**Project Architecture Design**
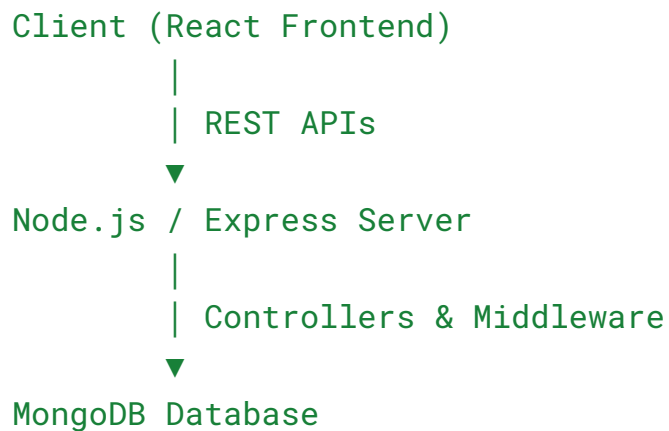
**Project Name:** Project Management System

**Account Types:**

- Admin

- User

The system follows a **Role-Based Access Architecture** where dashboards and features are accessible based on the logged-in user role.

---

## 1 High Level Architecture

```
Client (React Frontend)
        |
        | REST APIs
        ▼
Node.js / Express Server
        |
        | Controllers & Middleware
        ▼
MongoDB Database
```

---

**Components Explanation**

**Frontend**

- Developed using React.js

- Separate dashboards for Admin and User

- Kanban board with Drag & Drop functionality

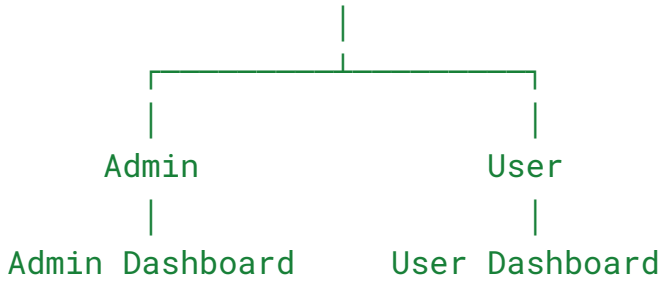- API integration using Axios/Fetch

---

**Backend**

- Built with Node.js and Express.js

- REST API architecture

- JWT-based Authentication

- Role-based Authorization
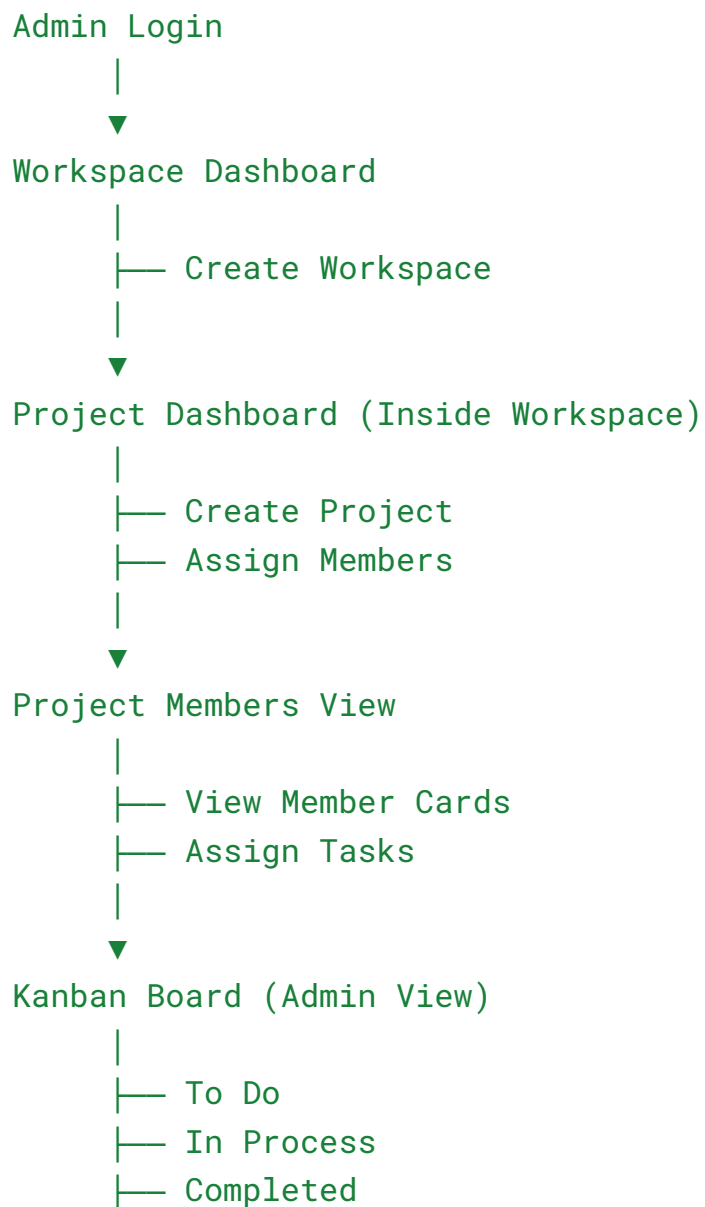
- Middleware for route protection

