



# Quiz Quest

Here's a breakdown of the provided project, focusing on the front-end aspects relevant to a quiz website:

**Project Title:** Quiz Platform

## Overall Project Goal:

Develop a web-based quiz application that allows users to take quizzes, view results, and potentially manage their profiles.

## UI Design:

- **Reference UI:**
  - Utilize layout and style principles from:
    - User Facing UI: <https://artsnob.webflow.io/>
- **Specific UI Requirements:**
  - Responsive design for various screen sizes.
  - Consistent color scheme and typography (energetic and engaging).
  - Smooth user experience for quiz taking, result viewing, and navigation.
  - Visual representation of quiz questions, options, and results.

## Front-End Functionality:

- **Quiz Taking:**
  - Display quiz questions and answer options.
  - Handle user selection of answers.
  - Provide immediate feedback (optional, front-end only).
  - Track user progress through the quiz.
  - Submit quiz answers.
  - Display quiz results (score, correct/incorrect answers).
- **User Profile (Optional):**

- User Login Page.
- User Profile Page (view quiz history/results).
- **Navigation:**
  - Home Page (featured quizzes, quiz categories).
  - Quiz Listing Page (list of available quizzes).
  - Quiz Details Page (quiz description, instructions).
  - Results Page.
  - Contact Us Page.
  - About Us Page.
  - Error Pages.

### **Application Requirements (Front-End Focus):**

- **User-facing pages:**
  - Home Page
  - Quiz Listing Page
  - Quiz Details Page (Rules)
  - Quiz Taking Page (combined with Registration Page)
  - Results Page
  - User Registration Page.
  - User Login Page.
  - User Profile Page (simplified results/history).
  - Contact Us Page.
  - About Us Page.
  - Error Pages.

### **Project Functionality (Front-End Focus):**

- User registration and login. (optional)
- Browse available quizzes.
- View quiz details.
- Take quizzes.
- View quiz results.
- Manage user profile (simplified to view results).

### **State Management:**

- Implement state management using React Context, Redux, or similar (for quiz data, user answers, results).

### **Data Fetching:**

[frontendQuiz.db](#)

**Form Validation:**

- Implement client-side validation for user registration and login forms.