PROJECT 1: Launching March 29, 2021 (**Demo Day** Friday, April 23, 2021 9:00am -12:00pm)

Purpose: To create a text based "**CHOOSE YOUR OWN ADVENTURE**" game.

Guiding Question: How might I create a *text-based interactive* game using HTTP APIs with Javascript and ExpressJS?

BIG IDEAS: "I understand and can demonstrate that..."				
Computer programming involves the writing of logic. Included are loops, variables, decisions, functions and results. Breaking down problems into their smallest component is how you accomplish big things. "Eat an elephant one bite at a time."	Design and development is an *iterative reflective* process. Modern web applications have a server-side API that manages the business logic of the app.	Learning new or unfamiliar technologies is a natural part of the process of development. Curiosity about learning and exploring new things is an essential part of being an effective developer.	Applying *standards* to complex ideas is the basis for making sound decisions and is necessary to have the most accurate view of what is actually happening. Writing out an *architectural vision* for your project will enable you to focus when it comes time to writing the *code*.	Ask for help when you need it! Dr. Google Is Your Friend - as is Discord and Basecamp!

Which PROJECT will YOU choose to tackle?

Remember, these are **guidelines. As you navigate the project, where do you see opportunities to **extend** your learning?*

MILD	**MEDIUM**	**SPICY**
Problem being addressed: ● Pass info to an API, execute business logic, return response **Your project should include**: ● NodeJS, ExpressJS ● API endpoints ● README on how to start the server & run the game **I will be able to**: ● Interact with the APIs from the command line ● Understand how information is passed between the "client" (command line) and the "server" (Node/Express APIs) **Example**: ● wget http://localhost/intro ● wget http://localhost/step1?player=fred ● wget http://localhost/step2?player=fred&choice=north	**Problem being addressed**: ● Learning different API verbs (POST, PUT) ● Notion of state **Your project should include**: ● More endpoints ● Data that changes as your user runs through their adventure **I will be able to**: ● Describe when to use other verbs and when to use them ● Change data in the server (although the data won't be persisted unless your spicy, so restarting the server will lose the data) **Example**: ● Your user can pick up some object but leave others ● A combat system with random damage to a user's hitpoints	**Problem being addressed**: ● HTML interface to the APIs ● Persistent storage (mongodb?) **Your project should include**: ● NodeJS serving HTML pages **I will be able to**: ● Offer a "GUI" interface to your users **Example**: ● A web page to host your adventure ● Your user can save their progress and start up again next time ● A point leader board