|  |  |
| --- | --- |
| **SITAM**    **SATYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT**  **VIZIANAGARAM**    **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**            **BONAFIDE CERTIFICATE**    This is to certify that the project work entitled “**CURRENCY CONVERTOR** **PROJECT**” is a fulfilment of project work done by **S.Sridhar** (22**B61A0569) , S.Bashkar (22B61A0570) , S.Padmaja (22B61A0572) , P.Dillesh (22B61A0561) , S.Lakshman (22B6510566)** for the award of degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING, SATYA INSTITUTE OF TECHNOLOGY** during the academic year 2023-2024.    **PROJECT GUIDE HEAD OF THE DEPARTMENT**    **Dr.G. Venu Madhav Rao**  **Associate. Professor & HOD**  **Dept. of CSE** | |
| **TABLE OF CONTENTS**     |  |  |  | | --- | --- | --- | | **S.NO** | **TOPIC NAME** | **Page No** | | 1. | ACKNOWLEDGEMENT | 4 | | 2. | DECLARATION | 5 | | 3. | ABSTRACT | 6 | |  | CONTENT |  | | 4. | INTRODUCTION | 7 - 8 | | 5. | SYSTEM ANALYSIS | 9 - 12 | | 6. | SYSTEM REQUIRMENT  SPECIFICATIONS  3.1 HARDWARE REQUIREMENTS  3.2 SOFTWARE REQUIREMENTS | 13 - 18 | | 7. | CODE DESIGN | 19 - 21 | | 8. | METHODOLOGY | 22 - 24 | | 9. | CODE | 25 - 43 | | 10. | OUTPUT SCREENS | 44 - 46 | | 11. | I  CONCLUSON | 47 - 48 | | 12. | REFERENCES | 49 - 51 | | |

## ACKNOWLEDGEMENT

We extend our heartfelt gratitude to all those who have contributed to the development and success of our Currency Converter project. This endeavor has been a collaborative journey, and we are deeply appreciative of the support and guidance we have received throughout its creation.

First and foremost, we express our sincere thanks to the dedicated team members who invested their time, skills, and knowledge into conceptualizing, designing, and implementing this Currency Converter. Each member of our team played a pivotal role in various aspects of the project, from coding and testing to user experience design and project management.

We also wish to convey our appreciation to our mentors and advisors for their invaluable insights, direction, and encouragement. Their expertise and feedback have been instrumental in shaping the project and ensuring its adherence to the highest standards of functionality and usability.

Our gratitude extends to our peers and colleagues who generously provided constructive feedback, engaged in testing, and offered insightful suggestions to enhance the overall quality of the Currency Converter. Your collaboration has been indispensable to the project's success.

Additionally, we acknowledge the unwavering support of our friends and family, whose encouragement and understanding served as a constant source of motivation throughout the development process.

Last but certainly not least, we express our gratitude to the users of our Currency Converter. Your feedback and engagement have been invaluable in refining the application and making it a useful tool for currency conversion needs.

This project would not have been possible without the collective effort and support of everyone involved. Thank you for being part of the Currency Converter journey.

Sincerely,

DigitalDollar Developers

## DECLARATION

We, the undersigned, hereby declare that the Currency Converter project, herein referred to as "the project," is the result of our collective effort and collaboration.

This project was conceptualized, designed, and implemented with the intention of providing a useful tool for currency conversion purposes. The features, functionalities, and codebase of the project have been developed with utmost diligence and adherence to best practices in web development.

We affirm that all resources, including but not limited to code snippets, libraries, and media assets, used in the development of this project are either original creations or have been appropriately attributed to their respective sources.

Furthermore, we assert that any third-party services or APIs utilized within the project have been integrated and used in compliance with their respective terms of service and usage policies.

We acknowledge that while every effort has been made to ensure the accuracy and reliability of the Currency Converter application, we cannot guarantee its flawless performance under all circumstances. Users are encouraged to utilize the application responsibly and understand that any reliance on its output is at their own discretion.

We hereby disclaim any liability for damages or losses arising from the use of the Currency Converter application, including but not limited to inaccuracies in exchange rates, technical issues, or unintended outcomes.

By accessing and using the Currency Converter application, users agree to abide by the terms and conditions outlined herein.

Furthermore, we extend our gratitude to all individuals, organizations, and resources that contributed directly or indirectly to the completion of this project. Their support, guidance, and inspiration have been invaluable throughout our journey.

Sincerely,

S.Sridhar

## ABSTRACT

The Currency Converter project, developed by the innovative team known as CoinTech Creators, presents a sophisticated yet user-friendly solution to the ubiquitous need for currency conversion. In an era characterized by global trade and cross-border transactions, the ability to accurately convert currencies is indispensable.

This project endeavors to address this need by offering a seamlessly integrated web application that empowers users to perform currency conversions with ease and precision.

At its core, the Currency Converter boasts a meticulously crafted interface designed to facilitate intuitive user interaction. Through thoughtful design choices and meticulous attention to detail, the application ensures a smooth and seamless user experience.

Leveraging modern web technologies, including HTML5, CSS3, and JavaScript, the Currency Converter delivers a responsive and feature-rich platform accessible across a variety of devices and browsers.

Key features of the Currency Converter include real-time exchange rate updates sourced from reliable APIs, dynamic currency selection options, and user-friendly input mechanisms.

By harnessing the power of reputable exchange rate data sources, the application ensures accuracy and reliability in currency conversions, enabling users to make informed financial decisions.

Moreover, the Currency Converter project embodies a commitment to collaboration and innovation. Through the collective expertise and dedication of the CoinTech Creators team, the project has evolved into a sophisticated tool that meets the diverse needs of users in today's interconnected world.

With a focus on usability, functionality, and accessibility, the Currency Converter project sets a new standard for currency conversion tools, empowering users to navigate the complexities of global finance with confidence and ease.

## INTRODUCTION

In an increasingly interconnected global economy, the ability to quickly and accurately convert currencies is essential for individuals and businesses alike. Whether it's for international travel, e-commerce transactions, or financial planning, having access to a reliable currency conversion tool can streamline processes and facilitate decision-making. With this in mind, we present the Currency Converter project, a web-based application designed to provide users with a seamless and efficient means of converting currencies in real-time.

