ITM SKILLS UNIVERSITY

School of Future Tech

Case Study Report

on

Forex Dashboard (Currency Converter)

by

Daksh Srivastava : 150096725087

Sumit Shingole : 150096725081

Anurag Kharke : 150096725100

Ayush Yadgiri : 150096725098

Index

INTRODUCTION

In today's globalized world, foreign exchange (Forex) plays a critical role in international trade, travel, investments, and online transactions. Users frequently need to convert one currency into another using up-to-date exchange rates. Manual conversion or outdated rates can lead to incorrect financial decisions.

This case study focuses on the design and implementation of a Forex Dashboard (Currency Converter) that allows users to convert currencies in real time using live exchange rate data fetched from an external API. The project demonstrates the practical use of core web development concepts such as Fetch API, asynchronous programming using async/await, and caching techniques to improve performance.

2. Problem Statement / Case Background (Abstract)

Background

Traditional currency conversion methods often rely on static exchange rates or manual calculations, which are inefficient and error-prone. With rapidly changing forex rates, there is a need for a dynamic system that fetches real-time data and presents accurate results instantly to users.

Abstract

This case study presents the development of a Forex Dashboard using JavaScript. The system fetches real-time exchange rates from the Open Exchange Rates API (open.er-api.com) using the Fetch API. Asynchronous operations are handled using async/await to ensure smooth user interaction without blocking the interface. To optimize performance and reduce unnecessary API calls, caching is implemented using browser localStorage. The application allows users to input an amount, select base and target currencies, and view the converted amount along with the last updated timestamp.

3. Case Study Design

The Forex Dashboard is designed as a simple, user-friendly web-based tool with the following components:

1. User Interface

- Input field to enter the amount
- Dropdown menus to select base currency and target currency

- A button to trigger currency conversion
- Display area for conversion result and last updated time

2. Data Fetching Layer

- Uses Fetch API to retrieve real-time exchange rates from the Open Exchange Rates API (open.er-api.com)

3. Caching Mechanism

- Uses browser localStorage to store fetched exchange rates
- Cached data is reused for a fixed duration to improve performance

4. Processing Logic

- Calculates converted amount based on fetched exchange rates
- Formats and displays output clearly

4. Methods & Technologies Applied

Key Concepts Used

1. Fetch API

- Used to send HTTP requests to a currency exchange API
- Retrieves latest exchange rates in JSON format

2. Async / Await

- Handles asynchronous API calls efficiently
- Improves code readability and avoids callback-based complexity

3. Caching

- Exchange rate data is stored in localStorage
- Cached data is reused for one hour to reduce API calls and latency

4. JavaScript DOM Manipulation

- Reads user input from form elements
- Dynamically updates the result and timestamp on the web page

Technology Stack

- Programming Language: JavaScript
- Frontend: HTML (structure), JavaScript (logic)
- Storage: Browser localStorage
- API: Open Exchange Rates API (open.er-api.com)
- Environment: Any modern web browser

5. Implementation Details and Snapshots

Implementation Overview

The project logic is implemented in JavaScript and includes the following modules:

1. Currency Initialization

- A predefined list of popular currencies is used to populate dropdown menus dynamically

2. Rate Fetching Function

- Fetches exchange rates using Fetch API
- Checks cached data before making a network request

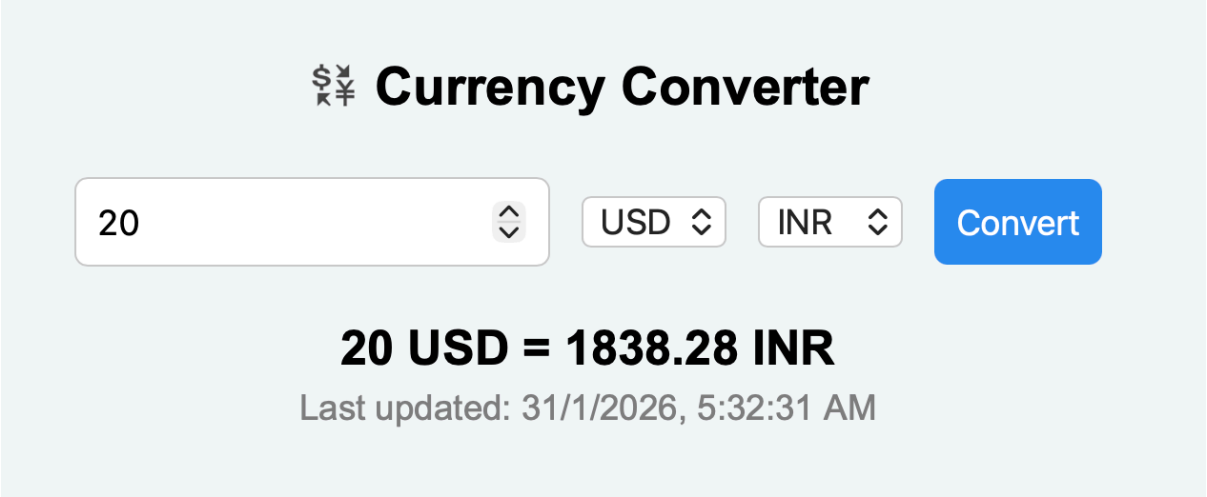
3. Conversion Logic

- Multiplies user-entered amount with the selected exchange rate
- Displays formatted conversion result

4. Error Handling

- Displays appropriate messages for invalid input or API failures

5. Snapshots :



6. Results and Conclusion

Results

- The Forex Dashboard successfully converts currencies using real-time exchange rates
- The use of caching significantly reduces repeated API calls
- Async/await ensures smooth and responsive user experience
- The system provides accurate conversion results along with update timestamps

Conclusion

This case study demonstrates the practical implementation of a real-world Forex Dashboard using JavaScript. By integrating Fetch API, async/await, and caching techniques, the project delivers an efficient and user-friendly currency conversion tool. The application highlights how core web development concepts can be applied to solve real-world financial problems and can be further extended with features like charts, historical data, and multi-language support.

7. References

- JavaScript Official Documentation

- MDN Web Docs – Fetch API
- MDN Web Docs – async/await
- Open Exchange Rates API Documentation (open.er-api.com)
- Web Development Tutorials on Currency Conversion Systems