## 📚 Group Project Specification

**Title:** ***Dynamic Community Portal Website using Node.js, Express & EJS***

---

### 📝 Project Overview

You are tasked with working as a team to build a **Community Portal Website** using:

- Node.js as the backend environment
- Express.js as the server framework
- EJS as the templating engine

The portal will consist of 5 structured web pages:

- Home
- About
- Events
- Contact
- Thank You

The final submission must reflect:

- Clean code
- Proper templating with partials
- Dynamic page rendering
- Clear collaboration history via Git
- Proper folder structure

---

### 📦 Project Scope

Your group must build a **fully functioning Node.js + EJS web application**, consisting of:

| Page | Description |
|---|---|
| **Home** | Welcome message + placeholder for upcoming events (dynamic later) |
| **About** | Info about the team or organization (from an array in the backend) |
| **Events** | Displays all events with title, date, location, and image (from an array) |
| **Contact** | Simple contact form (name, email, message) – stores to in-memory array |
| **Thank You** | Confirmation page after form submission |

---

### 👥 Group Requirements

- Group size: **4–5 members**
- **Groups will be auto assigned.**
- Assign at least one responsibility to each member:
    - **Team Lead**
    - **Backend Developer**
    - **Frontend Developer**
    - **Data Manager**
    - **Documentation Manager**

All members must contribute to both frontend and backend elements.

---

### 📂 Folder Structure (Provided)

You will receive a full **starter project** with the following structure:

```
/community-portal
├── /public
│   ├── /css/style.css
│   └── /images/art.jpg, tech.jpg, music.jpg
├── /views
│   ├── /partials/header.ejs, footer.ejs
│   └── /pages/home.ejs, about.ejs, events.ejs, contact.ejs, thankyou.ejs
├── /routes/pageRoutes.js
├── app.js
├── package.json
├── .gitignore
├── README.md
```

---

### 🚀 Technologies Used

- Node.js (v18+)
- Express.js (v4+)
- EJS (v3+)
- CSS or Bootstrap 5 (optional)
- Git & GitHub
- **nodemon** for live development

---

### ⚙️ Nodemon Setup

This project is configured to use nodemon for development.

Install dependencies:

```
npm install
```

Start development mode:

```
npm run dev
```

package.json includes:

```
"scripts": {
  "start": "node app.js",
  "dev": "nodemon app.js"
}
```

---

### 📋 Functional Expectations

| Feature | Details |
| --- | --- |
| **Express Routing** | Modular route setup in /routes/pageRoutes.js |
| **Dynamic EJS Templates** | Use <%= %> and <%- include %> for dynamic content + partials |
| **Static Assets** | Images and CSS stored in /public and served correctly |
| **Contact Form** | Uses POST method; submission is stored in a temporary array |
| **Thank You Page** | Display after submitting the contact form |

---

### ✨ Middleware and Implementation

The provided boilerplate includes:

- Pre-set routes
- EJS templates with placeholder content
- Sample header/footer partials

It does **not** include logic like loops, conditionals, or data rendering — that must be implemented by the team.

---

### 🗓️ Submission Instructions

- 🗓️ **Deadline:** 14 May 2025
- 📍 **Platform:** Moodle
- ⏰ **Upload your .zip at least one hour before the deadline**

**Your Moodle submission must include:**

- ✅ .zip file of the project (excluding node_modules)
- ✅ Final README.md
- ✅ .gitignore file
- ✅ Fully working folder structure
- ✅ Screenshots inside README.md (optional)
- ✅ Separate Peer Evaluation Form (submitted individually)

---

### ❌ Do NOT Include

- node_modules/ — must be listed in .gitignore
- .env or hidden config files (not applicable here)

---

### 📄 README.md Requirements

Your final README must include:

- 📌 Project Title and Overview
- 🔧 Technologies Used
- 👨‍💻 Team Members and Roles
- 🛠️ Setup Instructions (npm install, npm run dev)
- 🖼️ Screenshots (optional)
- 📝 Reflection (optional)

---

### 📊 Grading Rubric

| **Criteria** | Maximum Mark | **Weight** |
|---|---|---|
| **Dynamic Rendering with EJS** | 4 | 25% |
| **Routing and Page Structure** | 4 | 20% |

| | | |
|---|---|---|
| **Styling and Layout (CSS/Bootstrap)** | 4 | 15% |
| **Functional Contact Form** | 4 | 10% |
| **Project Structure & Code Readability** | 4 | 10% |
| **GitHub Collaboration (Commits)** | 4 | 10% |
| **Peer Evaluation + Submission Quality** | 4 | 10% |

---

### 🧩 Peer Evaluation via Moodle Workshop

Peer evaluation for this project will be conducted using the **Moodle Workshop activity**.

**What You Will Do:**

- You will **evaluate each of your teammates** using a structured rubric.
- Ratings will be on a **1–5 scale**, and comments may be required.
- Each group member will evaluate all **other** members (not themselves).
- This activity will be set up to **only show your group members**.
- Your assessment may be **anonymous**, at the instructor's discretion.

Each group member must complete a peer evaluation with:

| Category | Rating (1–5) |
|---|---|
| **Participation** | |
| **Quality of Work** | |
| **Responsibility** | |
| **Collaboration** | |
| **Problem-Solving** | |