1. **Background:**

The idea for the Currency Converter project stemmed from the recognition of the growing need for a user-friendly and accessible tool to perform currency conversions. In today's digital age, where borders are becoming increasingly porous and international transactions are commonplace, having a reliable currency conversion mechanism is indispensable. The project aims to address this need by offering a simple yet powerful solution that enables users to convert currencies with ease.

1. Purpose:

The primary purpose of the Currency Converter project is to provide users with a convenient and efficient way to perform currency conversions. Whether users are planning a trip abroad, conducting international business transactions, or simply keeping track of foreign exchange rates, the Currency Converter application seeks to simplify the process and provide accurate results in real-time. By offering a user-friendly interface and leveraging real-time exchange rate data, the project aims to enhance user experience and streamline currency conversion processes.

1. Objectives:

The objectives of the Currency Converter project are as follows:

1. Develop a web-based application that allows users to input a specific

amount in one currency and convert it to another currency.

1. Provide users with access to up-to-date exchange rate data sourced from

reliable sources to ensure accuracy and reliability.

1. Design an intuitive and user-friendly interface that makes it easy for

users to navigate the application and perform currency conversions effortlessly.

1. Implement responsive design principles to ensure that the Currency

Converter application is accessible across a wide range of devices and screen sizes.

1. Incorporate error handling and validation mechanisms to enhance the

robustness and reliability of the application.

1. Explore opportunities for future enhancements and integrations, such as

adding support for additional currencies or incorporating advanced features like historical exchange rate data analysis.

1. Scope:

The scope of the Currency Converter project encompasses the development of a fully functional web-based application that enables users to perform currency conversions in real-time. The application will support conversion between a wide range of currencies and provide users with access to accurate and up-to-date exchange rate data. Additionally, the project will focus on creating an intuitive and user-friendly interface that caters to the needs of both novice and experienced users

## SYSTEM ANALYSIS

1. Purpose and Scope:

The Currency Converter project aims to provide users with a reliable, user-friendly web application for converting currencies accurately. The system caters to a wide range of users, including individuals, businesses, and travelers, who require currency conversion services for various purposes.

1. Functional Requirements:

• Currency Selection: Users can select both the source and target currencies

from dropdown menus populated with a comprehensive list of currency

codes and corresponding flags.

• Real-time Exchange Rate Updates: The system fetches real-time

exchange rate data from the ExchangeRate-API to ensure accuracy in

currency conversions.

• Input Mechanism: Users can input the amount of the source currency they

wish to convert.

• Currency Conversion: Upon selecting the currencies and entering the

amount, the system calculates and displays the converted amount in the

target currency.

• User Interface Interaction: The interface allows users to interact with

elements such as dropdown menus, input fields, buttons, and exchange

rate display areas seamlessly.

• Exchange Rate Display: The system presents the exchange rate for the

selected currencies, providing users with transparency regarding the

conversion process.

1. Non-Functional Requirements:

• Performance: The system should respond promptly to user actions,

fetching exchange rates and updating conversion results efficiently.

• Reliability: Currency conversion should be accurate and reliable, ensuring

that users can trust the system for their financial transactions.

• Usability: The user interface should be intuitive and easy to navigate,

catering to users with varying levels of technical expertise.

• Compatibility: The application should be compatible with a wide range of

devices and browsers to ensure accessibility for all users.

• Security: While the system does not involve sensitive user data, ensuring

Secure communication with the ExchangeRate-API and protecting user

privacy are essential considerations.

1. System Components:

• HTML Structure: Defines the layout and structure of the web page,

including elements such as dropdown menus, input fields, buttons, and

exchange rate display areas.

• CSS Styles: Enhances the visual presentation and user experience by

applying styles to HTML elements, ensuring consistency and aesthetic

appeal.

• JavaScript Functions: Implements the core logic of the Currency

Converter, including fetching exchange rates, calculating conversions,

updating the user interface, and handling user interactions.

• External APIs: Utilizes the ExchangeRate-API to retrieve real-time

exchange rate data, integrating external resources to enhance the system's

functionality.

1. User Interaction:

• Selecting Currencies: Users interact with dropdown menus to select the

source and target currencies for conversion.

• Entering Amount: Users input the amount of the source currency they

wish to convert into the designated input field.

• Fetching Exchange Rates: The system communicates with the

ExchangeRate-API to fetch the latest exchange rates for the selected

currencies.

• Displaying Conversion Results: Upon receiving exchange rate data, the

system calculates the converted amount and displays it to the user.

1. System Constraints:

• Dependency on External APIs: The system relies on the availability and

reliability of the ExchangeRate-API for fetching exchange rate data. Any

downtime or issues with the API may impact the system's functionality.

• Internet Connectivity: Users must have an active internet connection to

access the Currency Converter and fetch real-time exchange rate data

from the external API.

1. Future Enhancements:

• Localization: Implementing support for multiple languages and

currencies to cater to a more diverse user base.

• Historical Exchange Rates: Adding functionality to retrieve historical

exchange rate data, enabling users to track currency trends over time.

• User Preferences: Allowing users to save their preferred currencies or

customize the interface according to their preferences for a personalized

experience.

• Offline Mode: Implementing a caching mechanism to store exchange

rate data locally, enabling users to perform currency conversions even

without an internet connection.

SYSTEM REQUIREMENT SPECIFICATION

#### HARDWARE REQUIREMENTS :

The hardware requirements for the front-end of the Currency Converter project focus on ensuring compatibility and responsiveness across various client devices. The following hardware specifications are recommended:

1. Client Devices:

• Desktops, laptops, tablets, and smartphones are the primary client

devices for accessing the Currency Converter web application.

• These devices should have standard hardware configurations to support

the rendering of HTML, CSS, and JavaScript content smoothly.

• Adequate processing power is necessary to ensure responsive

performance, especially during dynamic interactions and animations.

• Sufficient memory is required to handle browser operations effectively

and cache website assets for faster loading times.

1. Display Resolution:

• The Currency Converter application should support a wide range of

display resolutions to accommodate various devices and screen sizes.

• Common aspect ratios such as 16:9, 16:10, and 4:3 should be supported

to ensure compatibility across different devices.

