Team 7 (Money Mate)

1. **Allan Charlier**
   * **Student ID**:033767732
   * **CSULB Email**: Allan.charlier01@student.csulb.edu
2. **Jimmy Tran**
   * **Student ID**: 028343131
   * **CSULB Email**: jimmy.tran03@student.csulb.edu
3. **Toan Le**
   * **Student ID**: 017380673
   * **CSULB Email**: toan.le02@student.csulb.edu
4. **Edouard Belhomme**
   * **Student ID**: 033767199
   * **CSULB Email**: edouard.belhomme01@student.csulb.edu
5. **Thien Nguyen**
   * **Student ID**: 031138144
   * **CSULB Email**: Thien.Nguyen02@student.csulb.edu

## Preface

This document is intended to list the design, rationale, and implementation of MoneyMate. Readers to be expected are the professor, as well as potential customers/stakeholders looking to invest.

| Version | Date | Changes |
| --- | --- | --- |
| 1.0 | 3/01/2025 | Initial Version |

### Purpose

This document serves as a guide to help understand the architecture and design behind MoneyMate, a personal budgeting app.

### Audience

The intended audience is for potential stakeholders, testers, developers, and the professor.

### Introduction

Welcome to MoneyMate, a personal budgeting manager website designed to help users keep track of their expenses easily. Set personal budgets with specific categories and take better care of your financial expenses.

### Project Overview

MoneyMate is a web-based financial management system designed for individuals. It encapsulates all personal spending into one website for tracking and easier management.

### Project Goals

* Improve budgeting habits
* Ease budget tracking ability
* Provide budgeting goals and incentives

## Glossary

* **API**: Application Programming Interface.
* **JS:** JavaScript

### User Requirements and Use Cases

Users must be able to login to their account, and view all of their budgeting categories that they have created as well as the amount they have spent per category.

Users must also have the ability to add new expenses as well as modify the current amount that they have spent between each category.

Users must have the ability to view their progress on each budget and adjust the time frame.

### User Stories

1. As a budget manager, I want to view all of my budgets that I have created so I can track my spending
2. As a budget manager, I want to add a new expense so I can be updated on all of my spending.
3. As a budget manager, I want to login to my account so I can access all of my account information.
4. As a budget manager, I want to choose the length of my budget to reflect my real world needs.
5. As a budget manager, I want to receive notifications so that I can see what percentage of my budget that I spent so far.
6. As a budget manager, I want to have automatic updates when using Paypal/Venmo so that I do not have to update manually.
7. As a budget manager, I need to change my password to protect my security.
8. As a budget manager, I need to delete an expense that I do not wish to track anymore.
9. As a budget manager, I want to enter an amount that I spent for a certain expense to update my current spending.
10. As a budget manager, I want to change my account information so I can link to another account.

***Money Mate - Adding An Expense***

| Identifier | UC-1 |
| --- | --- |
| Goal | Add new expense |
| Requirements | Allow budget managers to add details (e.g. amount, date, time frame, payment method) and save them. |
| Initiating Actor | Budget Manager |
| Participating Actor(s) | N/A |
| Pre-conditions | Budget Manager must be logged in.  Budget Category must be available. |
| Post-conditions | The expense is recorded. |
| Included Use Case(s) | Edit Expense |
| Extension(s) | Error Handling(e.g. incorrect format) |

***Table 1: Typical Course of Action***

| Seq# | Budget Manager’s Action | Money Mate’s Response |
| --- | --- | --- |
| 1 | The Budget Manager navigates to the “Add Expense” screen. |  |
| 2 |  | Money Mate displays the Expense entry screen/form |
| 3 | The Budget Manager enters expense details. |  |
| 4 |  | Money Mate validates inputs |
| 5 | The Budget Manager submits the expense entry. |  |
| 6 |  | Money Mate saves the expense, and confirms success |
| 7 |  | In case of Error, Money Mate displays an error message and prompts for correction. |

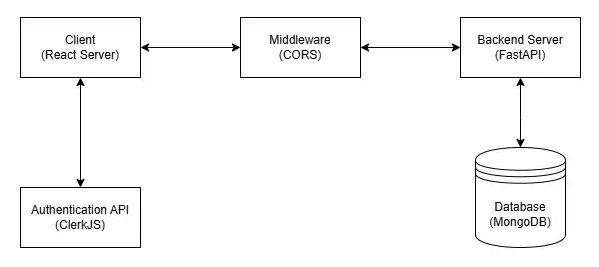
***Table 2: Alternate Course of Action***

| Seq# | Budget Manager’s Action | Money Mate’s Response |
| --- | --- | --- |
| 1 | The Budget Manager selects a category that does not exist. |  |
| 2 |  | Money Mate outputs error and remediation action, prompting the user to pick an appropriate category |
| 3 | The Budget Manager selects a valid category and the rest of expense details |  |
| 4 |  | Money Mate saves expense date to the database |

***Table 3: Exceptional Course of Action***

| Seq# | Actor’s Action | System’s Response |
| --- | --- | --- |
| 1 | The Budget Manager adds a new expense |  |
| 2 | Clicks “create” |  |
| 3 |  | Checks if entered information is acceptable |
| 4 |  | Attempts to enter information into the database |
| 5 |  | The database is unable to connect at the time |
| 6 |  | Prompts user that the request was unable to be completed |

## System Architecture



### Components

### Frontend: React with Tailwind CSS

### Backend: Python (FastAPI)

Database: MongoDB

### APIs: ClerkJS (Authentication), EmailJS (email notifications)

### Deployment Diagram

