

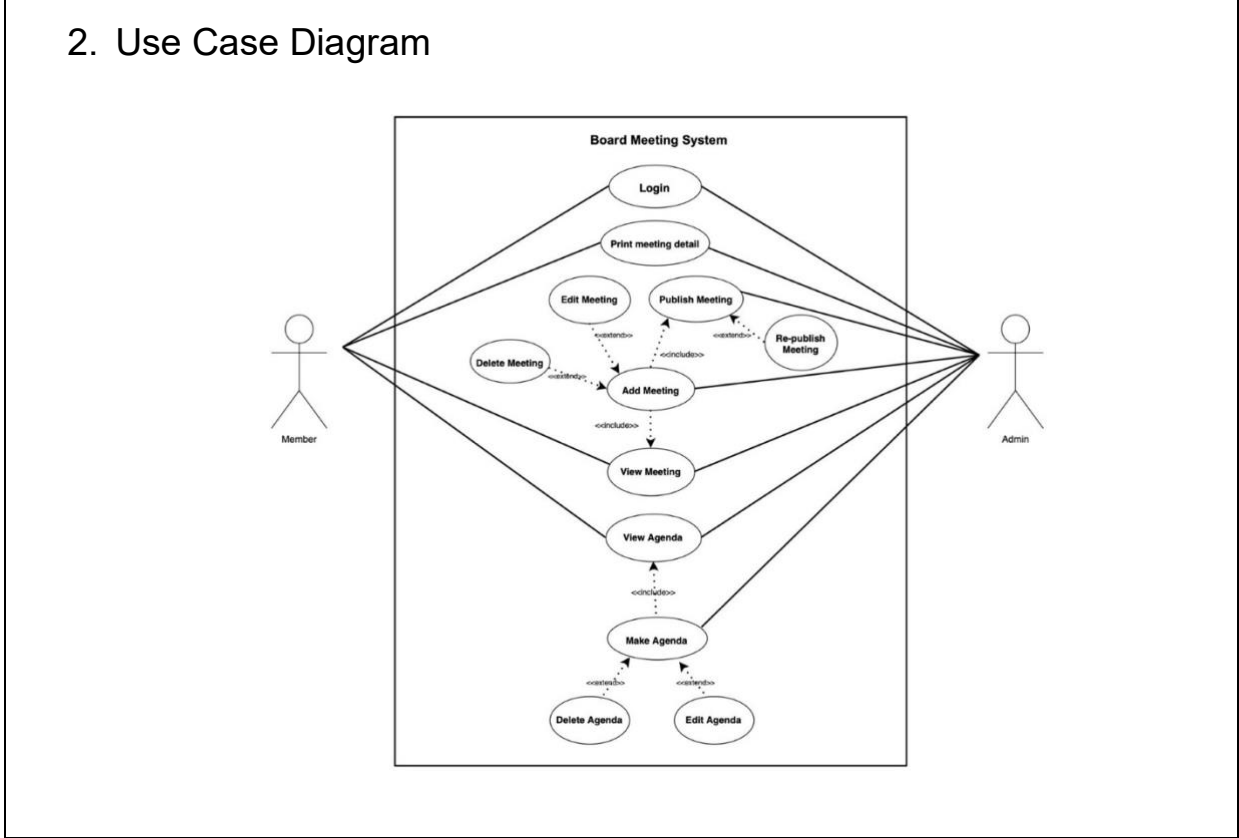
Project Overview

Project Name	Committee / Board Management System		
Background	This is a master's degree project under the topic of "Enterprise Systems DevOps".		
Objective	To apply DevOps principles with Agile methodologies and tools into real-world.		
Group Size	9 people	Duration	4 sprints, 12 weeks
Scope	<ul style="list-style-type: none"> - Develop a board meeting portal for internal used - Setup and configuration of database - Create and maintain DevOps pipelines - Conduct system testing 		
Main Features	User can login into system and create, view and edit a meeting as well as upload related documentation and print a meeting agenda.		
Technologies	<ul style="list-style-type: none"> - Front-End: Angular - Back-End: ExpressJs, RESTful API - Database: MySQL 		
Tools	<ul style="list-style-type: none"> - Azure DevOps - Git - AWS EC2 - JSON Web Tokens - Figma - Visual Studio Code - Postman 		
My Contribution	<p>My roles as a Lead Business Analyst,</p> <ul style="list-style-type: none"> - Gathered requirements from the client translating into System Requirements Specification documentation. - Created Use Case Diagram - Created User Stories - Created Activity Diagram - Conducted UI/UX research - Designed UI/UX Wireframe - Implemented Wireframe Guideline - Worked closely with Development team to ensure the product meets System Requirements - Reviewed and Hand-off product to the client in each sprint 		

