**Tasking, Time Estimation, and Functional Requirements Document**

**Project Name:** Document Manager [MAPSYS Assessment]

**Objective:** Create a web application that allows users to manage and view documents listed in the provided CSV file.

**Project Team:** Maison Arroyo, Software Developer

**Project Supervisor:** Brian Callhoun, Project Manager

**Task 1: Project Setup**

**Description:** Set up the initial MEVN project environment, including the frontend and backend frameworks.

**Time Estimate:** 1 hours

**Functional Requirements:**

* Initialize a Vue.js project for the frontend.
* Create an Express.js project for the backend.
* Set up a MongoDB database.
* Set up and fork repository.
* Configure the development environment with necessary dependencies.

**Task 2: Display Documents**

**Description:** Implement the feature to display documents listed in the "Documents.csv" file.

**Time Estimate:** 6 hours

**Functional Requirements:**

* Create a Vue.js component to display the list of documents.
* Fetch document data from the backend API.
* Display document names as clickable links.
* When a user clicks on a document link, open the document for viewing.

**Task 3: Upload New Document**

**Description:** Develop the functionality to allow users to upload new documents and update the "Documents.csv" file.

**Time Estimate:** 8 hours

**Functional Requirements:**

* Design and implement a form for uploading documents.
* Handle file uploads on the backend and store documents.
* Validate uploaded documents (e.g., file type, size).
* Update the "Documents.csv" file with the new document's information.

**Task 4: Delete Existing Document**

**Description:** Implement the feature to allow users to delete existing documents and update the "Documents.csv" file.

**Time Estimate:** 6 hours

**Functional Requirements:**

* Add a delete button next to each document.
* When a user clicks the delete button, confirm the deletion.
* Remove the selected document and update the "Documents.csv" file accordingly.

**Task 5: Testing and Debugging**

**Description:** Test application and address any bugs or issues.

**Time Estimate:** 3 hours

**Functional Requirements:**

* Create test cases for each feature.
* Test different scenarios, including uploading, viewing, and deleting documents.
* Debug and fix any issues that arise during testing.

**Task 6: Documentation**

**Description:** Prepare documentation for the project.

**Time Estimate:** 2 hours

**Functional Requirements:**

* Create a README file with instructions for running the application.
* Include information about the technologies used and any dependencies.
* Document the API routes and how to interact with them.