**Welcome to Day #5 of CGCC!**

Every day we will have a GitHub repository page that outlines each day and the activities that we will complete. We will also provide all homework on these pages.

Feel free to browse the other days to see what is coming up!

As always, let us know if you need any help or have any questions.

**Day 5: Prototyping!**

**SCHEDULE:**

* Instructors start the video call.
* Go over homework from day 4 (~10 min.)
* Full camp activity (~30 min.)
* Go over how to use GitHub wiki and what we can use it for in context of CGCC. (~5 min.)
* Go over what is Prototyping? Why do we do it? How do we do it? (~5 min.)
* Watch a video on prototyping if time allows (7.5 min.)
* Practice prototyping with drawing (using a common application like Spotify/YouTube) (~20 min.)
* GoDot prototype their game (90 min. + 15 min. break)
  + Brainstorming (30 min.)
  + BREAK (15 min)
  + Finding assets (20 min.)
  + GoDot development (40 min.)
* “game jam” homework?

**INSTRUCTION:** Go over homework from day 4 within their groups. (~10 minutes)

* Which game did you pick? Why did you pick that game?
* What specific user interface components did you find in your game?
  + Why do you believe these components were added?
* Is there anything that you think is missing from the user interface (in other words, is there anything you would add?)
  + Why would these features be beneficial to the player?

**ACTIVITY:** Full camp activity (~30 minutes)

* <Fill in with camp activity>

**INSTRUCTION:** Using GitHub wiki for personal journals/logs. (~5 minutes)

* Students should be guided back to GitHub to their game’s repository (Assuming they already have it created, even if it’s empty)
* Navigate to the wiki page and have students create the first page for their wiki
  + The first page should be an introduction to the game and its creator(s).
* Introduce using the wiki to create a daily log for their project.
  + Each day, students can create a new page marked for each day.
  + For each page, students will keep track on the progress that was made during the camp for that particular day.
  + Even if nothing “of use” was accomplished in developing their game, students can still log what they did during the camp and if they learned anything that they believe might be useful later during the camp.

**INSTRUCTION:** What is prototyping? (~10 minutes)

* Game prototyping is an important component in the game development process. This involves creating a method to test the concept of the game, to see if the idea of the video game can be put into practice before investing too much money and time in the project. (<https://starloopstudios.com/rapid-game-prototyping-why-is-it-important-in-game-development/#:~:text=Game%20prototyping%20is%20an%20important,and%20time%20in%20the%20project>.)
* Discuss why it is important we try to prototype games before development starts. (<https://www.uxpin.com/studio/blog/paper-prototyping-the-practical-beginners-guide/>)
  + Rapid iteration
  + Inexpensive
  + Increased creativity
  + Less of an up-front learning curve
  + Etc.
* Different ways to prototype:
  + Paper
  + Digital
  + Native
  + Any others or ‘subdivisions’ of the above methods (like sticky notes)?

**INSTRUCTION:** Summarize with a prototype video (7.5 minutes)

* <https://www.youtube.com/watch?v=JMjozqJS44M>

**ACTIVITY:** Create a paper prototype of a common application such as Spotify or YouTube (~20 min) (pairs)

* Draw this application with marker/paper/scissors/ whatever you feel is necessary.
* List all the functions of this application and how each element of your design is used to do this function (A play button plays the music)
* Take a picture of your design and upload it to (#WHERE\_TO\_UPLOAD)
* Share with your partner why you chose to include certain elements. Where there any that you decided to leave out? Why?
* Discuss what having too many features can do to an application. Where do we draw the line on what is a good, useful feature and what is meaningless clutter?