Project Implementation

1. Group Structure

```
graph TD; Andy[Andy (Product Owner)] --> David[David (Project Manager)]; David -.-> Hazel[Hazel (Scrum Master)]; David --> Kotesch[Kotesch (DevOps Engineer)]; David --> BATeam[BA Team]; David --> DevTeam[Dev Team]; David --> Architect[Architect]; David --> QATeam[QA Team]; BATeam --> Sharlene[Sharlene (Lead BA)]; DevTeam --> Kaung[Kaung (lead dev)]; Kaung --> Boom[Boom]; Boom --> ShengFan[Sheng Fan]; Architect --> Riya[Riya (Lead architect)]; QATeam --> Keiron[Keiron (Lead QA)];
```



2. Use Case Diagram

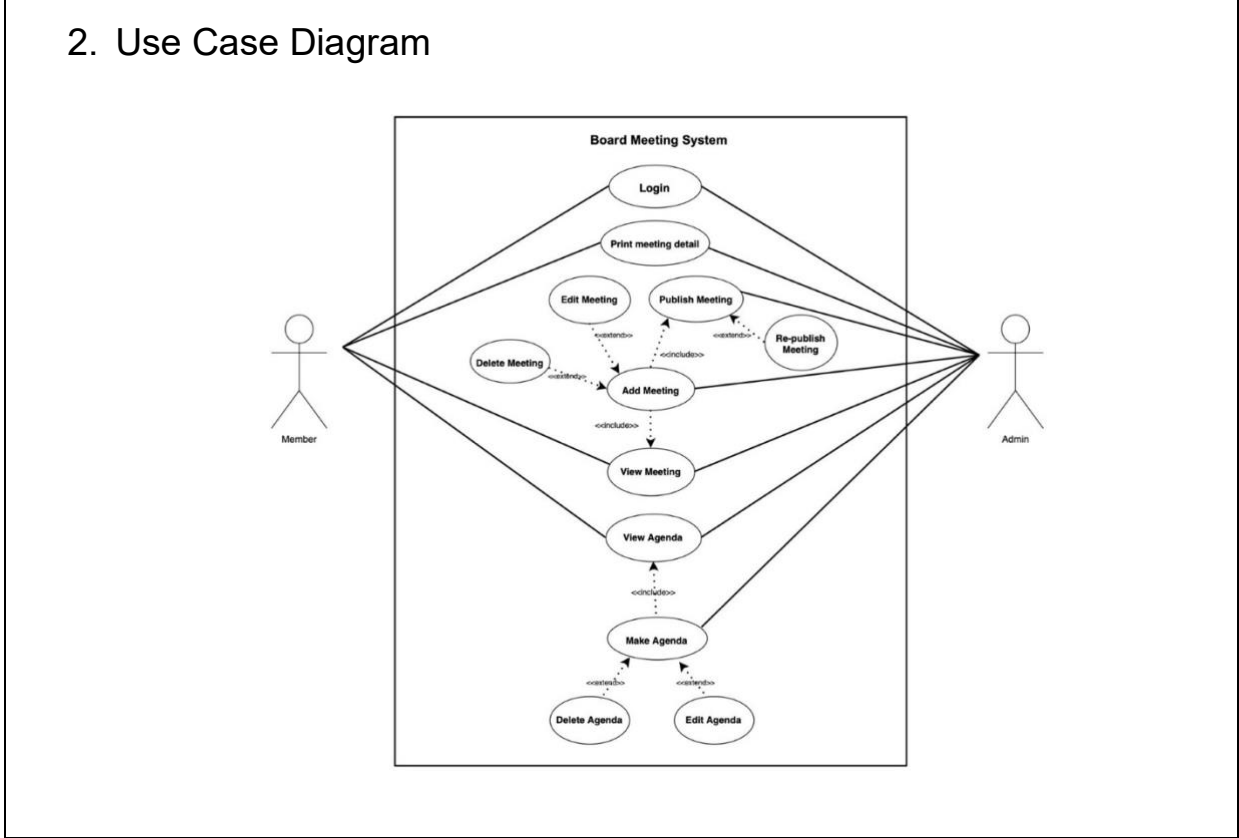
```

    usecaseDiagram
        actor Member
        actor Admin
        usecase BoardMeetingSystem as Board Meeting System
        usecase BoardMeetingSystem as Login
        usecase BoardMeetingSystem as PrintMeetingDetail as Print meeting detail
        usecase BoardMeetingSystem as AddMeeting
        usecase BoardMeetingSystem as EditMeeting
        usecase BoardMeetingSystem as DeleteMeeting
        usecase BoardMeetingSystem as ViewMeeting
        usecase BoardMeetingSystem as ViewAgenda
        usecase BoardMeetingSystem as MakeAgenda
