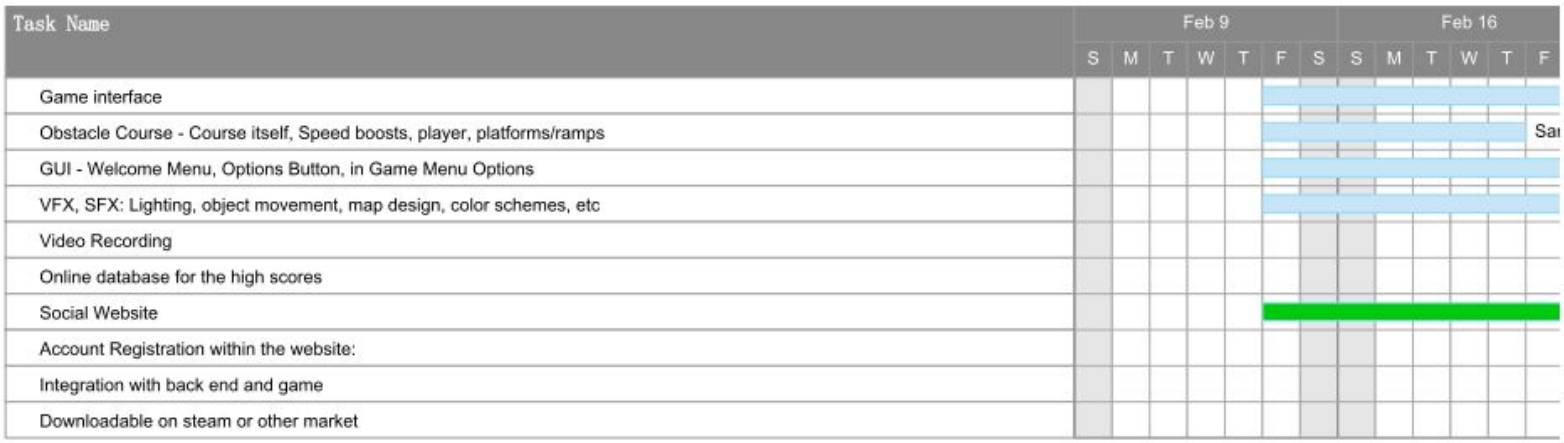
- Feature 1 - Game Interface:
    - Functional: We will need a game development engine that provides us with the means to generate a 3-D game. In this case, we will use Unity's 3-D engine. There are a plethora of resources available online that teach developers how to efficiently use the product as well as how to officially release it on open market applications. The engine also provides a means of hosting servers and sending information to your own servers if you want your game to have online capabilities