---

**Database**

- MongoDB (NoSQL)

- Stores:
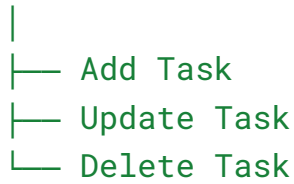
  - Users

  - Workspaces

  - Projects

  - Tasks

---

2 **Role Based Access Flow**

Login

```
                    |
        ┌───────────┴───────────┐
        |                       |
      Admin                   User
        |                       |
  Admin Dashboard        User Dashboard
```

---

### 3 Admin Architecture Flow

```
Admin Login
      |
      ▼
Workspace Dashboard
      |
      ├── Create Workspace
      |
      ▼
Project Dashboard (Inside Workspace)
      |
      ├── Create Project
      ├── Assign Members
      |
      ▼
Project Members View
      |
      ├── View Member Cards
      ├── Assign Tasks
      |
      ▼
Kanban Board (Admin View)
      |
      ├── To Do
      ├── In Process
      ├── Completed
```

```
      |
      ├── Add Task
      ├── Update Task
      └── Delete Task
```
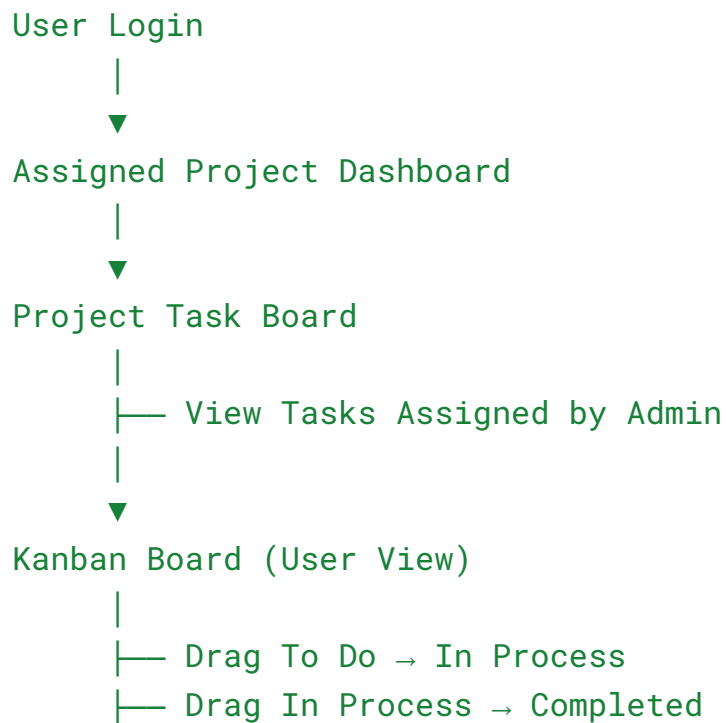
**Admin Permissions**

- Manage Workspaces

- Create/Delete Projects

- Assign/Remove Members

- Assign Tasks

- Manage Kanban Board

---

## 4 User Architecture Flow

```
User Login
      |
      ▼
Assigned Project Dashboard
      |
      ▼
Project Task Board
      |
      ├── View Tasks Assigned by Admin
      |
      ▼
Kanban Board (User View)
      |
      ├── Drag To Do → In Process
      ├── Drag In Process → Completed
```

```
        |
        └── Update Task Status Only
```

