**GEU Playground Web App Development Plan**

**Introduction:**  
GEU Playground is a web application designed to encourage and motivate students in their placement preparation. The platform will offer comprehensive resources for coding, DSA, programming basics, aptitude, and more.

**Step 1: Basic Setup**

* **Theme and Branding:**
  + Choose and apply theme colors.
  + Design and implement the logo.
  + Create and style buttons and other UI elements.

**Step 2: Landing Page**

* **Content and Layout:**
  + Design the landing page to effectively introduce the platform.
  + Highlight the key features and benefits for students.
  + Include clear calls-to-action to engage users.

**Step 3: Authentication Pages**

* **Login and Signup:**
  + Develop and implement login and signup functionality.
  + Ensure user authentication is secure and user-friendly.

**Step 4: Hero Section**

* **Main Features:**
  + Design the hero section to showcase the platform’s core offerings.
  + Clearly communicate what students can gain from using the platform.

**Step 5: Component Development**

* **Coding Journey Component:**
  + Provide resources and guidance on starting with programming languages and mastering them.
* **DSA Component:**
  + Offer 150 coding questions:
    - 80 basic
    - 50 medium
    - 20 hard
* **Theory Subjects Component:**
  + Cover essential theory subjects for placement preparation:
    - OOPs
    - DBMS
    - OS
    - Networking
* **Aptitude Section:**
  + Include practice materials for reasoning, aptitude, verbal skills, grammar, etc.

**Step 6: Student Dashboard**

* **Features:**
  + Create a student profile section to manage personal data and progress.
  + Implement a ranking system to compare students' performance with peers.

**Step 7: Compiler Development**

* **Programming Compilers:**
  + Develop a compiler for programming languages.
  + Create a separate compiler for web development (HTML, CSS, JS).

**Step 8: Friendly Match Contest**

* **Functionality:**
  + Allow users to create contests with selected DSA questions and time limits.
  + Enable users to invite teammates for competitive matches.

**Step 9: Web Development Section**

* **Course Structure:**
  + **Basic Web Development:**
    - HTML, CSS, JS, Tailwind
    - Set time limits for theory and practice questions (e.g., HTML in 5 days, CSS in 3 days).
  + **Intermediate Web Development:**
    - React
  + **Advanced Web Development:**
    - Backend development

**Step 10: Resume Builder**

* **Features:**
  + Implement a resume builder tool.
  + Include an ATS (Applicant Tracking System) checker to optimize resumes for job applications.

**Future Updates**

* **Alumni Section:**
  + Add a section where placed students can share their interview experiences and tips.