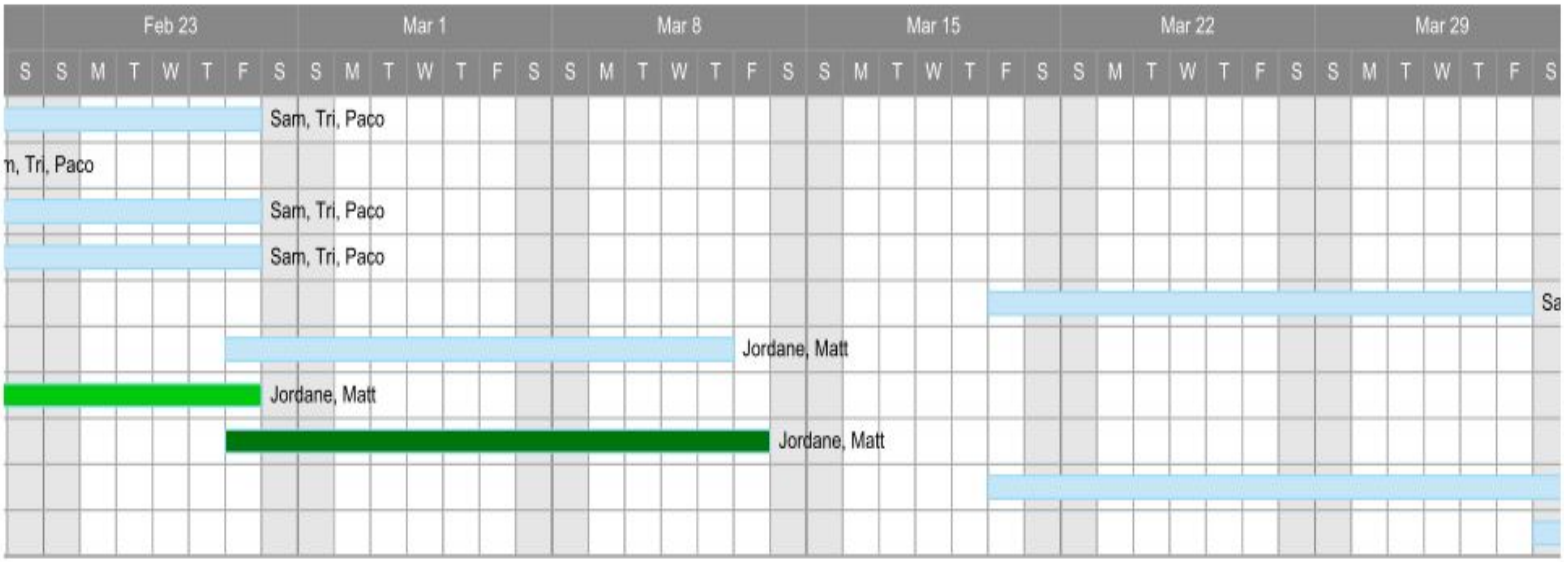
    - Functional: Player object, player controls, menu, timer, obstacle course, camera movement with the player. Everything that the user will experience while they play the game will need to be visible and moveable according to the player's whim.
    - Non-functional: Certain lighting of the map, color schemes of player objects and obstacle course, multiple ways to complete the course (including diverse enough terrain). These are all things that the player won't have any control over.
- Feature 2 - Online database for high scores:
    - Functional: SQL-based storage table that can append new player times. There will be a limit on the times people are allowed to post. The times must be fast enough-based on some not currently decided time-to actually make the table. This will prevent the table from being appended to every time someone plays the game. A maximum size of the table will also be determined to ensure that the table isn't enormous after a certain amount of time
    - Functional: Server to actually store the information in a way that allows everyone who plays the game to see other people's scores. This will also provide a way to save the information appended to the table online so that it is continuously accessible in the future.
    - Functional: We may need a way to work around not having a dedicated server, or finding a way to have a dedicated server so that we can save the global leader board for as long as the game runs and as long as users continue contributing to it
    - Non-Functional: Display username and time the user took to complete the course. This will be something that anyone on the website can view once they have logged into their account, people can comment on people's times, ask them how they accomplished it on the track, etc
- Feature 3 - Social Website
    - Functional: We will need a domain to host the website on. It could be a free domain that is hosted by a third-party website or one that we pay for if we can find a cheap one. This will need to provide users

