title

Subtitle



Author

TAble of  
Contents

Emphasis Heading 1 1

Heading 2 1

Heading 2 2

Designing the Applications

## Planning:

Analyze Requirements: I carefully reviewed the provided problem statement to understand the requirements for the full-stack web application.

Identify Features: I identified key features such as CRUD operations for articles, filtering articles, and implementing both backend and frontend components.

Architecture Planning: I determined the architecture of the applications, including the technology stack (Ruby on Rails, React, HTML), database design, and API design.

## Backend Application (Ruby on Rails):

Model Design: I designed the **Article** model with attributes: **title** (string), **body** (text), and **published** (boolean, default: false).

Migration and Validation: I created a migration file for the **Article** model to set up the database table and added validations to ensure data integrity.

Controller Implementation: I implemented controller actions for CRUD operations on articles, including GET all articles, GET by ID, POST, PUT, and DELETE.

Testing: I wrote unit tests for the **Article** model and controller actions using a testing framework like RSpec.

Frontend Application (React):

Component Development: I developed React components for displaying articles, adding/editing articles, and filtering articles based on their published status.

API Integration: I integrated the React client with the Ruby on Rails backend by making HTTP requests to the defined API endpoints.

Styling and UI Design: I designed the UI components using JSX and Tailwind, creating a user-friendly interface.

Testing: I wrote unit tests for React components using **Jest** and **React Testing Library**.

## HTML Client:

Form Development: I designed forms and UI elements for CRUD operations on articles using HTML and JavaScript.

HTTP Requests: I ensured that the HTML client makes appropriate HTTP requests to the backend API endpoints for each action.

Styling: I applied CSS styles to the HTML elements to enhance the visual presentation and usability.

Testing: I wrote tests to validate the HTML and JavaScript functionality using **Jest** again.

Testing and Integration

**Integration Testing**: I implemented integration tests to cover end-to-end scenarios for article creation, retrieval, updating, and deletion using a suitable testing framework: **Cypress**.

**Unit Testing vs. Integration Testing**: [Discuss the importance of unit testing and integration testing in the development process, including their respective roles and benefits.]

## Heading 2

* Use styles to easily format your Word documents in no time. For example, this text uses the List Bullet style.
* On the Home tab of the ribbon, check out Styles to apply the formatting you want with just a tap.

|  |  |  |
| --- | --- | --- |
|  | Column Heading | Column Heading |
| Row Heading | Text | 123.45 |
| Row Heading | Text | 123.45 |
|  |  |  |