

Team Role Assignment & Detailed Responsibilities

Project: Portfolio Manager (REST API + Frontend)

1. John (Team Leader) – Project Manager + Git Admin

Main Focus: Workflow & Collaboration

Responsibilities:

- 1. GitHub Setup**
 - Create the GitHub repo: `portfolio-manager-project`
 - Initialize with a README and `.gitignore`
 - Set project visibility (public or private)
 - Add all team members as collaborators
 - 2. Branching Strategy**
 - Create the `main` branch and protect it (no direct commits)
 - Guide teammates to create feature branches like `backend-api`, `frontend-ui`, etc.
 - 3. Pull Request (PR) Reviews**
 - Review PRs created by others before merging to `main`
 - Ensure proper commit messages and clean code
 - 4. Collaboration Tools**
 - Set up the Trello board (optional)
 - Organize tasks and help resolve blockers
 - Host short daily standup meetings (online or offline)
 - 5. Deadline Management**
 - Ensure everyone meets their goals on time
 - Help with testing, communication, and project wrap-up
-

2. Lily – Backend Lead (Node.js + MySQL)

Main Focus: API Development (Express + MySQL)

Responsibilities:

- 1. Backend Project Setup**
 - Create `server.js` and initialize the Node.js project
 - Install dependencies:

```
bash
CopyEdit
npm install express mysql2 cors body-parser
```
- 2. Define API Routes**

- Create routes to:
 - GET /portfolio → Get all items
 - POST /portfolio → Add new item
 - DELETE /portfolio/:id → Delete an item
- 3. **MySQL Connection**
 - Use mysql2 to connect to the database:
- 4. **CRUD Logic**
 - Write SQL queries inside each route (insert, select, delete)
- 5. **Return JSON**
 - Ensure each API sends a clean JSON response:

```
js
CopyEdit
const db = mysql.createConnection({...});
```

```
json
CopyEdit
{ message: "Item added successfully" }
```

3. Miry – Database Designer

Main Focus: MySQL Schema & Queries

Responsibilities:

1. **Design Database Schema**
 - Create minimal schema fields:
2. **Create Database**
 - Set up local MySQL DB (e.g., portfolio_db)
 - Create tables using CREATE TABLE script
3. **Write SQL Queries**
 - Prepare reusable queries for:
 - Inserting items
 - Selecting all items
 - Deleting by ID
4. **Collaborate with Backend**
 - Share schema details with Lily (Backend Lead)

```
sql
CopyEdit
id INT AUTO_INCREMENT PRIMARY KEY,
type VARCHAR(50),
ticker VARCHAR(10),
amount INT,
value DECIMAL(10, 2)
```

- Help troubleshoot SQL errors during development

4. Peter – Frontend Developer

Main Focus: HTML + CSS + JavaScript UI

Responsibilities:

1. Build Page Layout

- Create a simple HTML structure with:
 - Table to list portfolio items
 - Form to input new items

2. Form & Buttons

- Implement form with input fields: ticker, type, amount, value
- Add submit button for POST
- Add delete buttons beside each row

3. Use `fetch()` for API Calls

- Connect to the backend using:

```
js
CopyEdit
fetch('/portfolio', { method: 'GET' })
```

- Handle POST and DELETE as well

4. Style with CSS

- Use CSS to improve visuals and make it responsive

5. Mike – UI & Graphs

Main Focus: Visualization & Styling Enhancements

Responsibilities:

1. Use `Chart.js`

- Integrate `Chart.js` to show:
 - Pie chart of asset types
 - Bar chart of portfolio performance

2. Highlight Gains/Losses

- Use color codes to highlight:
 - Green for profits
 - Red for losses

3. Responsive UI

- Ensure the charts and UI scale properly on all devices

4. Coordinate with Peter

- Help Peter enhance the frontend using graphs & visuals
- Share JavaScript/chart code integration

Final Notes

- Everyone pushes to their own branches:
 - backend-api-lily
 - db-schema-miry
 - frontend-ui-peter
 - charts-mike
 - project-admin-john
- All merged to `main` only through **Pull Requests** after code review.