🗒️ Client Notes & Recommendations

Initial Client Interaction Notes

- **Target Audience**: The tool is designed for second-year undergraduate students.
- **Objective**:
  - Engage students and provide a platform to practice coding, logical thinking, and understanding the order of code execution.
  - The focus is on being an **educational tool**, not a full platform—no need for video or document integration.
- **Problem Format**:
  - Standard Parsons problems with no variations.
  - Problems consist of complete blocks of code with correct syntax; users are expected to rearrange code blocks, not write their own code.
- **Purpose**:
  - Aimed at users who struggle with coding.
  - Users interact by rearranging code blocks in the correct order to solve the problem.
  - The web app generates Parsons problems, specifically related to data analytics, and provides an avenue for users to engage with the content.
- **Constraints and Features**:
  - Users are limited to a certain number of attempts.
  - The UI will feature drag-and-drop functionality, potentially utilizing a library to enhance the user experience.
  - The app will be categorized by context, categories, and topics relevant to data analytics, maintaining an easy-to-use interface.
  - Problems are generated based on user input through ChatGPT, and all code is in Python.
- **Feedback and Analytics**:
  - Users receive feedback on submitted solutions similar to what an IDE would provide.
  - The application provides basic analytics such as the number of problems solved, generated, and time taken to solve each problem.
  - Problem status indicators include “finished,” “in progress,” and “new/unchecked.”

Second Client Interaction Notes

- **Problem Generation and Selection**:
  - Users can select a topic, then a context, and click submit to generate a problem based on the inputs.
  - Problem descriptions will be provided alongside the problem itself.
  - Users have the option to cancel a generated problem and choose to either generate a new one or solve the existing problem.
- **User Interface and Experience**:
  - The application will feature dropdown menus with categories predetermined by the client.
  - Contexts can be user-defined, allowing creative prompts like scenarios about koalas or other interesting topics.
  - The web app will maintain simplicity across multiple pages to ensure ease of use.
- **Feedback Mechanics**:
  - Users will receive feedback that mimics an IDE’s response if the code were executed directly.
  - The bare minimum feedback will include boolean checks to indicate correctness.
- **ChatGPT Integration**:
  - For the proof of concept, the application will use the ChatGPT chat interface instead of direct API integration.
  - The goal is to demonstrate the correct output generation for problems and deliver it back to the frontend seamlessly.