**DevMtn Capstone Project – Planning Overview**

**Project:**

“Noble Knight” – Text-based adventure game

**Description:**

Browser-based game with a medieval knight theme where the player navigates through several choices trying to solve the puzzle and win the game with as many points as possible.

**Features:**

* Players can create accounts with username + password to save previous games
  + Users are also given the option to play as guest, but their final score will not be saved to the database
  + When users log in, they can view their ranking and past plays (date + score)
* As players navigate through their choices in the game, points are gained and lost
  + Users’ scores are saved on state throughout the gameplay, then saved in the database with AXIOS.POST at the end of their game
* The game will be built with React. Each text prompt will be passed as a prop to a reusable component which references the text content from an object at a specific index in an array. The display will change through conditional rendering.
  + Text prompts will be animated with a typewriter effect
  + Any pixel art used will animate by loading into view line by line
* At “game over,” players are given their score and prompted to play again or quit
  + Points are retrieved from session storage, but then stored to user’s account with AXIOS.POST (as stated above)
* At winning stage, players are given their score and are shown a leaderboard with players ranking highest in the game
  + Points are retrieved from session storage, but then stored to user’s account with AXIOS.POST (as stated above)
  + Leaderboard is displayed with data retrieved (AXIOS.GET) from all users in database and sorted by highest scores

**Styling & Gameplay Overview:**

**Figma:**

[**https://www.figma.com/file/og1AMoS7ll0KWk0yIyD8OZ/Noble-Knight?node-id=0%3A1**](https://www.figma.com/file/og1AMoS7ll0KWk0yIyD8OZ/Noble-Knight?node-id=0%3A1)

**Graphical user interface, application

Description automatically generated**

**Miro (flowchart for storyline/gameplay):**

[**https://miro.com/app/board/uXjVOZF3itY=/?invite\_link\_id=845116465792**](https://miro.com/app/board/uXjVOZF3itY=/?invite_link_id=845116465792)

**Diagram

Description automatically generated**

**Coding To-Do List:**

1. **~~Create react app in VS Code~~**
2. **~~Create basic server~~**
3. **~~Set up initial CSS styling (will add to this for each page view as needed)~~**
4. **~~Write out array of objects for each step/choice during the gameplay (components will reference the text values at specific indices to know what needs to be displayed at what time)~~**
5. **Write out following components & functions & add references into App.js for each as I go –** 
   1. **~~Header (‘Noble Knight” game title that stays at the top of the page~~**
   2. **Input handler for getting player’s desired character name, changes onChange**
   3. **Embed automatic music player for the music throughout the game**
   4. **Text prompt (this will display text for whatever current position the player is at in the game – it takes in the user input for what choice they will make next, then is exported into App.js to display the correct data. The text context is referenced from the array of objects from step #4)**
      1. **Text prompts will include the animation of the typewriter effect from .css file**
   5. **Create user login / account creation function**
   6. **~~Set up sequelize database to hold user info including dates and scores~~**
   7. **Set up leaderboard feature which retrieves data with AXIOS.GET from the SQL db and sorts by highest score**
   8. **Create button components for the “enter,” “save and quit,” “save and play again,” “create user,” and “start new game” buttons** 
      1. **Save buttons will save score and use AXIOS.POST to put scores in database assigned to the specific user/player**
   9. **Test functionality of all buttons and features.**
   10. **Deploy game with Digital Ocean under domain name** [**www.playnobleknight.com**](http://www.playnobleknight.com) **(already purchased)**

**Functions to write:**

1. **User registration (create account, post this username/password to database)**
2. **Whatever allows users to login**
3. **Functions which GETS past playthroughs for users to see their history when logged in**
4. **Function which saves (on state) knight name for particular playthrough (this is different than username)**
5. **Function to randomize events array/allEvents**
6. **Functions in events component to display event text, check choice answer, display choice text, change score accordingly (save in useState/setState), and route to next appropriate event (including gameover and youwin)**
7. **At end of game, function which POSTS user/date/knightname/score to database**
8. **Function that GETs ranked players to display on leaderboard**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Function name?** | **What does this function do?** | **When/where does it run** | **What data/info does it need to run?** | **Where will that data/info come from?** | **What will the function output?** | **What endpoint will it be connected to?** |
| **getAllEvents** | **Inserts info into allEvents array – (1st) (random) (last) ((randomizes the middle events then compile whole allEvents array)** | **onSubmit - (getname)** | **We need the events from events.js** | **Events.js** | **Output an array with (1st) (random) (last2) events as allEvents** | **/api/events** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |