Assignment 3: Use enhanced GPT prompts and CrewAI to create an agentic framework for systematic game generation

Objective: Leverage the enhanced GPT prompts and CrewAI to develop a new, more ambitious HTML5 game utilizing multiple LLM-powered agents. This assignment aims to advance your game development skills by implementing an agentic framework that facilitates systematic and collaborative code generation, enhancing both the functionality and complexity of your game projects.

Instructions

- **Design a More Ambitious Game**
  - **Conceptualization:**
    - Develop a game concept that is significantly more complex and ambitious than your previous projects.
  - **Game Design Document:**
    - Create a brief game design document outlining the game's objectives, mechanics, story (if applicable), and key features.
- **Utilize CrewAI to Create a Crew of LLM-Powered Agents**
  - **Setup CrewAI**
    - Clone the CrewAI repository or install CrewAI following the documentation from the [GitHub page](#).
    - Set up a virtual environment and install the necessary dependencies.
    - You can integrate any Large Language Model (LLM) of your choice with your CrewAI agents to enhance their capabilities. You may use models like OpenAI's GPT, Anthropic's Claude, or any open-source LLM available via APIs or local deployment.
  - **Create Your CrewAI Project**
    - Create a new CrewAI project for your game project
  - **Initialize a GitHub Repository**
    - Create a new private or public GitHub repository for your project.
    - Initialize Git in your project directory
    - Add a README.md file describing your project.
    - Add a .gitignore file to exclude unnecessary files
  - **Commit and Push Your Project to GitHub**
    - Stage and commit your initial project files.
    - As you modify your project, maintain clear commit messages and a logical commit history for transparency and ease of review.
  - **Understanding CrewAI:**
    - Familiarize yourself with **CrewAI** and its capabilities in managing multiple LLM-powered agents for collaborative tasks.
  - **Agent Roles:**
    - Define distinct roles for each agent to handle different aspects of game development. Possible roles include but not limited to:

- **Game Logic Generation:** Responsible for creating the core game mechanics and rules.
- **UI/UX Design:** Focuses on designing the user interface and ensuring an intuitive user experience.
- **User Input Handling:** Manages how the game responds to user interactions and controls.
- **Code Integration:** Integrates code snippets generated by different agents into a cohesive codebase.
- **Testing and Debugging:** Ensures the game runs smoothly by identifying and fixing bugs or performance issues.
  - **Create and Coordinate Multiple Agents**
    - **Agent Creation:**
      - Set up multiple agents within CrewAI, each assigned to specific development tasks as defined in the agent roles.
    - **Coordination:**
      - Ensure that agents communicate effectively, coordinating their tasks to avoid conflicts and ensure seamless integration of their contributions.
      - Implement protocols or workflows that facilitate efficient collaboration among agents.
  - **Incorporate Enhanced GPT Prompts into Agents' Definitions and Tasks**
    - **Prompt Refinement:**
      - Refine your GPT prompts based on insights gained from Assignments 0-2 to better suit the development of a more sophisticated game.
    - **Integration:**
      - Embed these enhanced prompts into the agents' definitions and tasks, ensuring that each agent utilizes prompts tailored to their specific role.
- **Develop a More Sophisticated and Enhanced Game**
  - **Implementation:**
    - Use the collaborative efforts of your CrewAI agents to develop the game, ensuring that all aspects—game logic, UI/UX, user input handling, code integration, and testing—are addressed.
  - **Adherence to Template:**
    - Ensure that the game strictly follows the game template from A1 to maintain consistency and compatibility.
  - **Mobile Device Compatibility:**
    - Enhance the game to be **mobile-friendly** by:
      - **Adding Support for Touch Inputs:** Optimize controls for touchscreen devices.
      - **Responsive UI:** Adjust the user interface based on different screen sizes and orientations.
    - **Testing:**

- - - ■ Use browser developer tools to simulate mobile environments and ensure the game performs well across various devices.
    - ○ **Images and Sounds (manual download)**
      - ■ For completeness, you can manually download image/sound assets and integrate them into your games.
      - ■ You could also set up agents in your crew to generate/download image and sound assets as well.
- ● **Documentation and Reporting**
  - ○ **Agent Interaction Documentation:**
    - ■ Document how the agents interact and contribute to the game development process. This includes detailing the communication flow and how tasks are delegated and integrated.
  - ○ **Comprehensive Report:**
    - ■ Prepare a detailed report (2-3 pages) that includes:
      - ■ **Game Design and Objectives:** Overview of the game's concept, objectives, and key features.
      - ■ **Roles of Different Agents:** Description of each agent's role and responsibilities in the development process.
      - ■ **Challenges Faced and Solutions Implemented:** Discussion of any obstacles encountered during development and how they were addressed.
      - ■ **Enhanced GPT Prompts:** Explanation of the refined prompts used and how they improved the agents' performance.

Deliverables:

- ● Game files
  - ○ Game.html, game.css, game.js, images/* .png or .jpg, sounds/* .mp3 or .wav (JavaScript and CSS could also be within game.html).
- ● Video:
  - ○ Recorded video demo of game in action
  - ○ Recommended formats: .mp4 | .wav
  - ○ Link to your Git repository with all of your latest changes.
- ● A report (2-3 pages) including:
  - ○ **Game Design and Objectives:** Detailed description of the new game's concept and goals.
  - ○ **Roles of Different Agents:** Explanation of each agent's role and how they contributed to the development process.
  - ○ **Challenges Faced and Solutions Implemented:** Insight into any difficulties encountered and the strategies used to overcome them.
  - ○ **Agent Interaction Documentation:** How agents collaborated and coordinated their tasks.
  - ○ **Enhanced GPT Prompts:** The refined prompts used for each agent and their impact on the game development.