**Team leader**

S Yamuna

**Team members**

S kanniyammal

E meenatchi

P yuvasree

Here is a full documentation for a news app using JavaScript and Node.js:

# Table of Contents

1. #introduction

2. #system-requirements

3. #technology-stack

4. #project-structure

5. #api-documentation

6. #database-schema

7. #security-considerations

8. #deployment-instructions

9. #testing-strategy

10. #troubleshooting-guide

# Introduction

The news app is a web-based application that allows users to view and share news articles. The app is built using JavaScript and Node.js, and utilizes a MongoDB database to store news articles.

# System Requirements

- Node.js (version 14.x or higher)

- MongoDB (version 4.x or higher)

- JavaScript (ECMAScript 6 or higher)

# Technology Stack

- Frontend: JavaScript (ECMAScript 6 or higher), HTML5, CSS3

- Backend: Node.js (version 14.x or higher), Express.js (version 4.x or higher)

- Database: MongoDB (version 4.x or higher)

# Project Structure

The project is structured as follows:

- app.js: The main application file.

- routes/: Directory containing route files.

- models/: Directory containing model files.

- controllers/: Directory containing controller files.

- views/: Directory containing view files.

- public/: Directory containing static files.

- config/: Directory containing configuration files.

# API Documentation

The API is documented using Swagger. The API documentation can be found at /api-docs.

# Database Schema

The database schema is as follows:

- articles collection:

- \_id: ObjectId (primary key)

- title: String

- content: String

- author: String

- created\_at: Date

- users collection:

- \_id: ObjectId (primary key)

- username: String

- email: String

- password: String

- created\_at: Date

# Security Considerations

The app uses the following security measures:

- Authentication: The app uses JSON Web Tokens (JWT) for authentication.

- Authorization: The app uses role-based access control (RBAC) for authorization.

- Data encryption: The app uses HTTPS to encrypt data in transit.

- Password hashing: The app uses bcrypt to hash passwords.

# Deployment Instructions

To deploy the app, follow these steps:

1. Clone the repository.

2. Install dependencies using npm install.

3. Create a MongoDB database and add the connection string to the config/database.js file.

4. Start the app using node app.js.

5. Access the app at http://localhost:3000.

# Testing Strategy

The app uses Jest for unit testing and Cypress for end-to-end testing.

# Troubleshooting Guide

Common issues and solutions:

- Issue: Unable to connect to MongoDB.

Solution: Check the connection string in config/database.js.

- Issue: Unable to authenticate.

Solution: Check the JWT secret key in config/auth.js.

- Issue: Unable to access certain routes.

Solution: Check the RBAC configuration in config/roles.js.