        usecase BoardMeetingSystem as PublishMeeting
        usecase BoardMeetingSystem as RepublishMeeting as Re-publish Meeting
        usecase BoardMeetingSystem as DeleteAgenda
        usecase BoardMeetingSystem as EditAgenda

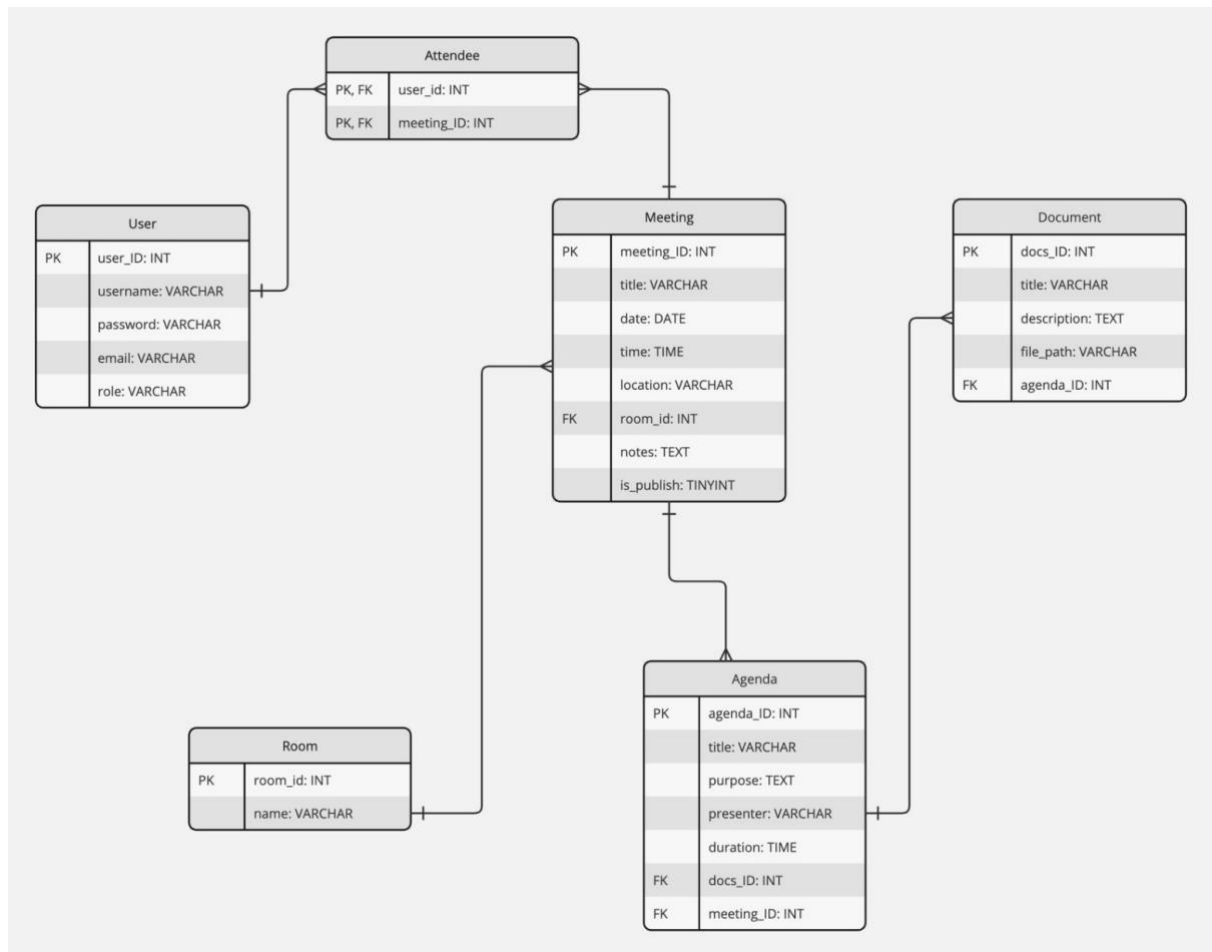
        Member --> Login
        Member --> PrintMeetingDetail
        Member --> AddMeeting
        Member --> ViewMeeting
        Member --> ViewAgenda
        Admin --> Login
        Admin --> PrintMeetingDetail
        Admin --> AddMeeting
        Admin --> ViewMeeting
        Admin --> ViewAgenda
        Admin --> MakeAgenda
        Admin --> PublishMeeting
        Admin --> RepublishMeeting
        Admin --> DeleteAgenda
        Admin --> EditAgenda

        AddMeeting -.->|<<extend>>| EditMeeting
        AddMeeting -.->|<<extend>>| DeleteMeeting
        AddMeeting -.->|<<include>>| PublishMeeting
        AddMeeting -.->|<<include>>| RepublishMeeting
        AddMeeting -.->|<<include>>| ViewMeeting
        MakeAgenda -.->|<<include>>| ViewAgenda
        MakeAgenda -.->|<<extend>>| DeleteAgenda
        MakeAgenda -.->|<<extend>>| EditAgenda
    
```