• Compatibility with high-definition (HD) and ultra-high-definition (UHD)

displays is essential to provide users with an optimal viewing experience.

• Responsive design principles should be implemented to ensure that the

user interface adapts seamlessly to different screen sizes and orientations.

1. Input Devices:

• Standard input devices such as keyboards, mice, touchpads, and

touchscreens should be supported for user interaction.

• The Currency Converter application should be compatible with various

input methods to accommodate different user preferences and

accessibility needs.

• Touchscreen devices should offer intuitive touch gestures for navigating

the user interface and inputting values.

1. Internet Connectivity:

• A reliable internet connection is crucial for accessing the Currency

Converter web application and fetching real-time exchange rate data.

• Users should have sufficient bandwidth to load web pages quickly and

retrieve data from external APIs efficiently.

• The application should be optimized for low-bandwidth scenarios to

ensure usability in environments with limited internet connectivity.

* 1. **SOFTWARE REQUIREMENTS:**

The successful development and deployment of the Currency Converter xcv project rely on a robust set of software requirements. These requirements encompass various tools, technologies, and platforms necessary for building a functional and efficient application. Below, we outline the key software requirements:

Development Environment:

• Code Editor: A modern and feature-rich code editor such as Visual Studio

Code, Sublime Text, Atom, or WebStorm is essential for writing and

managing HTML, CSS, and JavaScript code efficiently.

• Version Control System (VCS): Utilizing a VCS like Git is imperative for

managing source code, facilitating collaboration among developers, and

tracking changes throughout the de velopment process.

• Web Browser: Access to the latest versions of web browsers such as Google

Chrome, Mozilla Firefox, Safari, and Microsoft Edge is required for testing

and debugging the Currency Converter application across different

platforms.

Front-end Technologies:

• HTML (HyperText Markup Language): HTML serves as the backbone for

structuring the content of web pages, ensuring proper organization and

accessibility of information.

• CSS (Cascading Style Sheets): CSS is utilized for styling the presentation and

layout of HTML elements, providing a visually appealing and user-friendly

interface.

• JavaScript: JavaScript is essential for implementing interactivity and dynamic

behavior within the Currency Converter application, enabling features such as

real-time currency conversion and user input validation.

External Libraries and Frameworks:

• Font Awesome: Integration with Font Awesome provides access to a

comprehensive library of scalable vector icons, enhancing the visual appeal

and user experience of the application.

• ExchangeRate-API: Utilizing external APIs such as ExchangeRate-API allows

for fetching real-time exchange rate data, enabling accurate currency

conversion functionality within the application.

Web Hosting and Deployment:

• Web Server: Deployment of the Currency Converter application requires a

reliable web server such as Apache, Nginx, or Microsoft IIS to host and serve

the application to users over the internet.

• Domain Name: Acquisition of a registered domain name is necessary for

providing users with a branded and easily accessible online presence for

accessing the Currency Converter application.

• SSL Certificate: Securing the application with an SSL certificate is crucial for

enabling HTTPS protocol, encrypting data transmission, and ensuring the

privacy and security of user information during currency conversions.

Data Storage and Management:

• NoSQL Database: While optional, implementing a NoSQL database may be

beneficial for storing user preferences or historical exchange rate data,

facilitating efficient data retrieval and management within the application.

• File Storage: Provision for file storage may be required for storing images or

multimedia files used within the Currency Converter application, supporting

content management and delivery as needed.

Testing and Debugging:

• Developer Tools: Access to built-in browser developer tools is essential for

debugging and testing front-end code, ensuring code quality, and identifying

and resolving issues during development.

• Testing Frameworks: Integration with testing frameworks such as Jest, Mocha,

or Jasmine may be advantageous for implementing automated testing of

JavaScript code, validating the functionality and reliability of the Currency

Converter application.

Documentation and Collaboration:

• Documentation Tools: Utilization of documentation tools such as Markdown,

Microsoft Word, or Google Docs is essential for documenting project

requirements, design specifications, and user guides, ensuring clear

communication and knowledge sharing among team members.

• Communication Platforms: Adoption of collaboration platforms such as Slack,

Microsoft Teams, or Discord facilitates seamless communication,

coordination, and shari ng of updates and progress among project stakeholders

and team members.

#### 

#### CODE DESIGN

The Currency Converter application is designed to provide a user-friendly interface for converting currencies efficiently. It utilizes HTML, CSS, and JavaScript for the client-side functionality.

* 1. CLIENT-SIDE CODE DESIGN:

The client-side code is responsible for rendering the user interface, handling user interactions, and performing currency conversions.

HTML (index.html):

• Structured layout using HTML5 for semantic markup.

• Input fields for entering the amount and selecting source and target

currencies.

• Button for initiating currency conversion.

CSS (stylesheet.css):

• Styling the HTML elements for a visually appealing interface.

• Responsive design using media queries to ensure compatibility across

devices.

• Consistent styling for a seamless user experience.

JavaScript (javascript.js):

• Client-side validation to ensure valid input for amount and currency

selection.

• Asynchronous requests (AJAX) for fetching real-time exchange rates from

the server.

• Dynamic content updates based on user interactions.

• Event listeners for handling button clicks and form submissions.

* 1. SERVER-SIDE CODE DESIGN:

As the Currency Converter project primarily focuses on the front-end, there is no significant server-side code. However, the application may integrate server-side functionality for fetching real-time exchange rates from external APIs.

* 1. CODE ORGANIZATION:

File Structure:

• Organizing files into logical directories (e.g., CSS, JS) for better

maintainability.

• Keeping related files together (e.g., HTML, CSS, JS files in the same

directory).

Modularization:

• Breaking down JavaScript code into reusable functions for currency

conversion and data fetching.

• Ensuring modularity for easier debugging and future enhancements.

* 1. SECURITY CONSIDERATIONS:

As the Currency Converter project deals with client-side functionality and does not involve user authentication or sensitive data storage, security considerations are minimal. However, best practices such as input validation and using secure APIs for fetching exchange rates should be followed to mitigate potential risks.

Input Validation:

• Validating user input for amount to ensure it is a valid numerical value.

• Sanitizing user input for currency selection to prevent potential

vulnerabilities.

