**DESIGN:**

URL:

**Scenarios**

**Scenario 1:**

A frontend developer needs a responsive design for a login page, including username and password entry, a “remember” and “forget password” option, and more at the start of their design process. Instead of writing the HTML, CSS, and JavaScript from scratch, they describe the requirements to the UI Code Assistant on the Generate page, which generates the code instantly.

**Scenario 2:**

A UX designer in need of design review uploads a PNG mockup of their site’s completed landing page to the UI Code Assistant on the Analyze page. The tool analyzes the layout, and suggests improvements (e.g., "Increase font size for better readability") to successfully better the design in terms of accessibility and usability.

**Scenario 3:**

A frontend developer needing to verify their page’s layout in the middle of the programming process pastes their HTML/CSS code for a checkout screen into the UI Code Assistant. The tool flags inefficiencies (e.g., "Use CSS Grid to optimize layout") and suggests optimizations.

**Argument**

UI development is time-consuming and error-prone, especially for developers without lots of frontend experience. Existing tools like Bootstrap provide building blocks and can aid in speeding up the process but lack personalized feedback for a given developer’s goals. The process can be further slowed down due to lengthy code and design reviews. An AI-driven tool like UI AI can optimize the frontend development process by speeding up small, repetitive tasks through code generation as well as instant code and design feedback on demand.

**Prototype Plan**

The prototype is implemented as a web app known as AI UI with the goals outlined above. The design has three main components to provide well-rounded frontend support:

* Code Generator: Text area for describing components + dropdown for framework (HTML/React/Vue). Outputs code snippets (as in generator.html).
* Design Analyzer: Image file upload for designs/URL input for live sites. Returns AI feedback (analyze.html).
* Code Review: Paste actual code to receive suggestions (review.html).

**Socratic Questioning**

* + Use Socratic questioning to critique your scenarios and the argument
    - Ensure you cover all four types of questions covered in class 8
    - Document questions and answers

**Implementation**

**Walkthrough**

A screenshot of a computer

AI-generated content may be incorrect.

Front Page

Regular front page layout. Theme is clean and modern inspired by other AI tools like ChatGPT, DeepSeek, and UXPilot.

A screenshot of a computer screen

AI-generated content may be incorrect.

Quick UI search bar on front page has jQuery autocomplete for popular queries users may be asking.

A screenshot of a computer

AI-generated content may be incorrect.

UI Code Generator

“Generate HTML/JavaScript code for standard components such as login, shopping cart, etc.” Center panel for custom code generation, side buttons add code for popular components into the code box.

A screenshot of a computer

AI-generated content may be incorrect.

Ex. pressing “data table” will provide premade code and a preview (preview does not work for all items as they’re bootstrap which I couldn’t get to work in the preview lol oops)

A screenshot of a computer

AI-generated content may be incorrect.

Design Analysis

“Analyze your UI design and provide suggestions.” Uploading picture or linking website and pressing generate custom AI report on the right (right now dummy AI text).

A screenshot of a computer

AI-generated content may be incorrect.

Code Review

“Analyze your UI code and determine any inefficiencies, improvements.” Here instead of UI layout images, paste code to receive tips and reviews. Like above, not actually AI but should generate dummy AI review.

**Testing**

**Interface Metrics**

**Learnability**

|  |  |
| --- | --- |
| **Metric** | **Examples in Project** |
| **Gulfs of Execution** | * Explicit sidebar on side at all times to show how to navigate. |
| **Affordance** | * Buttons like "Generate Study Plan" and "Generate Weekly Schedule" are designed to look clickable/show selectability on hover. * Input fields are clear and method is evident:   + Self-entry text fields are white, while boxes where the TOOL will put text are tinted to indicate no clickability |
| **Signifier** | * Appropriate icons for each sidebar heading gives idea of each page (ex. code symbol for the code generator, code doc symbol for reviewer, stat symbol for analysis, etc.) * Page you are on is highlighted in sidebar. * Popular iconography for AI is used to show that it is an AI-driven tool (magic wand, sparkles, and lightning bolts are all rapidly becoming signifiers for AI). |
| **Gulfs of Evaluation** | * Quick and concise error messages upon erroneous behavior (ex. when pressing “generate” without writing any text in any of the AI query boxes) |

**Interface Architecture**

* + Describe your interface architecture using class 7 concepts:
    - States and events
    - Model-View-Controller
    - View hierarchies

**Pointing**

* + Discuss how pointing is used (class 8)