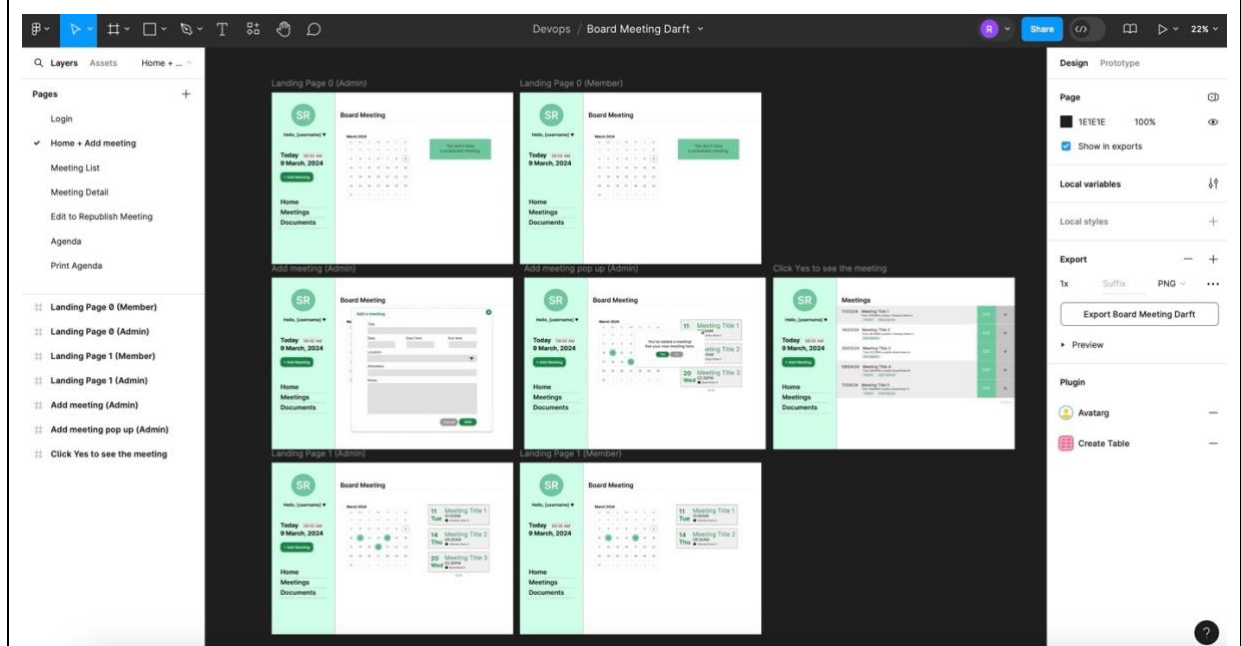
The diagram illustrates the functional requirements of a Board Meeting System. It features two actors, Member and Admin, and a central system boundary labeled Board Meeting System. The system contains several use cases: Login, Print meeting detail, Add Meeting, Edit Meeting, Delete Meeting, View Meeting, View Agenda, Make Agenda, Publish Meeting, Re-publish Meeting, Delete Agenda, and Edit Agenda. The diagram shows the relationships between these use cases, including include and extend relationships, and the actors' interactions with the system.