Secure APIs:

• Utilizing secure APIs for fetching exchange rates to ensure data integrity

and confidentiality.

• Verifying the authenticity and reliability of the exchange rate provider.

## 

## 

## METHODOLOGY

Iterative Development Model

The Currency Converter project adopts an Iterative Development Model for its implementation. This approach allows for continuous improvement, flexibility, and adaptation throughout the development lifecycle.

1. Team Collaboration:

• The development team collaborates closely, with each member assigned to

specific aspects of the project such as front-end development, back-end

integration, and quality assurance.

• Individual responsibilities are aligned with the incremental goals set for

each iteration of development.

1. Continuous Updates:

• The project undergoes continuous updates and refinements at the

conclusion of each iteration.

• Feedback from users and stakeholders is collected and incorporated

into subsequent iterations to address any issues or suggestions for

improvement.

1. Iterative Development:

• Development proceeds in iterative cycles, with each cycle focusing on

a specific set of features or enhancements.

• New functionalities and improvements are introduced incrementally,

building upon the achievements of previous iterations.

1. Flexibility to Accommodate Changes:

• The Iterative Model provides flexibility to accommodate changes to

requirements or design preferences.

• Modifications can be implemented easily within each iteration without

disrupting the overall progress of the project.

1. Modular Development Approach:

• The project follows a modular development approach, treating different

components such as UI elements, currency conversion logic, and API

integrations as independent modules.

• Modularity facilitates easier management, testing, and integration of

individual components, enhancing overall project efficiency.

1. Testing and Quality Assurance:

• Comprehensive testing and quality assurance procedures are integrated

into each iteration to ensure the stability, reliability, and functionality

of the Currency Converter application.

• Automated testing frameworks and manual testing techniques are

employed to validate the performance and accuracy of the system.

1. Continuous Integration and Deployment:

• Continuous integration and deployment pipelines are established to

automate the process of code integration, testing, and deployment.

• This ensures that new features and updates are seamlessly integrated into

the application and made available to users in a timely manner.

1. User-Centric Design and Feedback Loop:

• The development process is guided by a user-centric design approach,

with a focus on meeting the needs and preferences of the end-users.

• User feedback is actively solicited and incorporated into the development

process to ensure that the Currency Converter application delivers an

intuitive and satisfying user experience.

1. Documentation and Knowledge Management:

• Comprehensive documentation is maintained throughout the documents,

development lifecycle, including requirements specifications, design

and user manuals.

• Knowledge management practices are implemented to ensure that critical

project information is captured, shared, and preserved for future

reference.

1. Agile Principles and Continuous Improvement:

• The development team embraces agile principles such as adaptability,

collaboration, and transparency to drive continuous improvement and

innovation.

• Regular retrospectives are conducted to reflect on past experiences,

identify areas for improvement, and implement corrective actions in

subsequent iterations.

CODES

### HTML CODE

**User Interface (UI.html):**

<!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="utf-8">

<title>Currency Converter App in JavaScript | CodingNepal</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css" />

<link rel="stylesheet" href="stylesheet.css">

</head>

<body>

<video autoplay muted loop id="background-video">

<source src="video.mp4" type="video/mp4">

</video>

<div class="wrapper">

<div class="heading-box">

<header>Currency Converter</header>

<form action="#">

<div class="amount">

<p>Enter Amount</p>

<input type="text" value="1">

</div>

<div class="drop-list">

<div class="from">

<p>From</p>

<div class="select-box">

<img src="https://flagcdn.com/48x36/us.png" alt="flag">

<select></select>

</div>

</div>

<div class="icon"><i class="fas fa-exchange-alt"></i></div>

<div class="to">

<p>To</p>

<div class="select-box">

<img src="https://flagcdn.com/48x36/np.png" alt="flag">

<select></select>

</div>

</div>

</div>

<div class="exchange-rate">Getting exchange rate...</div>

<button>Get Exchange Rate</button>

</form>

</div>

<video autoplay muted loop id="background-video">

<source src="video.mp4" type="video/mp4">

</video>

<script src="javascript.js"></script>

</body>

</html>

CSS CODE

Stylesheet (Stylesheet.css):

/\* Import Google Font - Poppins \*/

@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=swap');

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

body {

display: flex;

align-items: center;

justify-content: center;

min-height: 100vh;

padding: 0 10px;

background: #675AFE;

}

#background-video {

position: fixed;

top: 0;

left: 0;

min-width: 100%;

min-height: 100%;

z-index: -1;

}

::selection {

color: #fff;

background: #675AFE;

}

.wrapper {

width: 370px;

padding: 30px;

border-radius: 7px;

background:aqua;

box-shadow: 7px 7px 20px rgba(0, 0, 0, 0.05);

}

.wrapper header {

margin: 20px 0 30px;

font-size: 28px;

font-weight: 500;

text-align: center;

}

.wrapper form {

margin: 40px 0 20px 0;

}

form :where(input, select, button) {

width: 100%;

outline: none;

border-radius: 5px;

border: none;

}

form p {

font-size: 18px;

margin-bottom: 5px;

}

form input {

height: 50px;

font-size: 17px;

padding: 0 15px;

border: 1px solid #999;

}

form input:focus {

padding: 0 14px;

border: 2px solid #675AFE;

}

form .drop-list {

display: flex;

margin-top: 20px;

align-items: center;

justify-content: space-between;

}

.drop-list .select-box {

display: flex;

width: 115px;

height: 45px;

align-items: center;

border-radius: 5px;

justify-content: center;

border: 1px solid #999;

}

.select-box img {

max-width: 21px;

}

.select-box select {

width: auto;

font-size: 16px;

background: none;

margin: 0 -5px 0 5px;

}

.select-box select::-webkit-scrollbar {

width: 8px;

}

.select-box select::-webkit-scrollbar-track {

background: #fff;

}

.select-box select::-webkit-scrollbar-thumb {

background: #888;

border-radius: 8px;

border-right: 2px solid #ffffff;

}

.drop-list .icon {

cursor: pointer;

margin-top: 30px;

font-size: 22px;

}

form .exchange-rate {

font-size: 17px;

margin: 20px 0 30px;

