Financial App Work Documentation

Table of Contents

[Project Setup: 2](#_Toc162271860)

[Frontend Development: 2](#_Toc162271861)

[Backend Development: 2](#_Toc162271862)

[Testing: 2](#_Toc162271863)

[Integration and Deployment: 3](#_Toc162271864)

[Maintenance and Documentation: 3](#_Toc162271865)

[Communication and Management: 3](#_Toc162271866)

[Finalization: 3](#_Toc162271867)

[Project Overview 4](#_Toc162271868)

[Project Team 4](#_Toc162271869)

[Project Objectives 4](#_Toc162271870)

[Tasks and Responsibilities 4](#_Toc162271871)

[Frontend Developer 4](#_Toc162271872)

[Backend Developer 5](#_Toc162271873)

[Project Manager 5](#_Toc162271874)

[Project Timeline 5](#_Toc162271875)

[Communication Channels 7](#_Toc162271876)

[Risks and Mitigation Strategies 7](#_Toc162271877)

[ Technical Risks 7](#_Toc162271878)

[ Mitigation 7](#_Toc162271879)

[ Timeline Risks 7](#_Toc162271880)

[ Mitigation 7](#_Toc162271881)

[Project Deliverables 7](#_Toc162271882)

[Conclusion 7](#_Toc162271883)

# Project Setup:

1. **Define project scope, objectives, and deliverables.**
2. **Set up development environments for frontend and backend.**
3. **Finalize technology stack and tools.**
4. **Create project timelines and milestones.**

## Frontend Development:

1. **Implement UI wireframes and designs.**
2. **Set up React project structure.**
3. **Implement UI components using React and Tailwind CSS.**
4. **Set up routing and navigation using React Router.**
5. **Integrate API endpoints for data retrieval and management on frontend.**
6. **Implement user authentication and session management on frontend.**

## Backend Development:

1. **Set up the backend API using Node.js and Express.js.**
2. **Implement API endpoints for managing transactions, users, and authentication on backend.**
3. **Integrate with the MSSQL database for data storage and retrieval on backend.**
4. **Implement user authentication using JWT tokens on backend.**

## Testing:

1. **Write unit tests for UI components on frontend.**
2. **Write integration tests for frontend API interactions.**
3. **Write unit tests for API endpoints and controllers on backend.**
4. **Write integration tests for backend database interactions.**

## Integration and Deployment:

1. **Integrate frontend and backend components.**
2. **Conduct end-to-end testing of the entire application.**
3. **Deploy frontend and backend to hosting environments.**
4. **Configure server settings and domain.**
5. **Conduct final testing and troubleshooting.**

## Maintenance and Documentation:

1. **Monitor application performance and user feedback.**
2. **Address any bugs or issues reported by users.**
3. **Plan for future updates and enhancements.**
4. **Document API endpoints and usage.**
5. **Document project setup and development processes.**
6. **Document testing procedures and results.**
7. **Document deployment configurations and procedures.**

## Communication and Management:

1. **Schedule regular project meetings to discuss progress and issues.**
2. **Use collaboration tools like Slack and GitHub for communication and collaboration.**
3. **Identify potential risks and mitigation strategies.**
4. **Regularly review project progress and adjust plans as needed.**

## Finalization:

1. **Ensure all project deliverables are completed and meet quality standards.**
2. **Review project outcomes and lessons learned.**
3. **Celebrate project completion and team achievements.**

# Project Overview

The Financial App project aims to develop a web application for personal finance management. The project consists of building both the frontend user interface and the backend API.

# Project Team

* Project Manager: MoneyMap
* Frontend Developer: Shazil Shahid
* Backend Developer: Shazil Shahid

# Project Objectives

1. Develop a user-friendly frontend interface for managing finances.
2. Implement a backend API to handle data interactions and authentication.
3. Integrate frontend and backend components to create a seamless user experience.
4. Ensure scalability, security, and performance of the application.

# Tasks and Responsibilities

# Frontend Developer

* Implement UI components using React and Tailwind CSS.
* Set up routing and navigation using React Router.
* Integrate API endpoints for data retrieval and management.
* Implement user authentication and session management.
* Write unit tests and integration tests for frontend components.
* Collaborate with the backend developer to ensure seamless integration.

# Backend Developer

* Set up the backend API using Node.js and Express.js.
* Implement API endpoints for managing transactions, users, and authentication.
* Integrate with the MSSQL database for data storage and retrieval.
* Implement user authentication using JWT tokens.
* Write unit tests and integration tests for backend routes and controllers.
* Ensure security best practices and data validation measures are implemented.

# Project Manager

* Define project scope, objectives, and deliverables.
* Create project timelines and milestones.
* Allocate resources and assign tasks to team members.
* Coordinate communication between frontend and backend teams.
* Monitor project progress and address any issues or blockers.
* Ensure project meets quality standards and is delivered on time.

# Project Timeline

* **Phase 1: Planning and Setup**
  + Define project requirements and scope.
  + Set up development environments for frontend and backend.
  + Finalize technology stack and tools.
  + Duration: [Insert Duration]
* **Phase 2: Development**
  + Frontend Development:
    - Implement UI components and navigation.
    - Set up user authentication and session management.
    - Duration: [Insert Duration]
  + Backend Development:
    - Set up API endpoints and database connections.
    - Implement authentication and authorization.
    - Duration: [Insert Duration]
* **Phase 3: Testing and Integration**
  + Frontend Testing:
    - Write and execute unit tests for UI components.
    - Conduct integration tests with backend API.
    - Duration: [Insert Duration]
  + Backend Testing:
    - Write and execute unit tests for API endpoints and controllers.
    - Conduct integration tests with the database.
    - Duration: [Insert Duration]
* **Phase 4: Deployment**
  + Deploy frontend and backend to hosting environments.
  + Configure server settings and domain.
  + Conduct final testing and troubleshooting.
  + Duration: [Insert Duration]
* **Phase 5: Maintenance and Support**
  + Monitor application performance and user feedback.
  + Address any bugs or issues reported by users.
  + Plan for future updates and enhancements.
  + Duration: Ongoing

# Communication Channels

* Project Meetings: Weekly meetings to discuss progress, issues, and updates.
* Collaboration Tools: Slack for instant messaging, GitHub for code collaboration and version control.

# Risks and Mitigation Strategies

* Technical Risks: Potential issues with integration or compatibility between frontend and backend components.
  + Mitigation: Conduct thorough testing and ensure clear communication between frontend and backend teams.
* Timeline Risks: Delays in development or unforeseen obstacles impacting project deadlines.
  + Mitigation: Regularly monitor progress, adjust timelines if necessary, and allocate additional resources if needed.

# Project Deliverables

* Frontend UI components and pages
* Backend API endpoints and documentation
* Testing reports and documentation
* Deployment configurations and documentation

# Conclusion

The Financial App project aims to deliver a comprehensive solution for personal finance management. By following the outlined tasks, responsibilities, and timelines, we aim to achieve our project objectives efficiently and effectively.