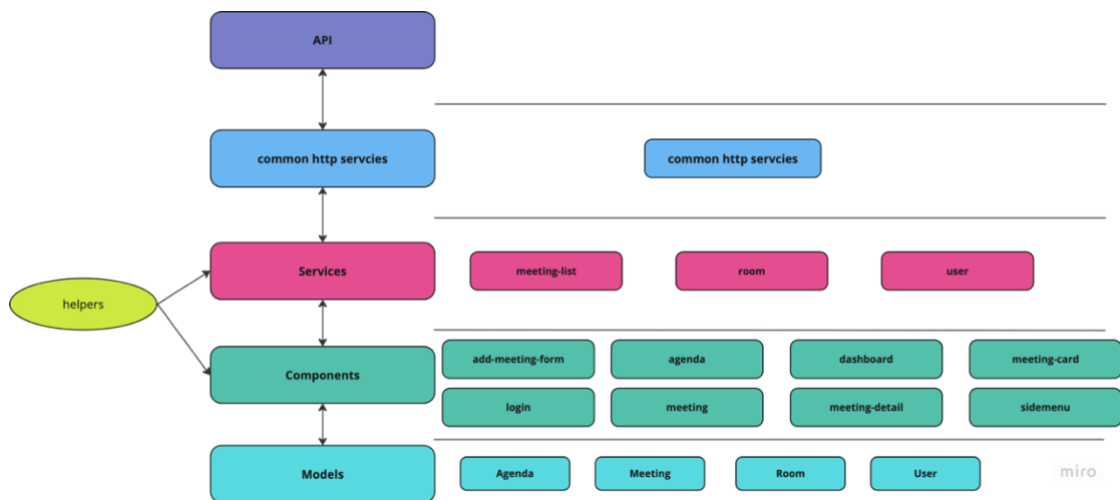
3. ER Diagram



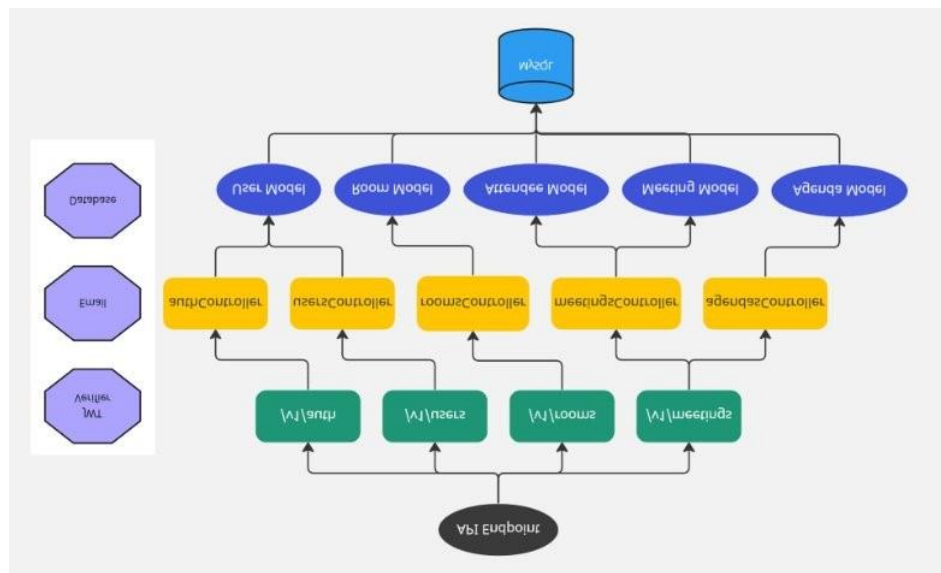
4. UI/UX Wireframe



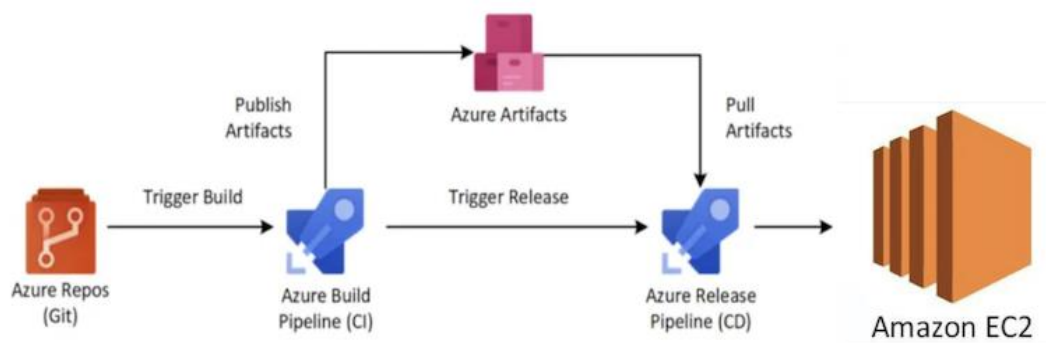
5. Front-End Architecture Overview