}

form button {

height: 52px;

color: #fff;

font-size: 17px;

cursor: pointer;

background: #675AFE;

transition: 0.3s ease;

}

form button:hover {

background: #4534fe;

}

### 

### JAVA SCRIPT CODE

JavaScript (javascript.js):

let country\_list = {

"AED" : "AE",

"AFN" : "AF",

"XCD" : "AG",

"ALL" : "AL",

"AMD" : "AM",

"ANG" : "AN",

"AOA" : "AO",

"AQD" : "AQ",

"ARS" : "AR",

"AUD" : "AU",

"AZN" : "AZ",

"BAM" : "BA",

"BBD" : "BB",

"BDT" : "BD",

"XOF" : "BE",

"BGN" : "BG",

"BHD" : "BH",

"BIF" : "BI",

"BMD" : "BM",

"BND" : "BN",

"BOB" : "BO",

"BRL" : "BR",

"BSD" : "BS",

"NOK" : "BV",

"BWP" : "BW",

"BYR" : "BY",

"BZD" : "BZ",

"CAD" : "CA",

"CDF" : "CD",

"XAF" : "CF",

"CHF" : "CH",

"CLP" : "CL",

"CNY" : "CN",

"COP" : "CO",

"CRC" : "CR",

"CUP" : "CU",

"CVE" : "CV",

"CYP" : "CY",

"CZK" : "CZ",

"DJF" : "DJ",

"DKK" : "DK",

"DOP" : "DO",

"DZD" : "DZ",

"ECS" : "EC",

"EEK" : "EE",

"EGP" : "EG",

"ETB" : "ET",

"EUR" : "FR",

"FJD" : "FJ",

"FKP" : "FK",

"GBP" : "GB",

"GEL" : "GE",

"GGP" : "GG",

"GHS" : "GH",

"GIP" : "GI",

"GMD" : "GM",

"GNF" : "GN",

"GTQ" : "GT",

"GYD" : "GY",

"HKD" : "HK",

"HNL" : "HN",

"HRK" : "HR",

"HTG" : "HT",

"HUF" : "HU",

"IDR" : "ID",

"ILS" : "IL",

"INR" : "IN",

"IQD" : "IQ",

"IRR" : "IR",

"ISK" : "IS",

"JMD" : "JM",

"JOD" : "JO",

"JPY" : "JP",

"KES" : "KE",

"KGS" : "KG",

"KHR" : "KH",

"KMF" : "KM",

"KPW" : "KP",

"KRW" : "KR",

"KWD" : "KW",

"KYD" : "KY",

"KZT" : "KZ",

"LAK" : "LA",

"LBP" : "LB",

"LKR" : "LK",

"LRD" : "LR",

"LSL" : "LS",

"LTL" : "LT",

"LVL" : "LV",

"LYD" : "LY",

"MAD" : "MA",

"MDL" : "MD",

"MGA" : "MG",

"MKD" : "MK",

"MMK" : "MM",

"MNT" : "MN",

"MOP" : "MO",

"MRO" : "MR",

"MTL" : "MT",

"MUR" : "MU",

"MVR" : "MV",

"MWK" : "MW",

"MXN" : "MX",

"MYR" : "MY",

"MZN" : "MZ",

"NAD" : "NA",

"XPF" : "NC",

"NGN" : "NG",

"NIO" : "NI",

"NPR" : "NP",

"NZD" : "NZ",

"OMR" : "OM",

"PAB" : "PA",

"PEN" : "PE",

"PGK" : "PG",

"PHP" : "PH",

"PKR" : "PK",

"PLN" : "PL",

"PYG" : "PY",

"QAR" : "QA",

"RON" : "RO",

"RSD" : "RS",

"RUB" : "RU",

"RWF" : "RW",

"SAR" : "SA",

"SBD" : "SB",

"SCR" : "SC",

"SDG" : "SD",

"SEK" : "SE",

"SGD" : "SG",

"SKK" : "SK",

"SLL" : "SL",

"SOS" : "SO",

"SRD" : "SR",

"STD" : "ST",

"SVC" : "SV",

"SYP" : "SY",

"SZL" : "SZ",

"THB" : "TH",

"TJS" : "TJ",

"TMT" : "TM",

"TND" : "TN",

"TOP" : "TO",

"TRY" : "TR",

"TTD" : "TT",

"TWD" : "TW",

"TZS" : "TZ",

"UAH" : "UA",

"UGX" : "UG",

"USD" : "US",

"UYU" : "UY",

"UZS" : "UZ",

"VEF" : "VE",

"VND" : "VN",

"VUV" : "VU",

"YER" : "YE",

"ZAR" : "ZA",

"ZMK" : "ZM",

"ZWD":"ZW"

}

const dropList = document.querySelectorAll("form select"),

fromCurrency = document.querySelector(".from select"),

toCurrency = document.querySelector(".to select"),

getButton = document.querySelector("form button");

for (let i = 0; i < dropList.length; i++) {

for (let currency\_code in country\_list) {

let selected =

i === 0

? currency\_code === "USD"

? "selected"

: ""

: currency\_code === "NPR"

? "selected"

: "";

let optionTag = `<option value="${currency\_code}" ${selected}>${currency\_code}</option>`;

dropList[i].insertAdjacentHTML("beforeend", optionTag);

}

dropList[i].addEventListener("change", (e) => {

loadFlag(e.target);

});

}

function loadFlag(element) {

for (let code in country\_list) {

if (code == element.value) {

let imgTag = element.parentElement.querySelector("img");

imgTag.src = `https://flagcdn.com/48x36/${country\_list[code].toLowerCase()}.png`;

}

}

}

window.addEventListener("load", () => {

getExchangeRate();

});

getButton.addEventListener("click", (e) => {

e.preventDefault();

getExchangeRate();

});

const exchangeIcon = document.querySelector("form .icon");

exchangeIcon.addEventListener("click", () => {

let tempCode = fromCurrency.value;

fromCurrency.value = toCurrency.value;

toCurrency.value = tempCode;

loadFlag(fromCurrency);

loadFlag(toCurrency);

getExchangeRate();

});

function getExchangeRate() {

const amount = document.querySelector("form input");

const exchangeRateTxt = document.querySelector("form .exchange-rate");

let amountVal = amount.value;

if (amountVal === "" || amountVal === "0") {

amount.value = "1";

amountVal = 1;

