Domain Core Team Member Recruitment 2025-26

Web Development Team - Task Round

Guidelines:

- The answers to the below question are to be submitted in a PDF document named **DomainName_FullName**, Example: **WebDevelopment_ElonMusk** in the google form.
- You must create a GitHub repository and **share the GitHub repo's link** in the Google Form.
- Original work is mandatory. Any form of plagiarism will result in disqualification. Your submission will be checked using an AI plagiarism detector. If your document is found to be over 35% AI-generated, you will be disqualified from the recruitment process.

➢ Answer the following question

Mention the Web-related frameworks and technologies you are familiar with. Mention 2 or 3 web based projects that showcase your expertise in the field. Note : The Projects should have an adequate description about the them, the ideology used for the project and the proper description of any parts of the code that you feel, made the code efficient.

➢ Task :

Design and build an original, visually creative, and functional GDGC Member profile Showcase Application.
This task is divided into two:
- Frontend Task
- Backend Task

Candidates must complete both sections.

➢ **Task1:Frontend Task:**

 Create an original, and creative UI to display GDGC member profiles.

● **Deliverables:**

○ Splash Screen

- Shows app name + logo.

- Short animation (your choice).

- Auto-redirects to the main page.

○ Member Cards

- Grid layout.

- Each card must include: photo, name, role, skills, short bio.

- Use your own design style (colors, shapes, layout).

○ Search & Filters

- Search by name or bio.

- Add at least two filters (role, skills, location).

○ Light/Dark Mode

- Toggle, with stored preference.

- Affects full UI including splash.

○ Data Fetching

- Fetch member list from backend /members.

- Show loading and error states.

   Deployment Link: Must

➢ **Task2:Backened Task**

Build a small API to provide member data for the frontend.
Required Endpoints
1.GET /members (Returns a list of all members in JSON.)
2.GET /members/:id (Returns details of one member.)
Github Link: Must