## User Permissions

- View Assigned Projects

- View Assigned Tasks

- Update Task Status via Drag & Drop

---

## 5 Module Wise Architecture

---

## Workspace Module

- Created and managed by Admin

- Works as a container for projects
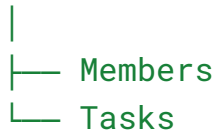
- Default dashboard is empty until created

```
Workspace
   |
   └── Projects
```

---

## Project Module
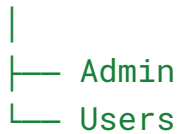
- Created inside workspace

- Members assigned by Admin

```
Project
    |
    ├── Members
    └── Tasks
```

---

**Task Module**

- Assigned to project members

- Managed using Kanban board

**Task Status Flow**

```
To Do → In Process → Completed
```

---
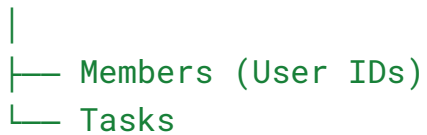
## 6 Database Architecture (Logical Structure)

```
Users Collection
    |
    ├── Admin
    └── Users


Workspace Collection
    |
    └── Projects


Project Collection
    |
    ├── Members (User IDs)
    └── Tasks


Task Collection
    |
```

```
├── Title
├── Assigned User
├── Project ID
└── Status
```

---

## 7 Request Flow Architecture

```
Frontend Action
       |
       ▼
API Request
       |
       ▼
Authentication Middleware (JWT)
       |
       ▼
Controller Logic
       |
       ▼
Database Operation
       |
       ▼
Response → Frontend Update
```

---
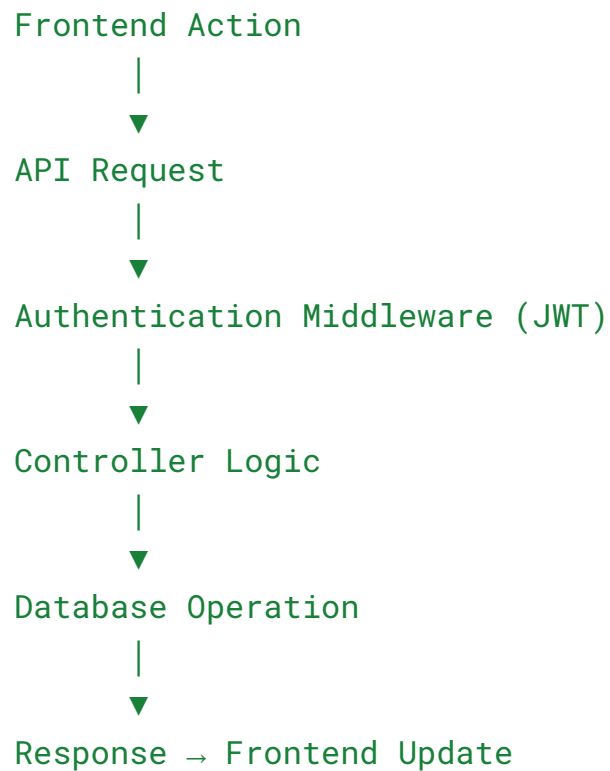
## 8 Kanban Board Architecture

```
              Kanban Board

    ┌──────────┬─────────────┬──────────────┐
    │  To Do   │  In Process │  Completed   │
    └──────────┴─────────────┴──────────────┘
```

**Admin Capabilities**

- Add Task

- Update Task

- Delete Task

- Change Status

**User Capabilities**

- Change Status Only

---

## 9️⃣ Security Architecture

- JWT Authentication

- Role-Based Authorization

- Protected Admin Routes

- Middleware Validation

- Token Verification on Each Request

---

## 🔟 System Flow Summary

```
Admin Creates Workspace
        |
Admin Creates Project
        |
Admin Assigns Members
        |
```

```
Admin Assigns Tasks
        |
Users View Assigned Projects
        |
Users Update Task Status
        |
Admin Monitors Progress via Kanban Board
```