}

exchangeRateTxt.innerText = "Getting exchange rate...";

let url = `https://v6.exchangerate-api.com/v6/1016372acd2c7b98174f6aca/latest/${fromCurrency.value}`;

fetch(url)

.then((response) => response.json())

.then((result) => {

let exchangeRate = result.conversion\_rates[toCurrency.value];

let totalExRate = (amountVal \* exchangeRate).toFixed(2);

exchangeRateTxt.innerText = `${amountVal} ${fromCurrency.value} = ${totalExRate} ${toCurrency.value}`;

})

.catch(() => {

exchangeRateTxt.innerText = "Something went wrong";

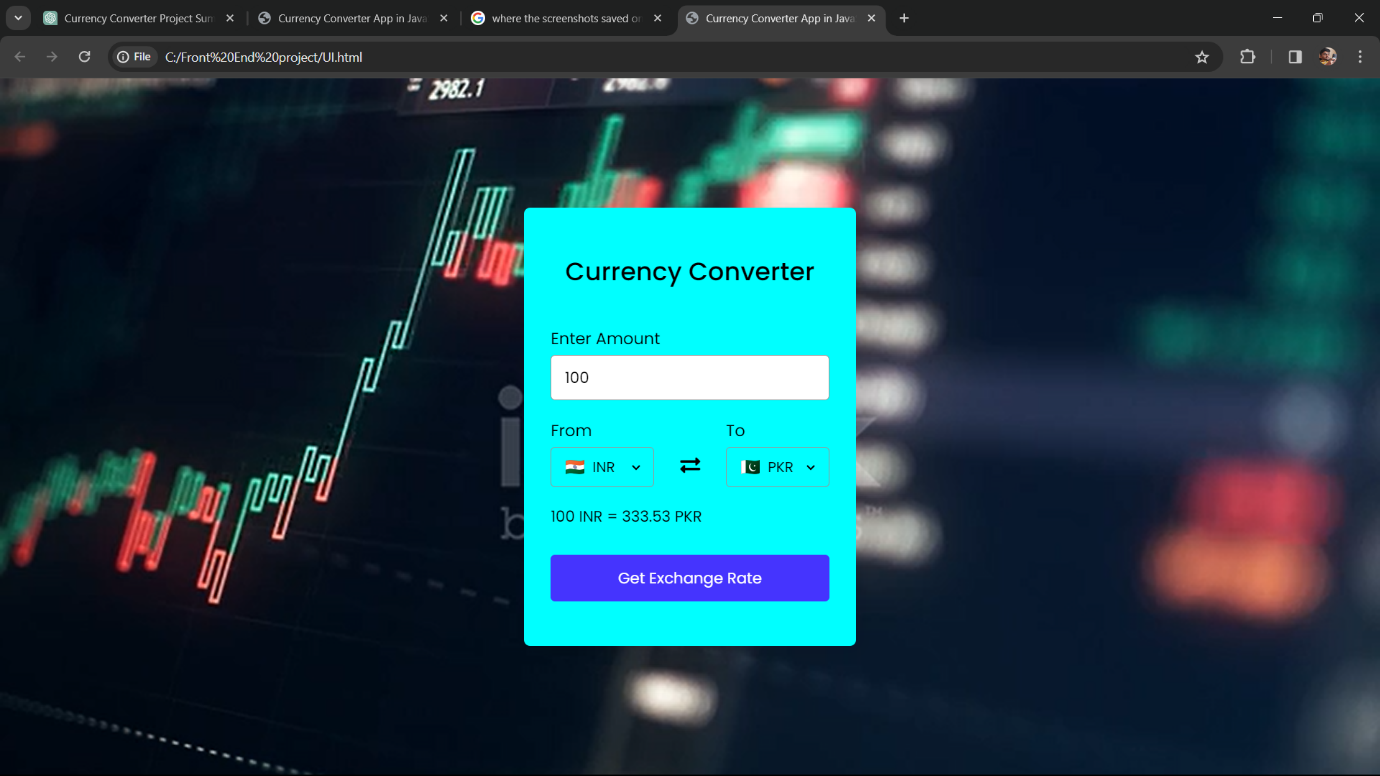
});