**ACTIVITY:** Brainstorm their game (~30 min)

* Characters: (10 min)
  + The Fighter: A hero who fights the enemy with their fists, feet, or weapon.
    - Examples: Street Fighter, Karateka, Mortal Kombat
  + The Big Bad: A dastardly villian who appears to be stronger than the hero, but has a hidden weakness.
    - Examples: Gannon, Donkey Kong
  + The Sage: A mystic, an odd character allied with the hero who provides helpful hints at crucial times during the game.
    - Examples: Legon of Zelda, Skyrim, Metal Gear Solid
  + The Sidekick: A companion to the hero who provides comic relief or aids the hero in solving their quest.
    - Examples: Luigi in Mario Bros, Yoshi in Mario World, Sonic 2's Tails.
* Narrative: (10 min)
  + Overcoming the monster: The hero must flight and slay the monster that threatens their community.
    - Examples: Beowulf, Dracula, King Kong, Pacman, Mario Bros., Space Invaders, Asteroids, Galaga
  + Rags to Riches: An insignificant person is dismissed by others. Something happens to elevate them, revelaing that person to be exceptional.
    - Examples: Ugly Duckling, Aladdin, Superman
  + The Quest: The hero must set out on a long hazardous journey to battle obstacles until they are triumphant.
    - Examples: Lord of the Rings, Harry Potter, Wizard of Oz
  + Voyage and Return: The hero travels out of their normal world into the unknown and overwhelming, before escaping back to the safety of their home.
    - Examples: Alice in Wonderland, Finding Nemo, Gulliver's Travels, Legend of Zelda, Super Mario Bros.
  + Rebirth: The hero falls under a dark spell (e.g. sleep, sickness, enchantment) before breaking free and being redeemed.
    - Examples: Sleeping Beauty, Beauty and the Beast
  + The Neverending Story: A repetitive story with infinite challenges that get more and more difficult to beat
    - Examples: Donkey Kong, Q\*bert, Tetris
* Rules: Defines how the characters can move through the game world and describes the actions they can take and their effects. (10 min.)
  + Navigation
    - Walking, Running, Swimming, Flying
    - Constraints to only walk up/down, left/right
  + Information
    - Reading a scroll
    - Listening to a character
  + Inventory
    - Picking up an item
    - Choosing to use an item
    - Dropping an item
    - Losing an item
  + Obstacles
    - Jumping
    - Running through
    - Punching at
  + Fighting
    - Punching at
    - Jumping on top of
    - Kicking
    - Running through
    - Round-off back handspring
  + Dying
    - Getting run over
    - Getting hit
    - Jumping into a pit
    - Running into yourself
  + Winning
    - Eating all the food
    - Defeating all the enemies
    - Solving all the puzzles
* Technology:
  + Mapping the game buttons to player actions
  + Secret game modes
    - Example: Pressing A and B buttons together makes the character invisible to monsters.

15 Minute Break

**ACTIVITY:** Find some basic assets (~20 min)

* Now that we have a basic idea of what the game is going to be, we’ll start looking for assets to prototype the idea
* Linked are several different free and Creative Commons / public domain asset libraries that will make it easy to just search, download, and add different assets.
  + <https://opengameart.org/>
  + <https://itch.io/game-assets/free> (itch does have premium assets that need to be paid for, but still have thousands of good free ones)
  + <https://www.gamedevmarket.net/category/2d/?type=free> (basically the same as itch.io)
* Download several different kinds of each type of assets so you have options. Consider an asset folder with different sub-folder marked for what they’re used for (character, trees, weapons, etc.)

**ACTIVITY:** GoDot prototype development (~40 min)

* Using the assets we’ve now collected, let’s get into GoDot and start developing a prototype for some of our game.
* Say my game is a 2D top-down RPG where the player goes on a quest to slay a dragon that’s been threatening the local village for years. I now want to prototype some of the aspects of the game like combat.
* **Prototype example plan for 2D RPG attack prototype**
  + Start by creating a basic scene and playing the play sprite inside.
  + Add keyboard controls to the player sprite.
  + Create a collide-able object and have it work correctly with the player (the player should not be able to move through a rock for instance
  + Create a weapon sprite on character.
  + Create attacking hitbox / button control (animations can be a different beast that we’ll save for a different day, for now, just have it play a sound and change from one sprite to another like a “2 frame animation”)
  + Create an enemy sprite and add collision.
  + Create a means to defeat an enemy (if the player hits the attack and the collision of the sword hits the collision of the enemy, remove the enemy object)
* Use steps similar to these to create a basic prototype for your game.
* Be sure to ask questions when needed.

**INSTRUCTION**: Ask if there are any final questions for the day before bringing the day to a close