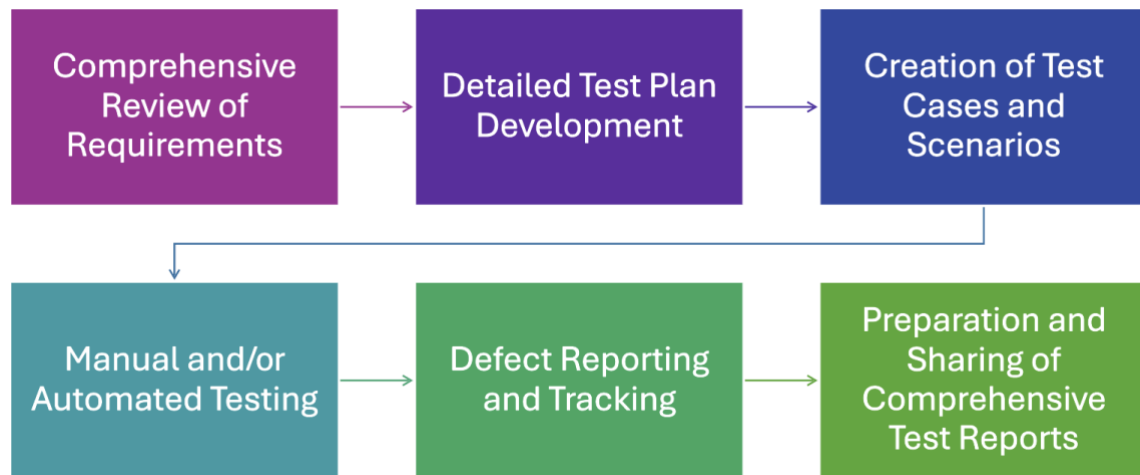
6. Back-End Structure Overview



7. CI/CD Pipeline



8. System Testing Approach



***** Final Product source code cannot be published due to integrity *****