

# Development Environment Setup Overview

## 1. Setting Up AWS Environment

- Created an AWS EC2 instance and assigned an Elastic IP (13.51.149.114).
- Configured security groups to allow necessary inbound/outbound rules (SSH, HTTP, HTTPS, and required ports).
- Installed PostgreSQL, Node.js, and React on the EC2 instance.

## 2. Setting Up GitHub Repository

- Created a company GitHub account and a repository for the project.
- Cloned the company repository onto the AWS EC2 instance.
- Added SSH keys to secure access to GitHub.

## 3. Database Setup (PostgreSQL)

- Installed PostgreSQL on AWS EC2.
- Created the required database and tables (countries, roles, customer\_list, companies, employee\_list, work\_assigned\_by\_employees, cost\_vs\_ROI, etc.).
- Imported database structure and data from Palani's instructions.

## 4. Backend Setup (Node.js & Express)

- Configured the backend server using Node.js and Express.js.
- Connected the backend to the PostgreSQL database.
- Tested API endpoints using Postman (<http://13.51.149.114:8000/api/cost-roi>).

## 5. Frontend Setup (React.js)

- Installed React and set up the frontend project.

- Connected the frontend to the backend API.
- Displayed data from the cost\_vs\_ROI table in the frontend.

## 6. Deployment & Testing

- Ran the backend and frontend applications on AWS.
- Verified data retrieval in the frontend by clicking a button to display cost\_vs\_ROI table data.
- Ensured smooth functionality by testing both frontend and backend APIs.