Project Report

# 1. Introduction

## Problem Statement

Create a dashboard to analyze exchange rates between two currencies over time. Features include charts (weekly, monthly, quarterly, yearly), highest/lowest rates, and print options. The input will be a currency exchange dataset, with USD as the base currency and various other currencies (USD/INR, USD/GBP, etc.).

# 2. Technology Used

- Frontend: React with interactive Chart.js for data visualization.

- Backend: MongoDB.

- Programming Languages: Python, JavaScript.

- Framework: FastAPI.

# 3. Prerequisites

## Database Setup (MongoDB)

Install MongoDB Shell and Compass:

* macOS: brew install mongodb, brew install --cask mongodb-compass
* Windows/Linux: Follow the official MongoDB installation guides.

## Python Setup

Ensure Python 3.0+ is installed:

* macOS: brew install python
* Windows/Linux: Install from Python’s official website.

## React and Frontend Libraries

Install necessary libraries:

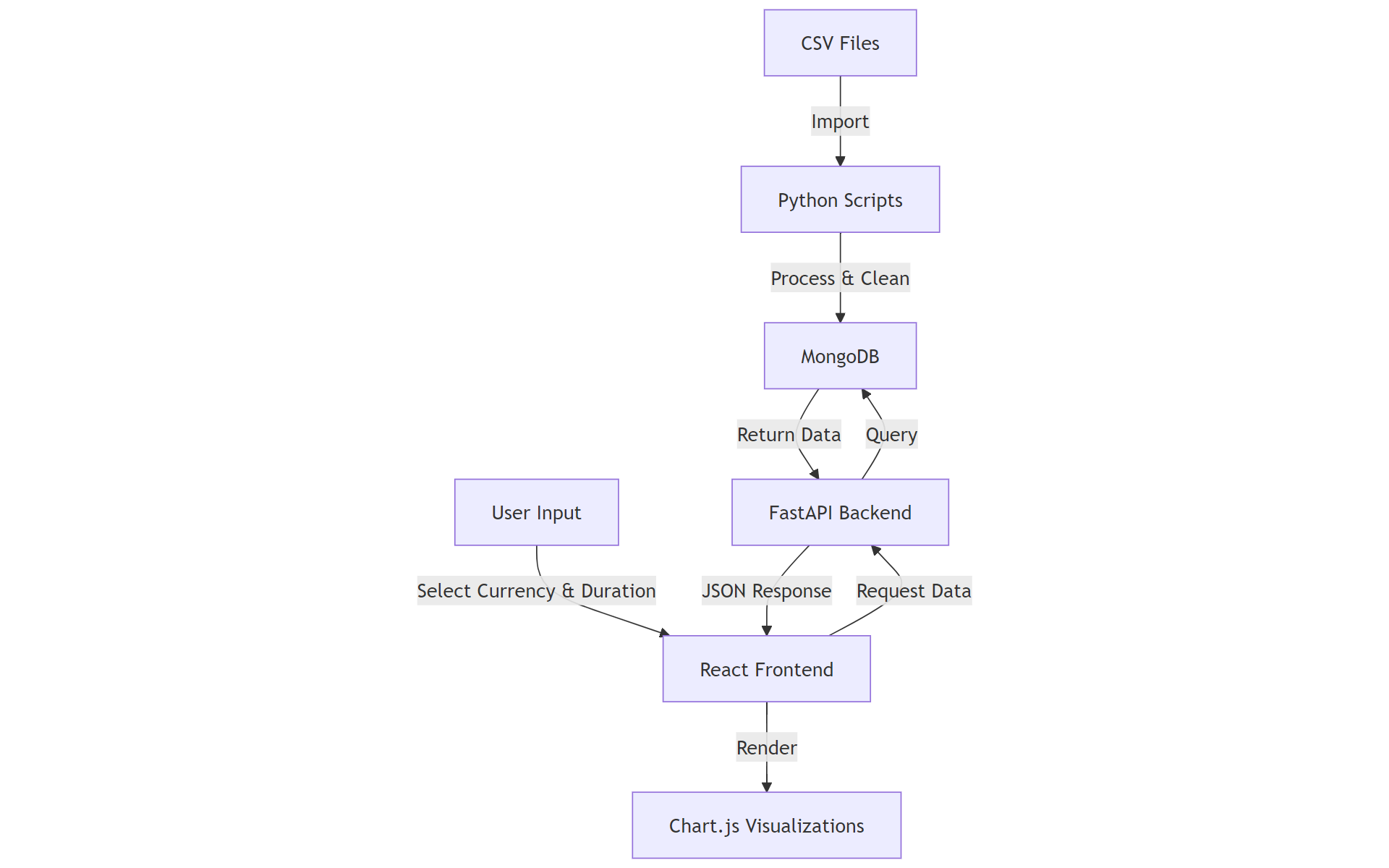
* npm install axios chart.js react-chartjs-2 papaparse

## Python Libraries

Install Python libraries:

* pip install pandas numpy flask matplotlib fastapi pymongo

# 4. System Architecture



# 5. Functional Tasks

1. Data Storage and Processing: The dataset (Exchange\_Rate\_Final-2013-2024.csv) contains exchange rate data from 2013 to 2024 and is used for analysis. It includes IMF and company data.

2. Data Merging & Preparation: Data from the IMF (2023-2024) and the company’s dataset (2013-2023) were merged, cleaned, and formatted for analysis.

3. Data Visualization: Data is displayed in Chart.js graphs, with interactive updates based on user inputs for currency pairs and time frames. Missing data is interpolated using polynomial methods and forward/backward fills.

4. Custom Currency Basket: Users can create custom baskets, assign weights, and see dynamic recalculations based on real-time exchange rates.

# 6. Backend and Frontend Integration

Frontend: Built using React.js with dynamic chart updates.

Backend: Uses FastAPI to fetch data from MongoDB based on user selections.

Data Format: Data is returned in JSON format and displayed on the frontend.

# 7. Handling Missing Data

Interpolation: Polynomial interpolation for minor gaps.

Forward/Backward Filling: Used for missing values at dataset ends.

Median Imputation: Applied for large sections of missing data.

Consistency: Data rounded to four decimal places for smooth visual trends.

# 8. Risk Indicator

A risk indicator (color-coded: green = low, yellow = medium, red = high) helps users gauge currency volatility, based on standard deviation and variance.

# 9. Additional Features

Implemented RNN Model: Predicts exchange rates with a mean square error of 0.0899.

Dataset Preprocessing: IMF data for 2023-2024 was cleaned and incorporated.

# 10. Conclusion

The dashboard enables users to analyze currency trends, create custom baskets, and assess volatility, improving decision-making on exchange rate fluctuations.