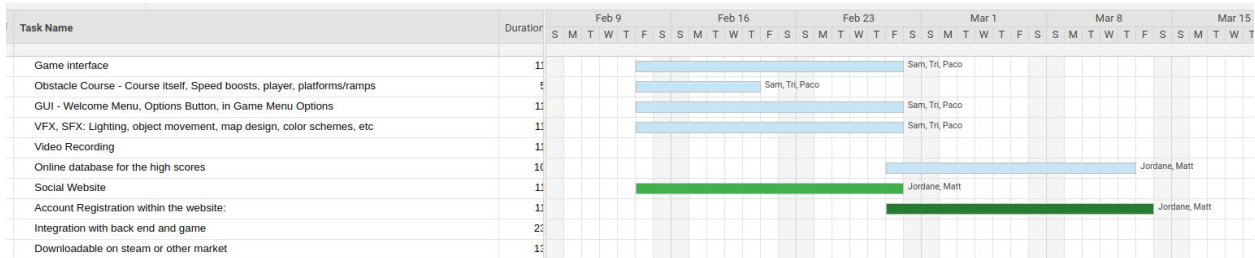


TITLE OF PROJECT: Guayaki the Game

WHO: Matthew Lehmann, Tri Bui, Sam Ehrlich, Luis Ramirez, Jordane Coombs

PROJECT DESCRIPTION: Guayaki the game is a single player obstacle course-based game with an active website hosted on Heroku. The game records how long it takes for a user to complete the obstacle course and compares this time to the quickest time they've previously completed the course. This is then updated on the live website's global leaderboard which keeps track of all of the users who have created an account with the Guayaki game. Users can then create their own profile, customize their profile picture, and compete with other players to see who can finish the track the fastest. The game itself is a Unity developed and deployed application that can be run completely on its own. The website is hosted on Heroku and allows users to create an account, write a short bio that will be displayed on their profile page, and choose a profile picture from online that they want to be associated with their user account. Sending the user's score from the game to the website was the last feature that needed to be implemented, and unfortunately the group did not have the time/skills to implement it. Therefore, the users best time column is blank on the website.

PROJECT TRACKER:



<https://app.smartsheet.com/sheets/28M6RQPCPGCYFQV38MWMQJH25CGFWJC9JWWVFX6I?VIEW=GANTT>

VCS: [HTTPS://GITHUB.COM/MATTHEW-LEHMANN/GUAYAKI_CODING_REPO](https://github.com/matthew-lehmann/guayaki_coding_repo)

CONTRIBUTIONS: This needs to be clarified. Most of the contributions from the team came within Unity specifically. This means that we had 3 people primarily working in Unity's own version control system and the other 2 working in Github. So our Git commits are not actually reflective of the overall contributions from the team and should be taken with a grain of salt.



Luis Ramirez: Focused mainly on game development including the importing of game assets, environment updates, physics engine updates and timing features. Was the main driver of the later and final stages of the game design. Fully deployed the game as its own application for Windows and Mac operating systems.

Sam Ehrlich: Focused mainly on game development, including making a simplified version of the full Guayaki game. Made a ball object and full obstacle course that had multiple levels, adjusted game engine physics, a timer mechanism, and overall, multiple ways to be completed. That is to say, the user could take different paths through the course to finish it, which would provide ways for users to record faster and faster times.

Tri Bui: Focused mainly on game development, specifically the early stages of game development. This included making custom structures and platforms, adjusting early game engine physics and setting up user control of game components.

----- Tri, Sam, and Luis maintained their own VCS in Unity as collaborators-----

Jordane Coombs: Focused mainly on website and postgresSQL development. Setup the columns and table in postgresSQL to handle users choosing their own profile pictures, to handle writing their own short bios and managing their usernames and passwords. Then aided in understanding the node modules package that comes with the express node.js files we are accustomed to using in previous labs in the course. Jordane sent Matthew all of his code so we could integrate it before pushing it (to avoid having a bunch of branches).

Matthew Lehmann: Focused mainly on website development and deployment to Heroku. Developed all of the EJS pages for the website and created the server.js file to handle post and get requests from each of the website pages (so handled the integration layer between the website and the database). Handled user account creation process modal as well as the login verification process with the postgres database. Deployed the database and website to Heroku in the last step of the website creation.

DEPLOYMENT: [HTTPS://GUAYAKI.HEROKUAPP.COM/](https://guayaki.herokuapp.com/)

- If you are making an account, make sure that your profile picture comes from online.
- So you can google for a picture. When you find one you like, make sure to right click it and press “open image in new tab”. Take the link of this new tab and insert that into the profile picture box. MAKE SURE that the link is not extremely long, if it is beyond a certain amount of characters, the website will return an error. Choose a short link.
- Like this one: <https://i.imgur.com/idLV3vLg.jpg>

GAME DEPLOYMENT: Download the application for your operating system and run it.