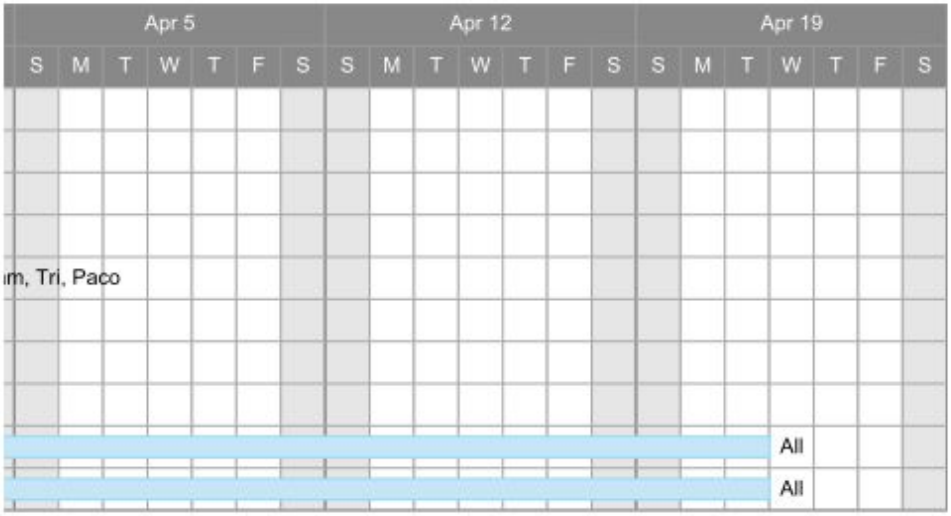
with a single location they can go to, to access their Guayaki accounts and the main game leaderboard
    - Functional: The website needs to be able to communicate with the individual instances of the game that is downloaded on each user's computers. This way their scores can be sent to the online database and stored for others to see. This will also provide the means to upload video recordings if the user chooses to
    - Functional: We will need the appropriate HTML, CSS, and JavaScript to account for all of the website functions that will be required for the user experience. The user will need to be able to switch between pages, see an updated leaderboard when new scores are added, etc.
    - Non-Functional: The CSS and style of the website should be appealing to the users. The interface should be uncluttered, functional, and provide each user with everything they need in the most efficient way. This will include color schemes on the website, button styles, how the database is displayed, where to login/register, etc.
- Feature 4 - Downloadable on Steam or other platform:
    - Functional: This will require a single executable program that contains the entire game within itself. The user should be able to simply go to the marketplace at which the game is available, download the game and play it. This will need to account for the operating system the users tend to use. We will need to make sure the game is capable of running on whichever OS's we want it to be available on. Unity provides a lot of open-source information on how to accomplish this through online tutorials and written documents
    - Non-Functional: We will want a way to display who worked and contributed to the project. The icon of the game will need to be decided upon, any recognition for the Unity game engine will need to be accounted for accordingly as well
- Feature 5 - Account registration within website
    - Functional: We will need to utilize the appropriate HTML and CSS code to develop the interface for registering. This will be similar to

the registration page that can be created in the fourth lab of this course.
    - There will need to be a database to store user accounts and remember users so that they can log in and out of their accounts in between using their browsers.
    - Non-Functional: The user needs to be able to customize their account in some way to distinguish themselves from other users. This can include choosing avatars, writing bios, etc.
- Feature 6 - Video Recording
    - Functional: An actual means to record the player while they are completing the course. This will need to be included within the Unity game itself so that the user won't need to open any third-party software. The user should then be asked whether or not they want to save the video that was recorded for them, if so they can choose to send it to the website. Storing the video will not be something within the scope of this project, so the website will be designed to access the player video from their local device, or the player can simply hold on to the video themselves and not choose to share it. That way the video does not need to be stored on a separate server.
    - Non-Functional: How the video is stored so that users can play it on whatever video players they have on their own computer
    - Non-Functional: Whether or not players want to store the video on their local device or if they want to try and post it to the website. The website posting won't be a requirement for developing this completed project.

GANTT CHART:


Task Name
Feb 9
Feb 16
S M T W T F S S M T W T F
1 Game interface
2 Obstacle Course - Course itself, Speed boosts, player, platforms/ramps
3 GUI - Welcome Menu, Options Button, in Game Menu Options
4 VFX, SFX: Lighting, object movement, map design, color schemes, etc
5 Video Recording
6 Online database for the high scores
7 Social Website
8 Account Registration within the website:
9 Integration with back end and game
10 Downloadable on steam or other market


Feb 23
Mar 1
Mar 8
Mar 15
Mar 22
Mar 29
Sam, Tri, Paco
Sam, Tri, Paco
Sam, Tri, Paco
Jordane, Matt
Jordane, Matt
Jordane, Matt


Apr 5
Apr 12
Apr 19
All
All