}

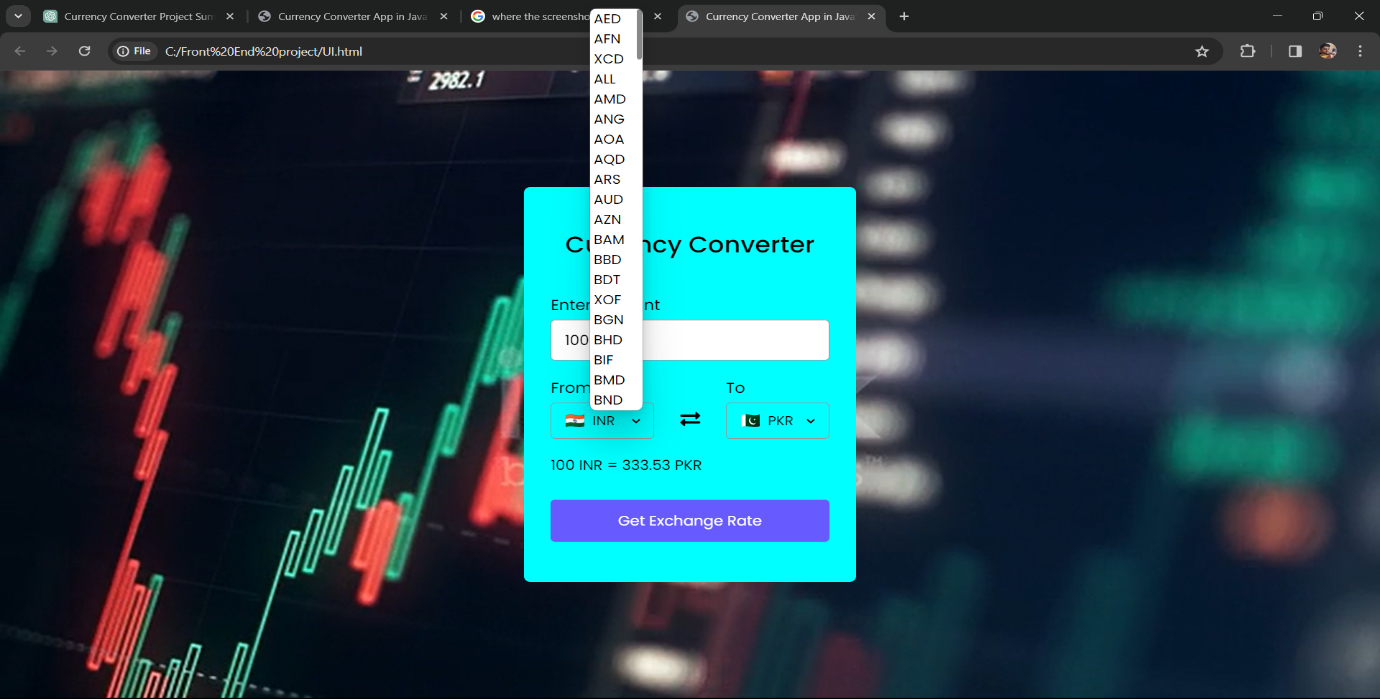
##### 

##### OUTPUT SCREENS

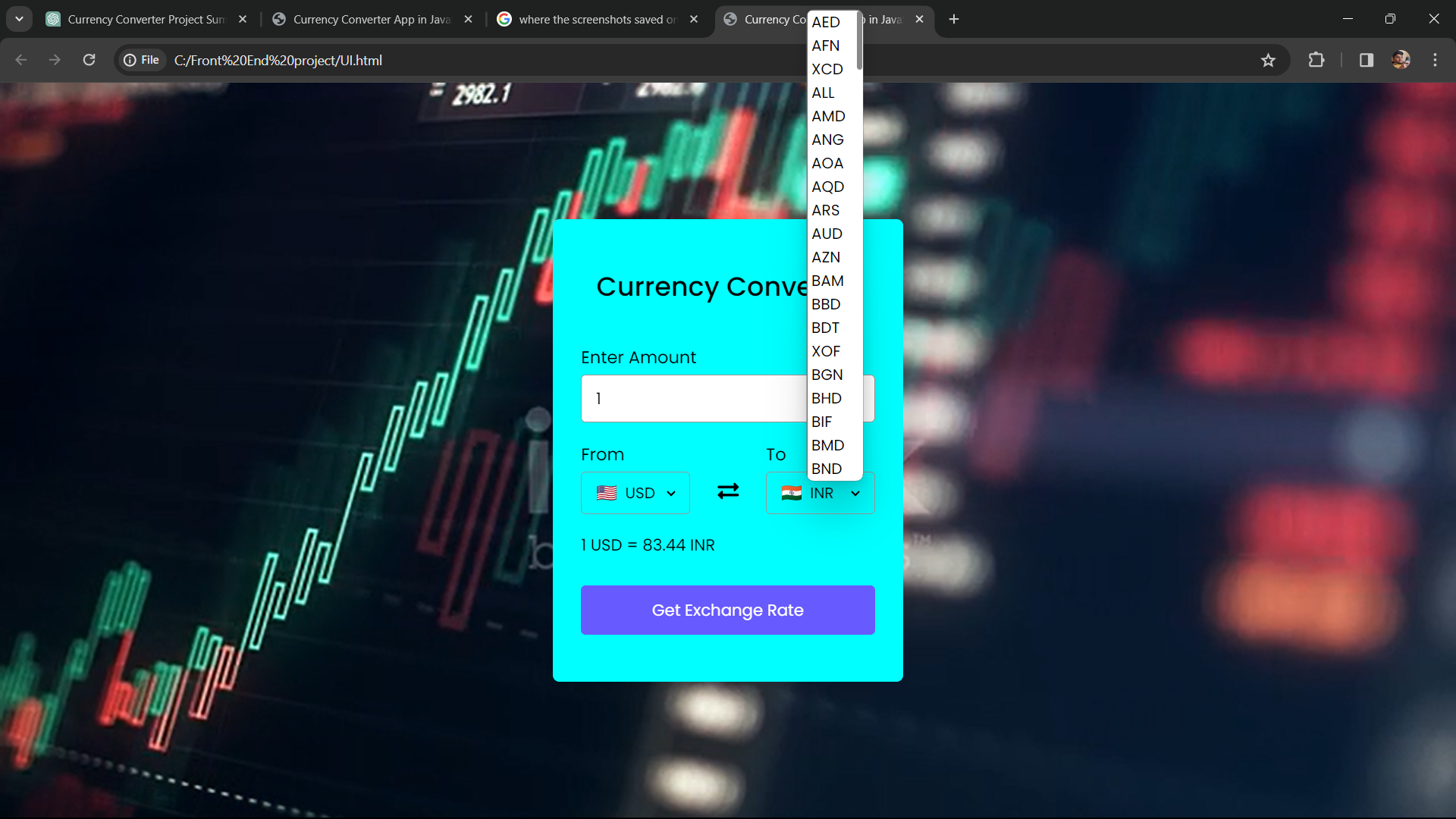
**User Interface :**

****

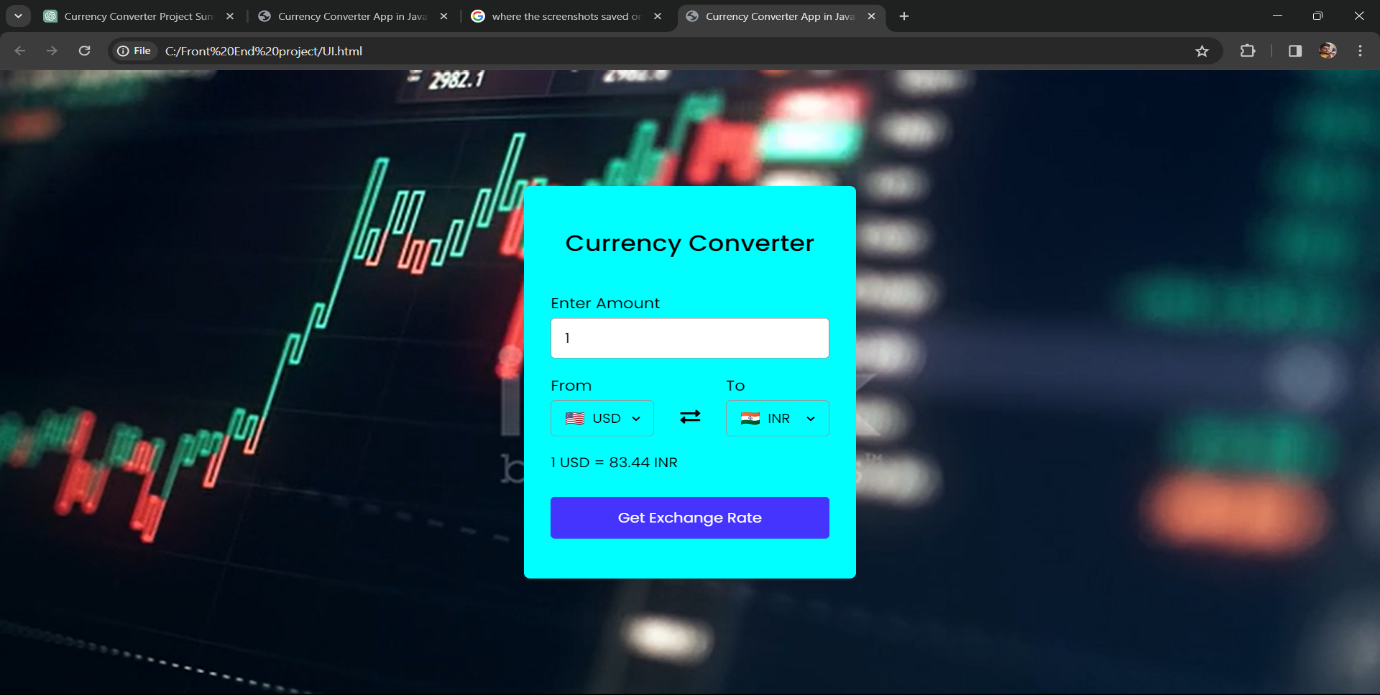
**From Currencies List :**

****

**To Currencies List :**

****

1 USD TO INR **:**



#### 100 INR TO PKR :



CONCLUSION

The development journey of the Currency Converter project has been a testament to our team's dedication, ingenuity, and collaborative spirit. Through meticulous planning, coding, and testing, we have successfully crafted a sophisticated currency conversion application that meets the needs of users worldwide. As we conclude this phase of the project, we reflect on the key highlights, challenges, and future prospects.

1.Successful Implementation:

The meticulous planning, coding, and testing efforts have culminated in the successful launch of the Currency Converter application. Our unwavering commitment to quality assurance ensured a high-quality, reliable, and user-friendly platform for currency conversion.

1. User-Centric Approach:

Our relentless focus on a user-centric approach has resulted in an intuitive and seamless experience for users seeking to convert currencies. By prioritizing user needs and preferences, we have created a platform that caters to both novice and experienced users alike.

1. Technology Stack:

Adopting a robust technology stack has empowered us to create a

scalable, efficient, and future-proof system. The continuous integration of

cutting-edge technologies ensures that the Currency Converter application

remains adaptable to emerging trends and advancements in the field of web

development.

1. Challenges and Solutions:

Throughout the development process, we encountered various technical challenges and evolving project requirements. However, our team's adaptability, resilience, and problem-solving skills enabled us to overcome these hurdles effectively. Each challenge presented an opportunity for growth and innovation, ultimately strengthening the project's foundations.

1. Looking Ahead:

As we conclude this phase of the Currency Converter project, our commitment extends to future updates, improvements, and enhancements. We remain dedicated to delivering ongoing value to our users by incorporating their feedback, embracing technological advancements, and refining the application to meet evolving needs and expectations.

1. Gratitude:

We extend our sincere thanks to every team member, stakeholder, and contributor who played a vital role in bringing the Currency Converter project to fruition. Your dedication, expertise, and collaborative efforts have been instrumental in our success.

1. Continuous Improvement:

We welcome feedback from our users, stakeholders, and the wider community for continuous improvement. By listening to our users' needs and incorporating their suggestions, we strive to make the Currency Converter application even more efficient, intuitive, and valuable.

Thank You:

Finally, we express our heartfelt gratitude to everyone who has been part of this journey. Your support, encouragement, and participation have been invaluable, and we are excited about the positive impact that the Currency Converter application will have on facilitating currency conversions worldwide.

Thank you for being part of this journey...

S.sridhar

DigitalDollar Developers

**REFERENCES**

Technology Stack:

1.HTML, CSS, JavaScript:

The foundation of the Currency Converter project's front-end development, enabling the creation of interactive user interfaces and dynamic content presentation.

2. PHP:

Utilized for server-side scripting and backend logic implementation, allowing for data processing, currency conversion calculations, and interaction with the database.

3.MySQL:

The chosen relational database management system responsible for storing and managing currency exchange rate data efficiently.

Development Tools:

1. Visual Studio Code:

The primary code editor employed for writing, editing, and debugging HTML, CSS, JavaScript, and PHP code, providing a seamless development experience.

1. Git and GitHub:

Version control system and collaborative platform utilized for managing project versions, tracking changes, and facilitating team collaboration.

1. XAMPP:

A local server environment used for PHP and MySQL development, enabling developers to test and debug server-side functionalities locally before deployment.

External Sources and Libraries:

1.Font Awesome:

A popular external library providing a vast collection of scalable vector icons, enhancing the visual appeal and usability of the Currency Converter project's user interface.

2.ExchangeRate-API:

An external API integrated into the project for fetching real-time exchange rate data, ensuring accuracy and reliability in currency conversion calculations.

Educational Resources:

1. MDN Web Docs:

A comprehensive documentation resource for web development technologies, offering in-depth explanations, tutorials, and references for HTML, CSS, JavaScript, and related topics.

2. W3Schools:

An online educational platform providing tutorials, examples, and references for various web development technologies, assisting developers in learning, practicing, and troubleshooting.

3.Stack Overflow:

A community-driven question-and-answer platform where developers seek solutions to coding challenges, troubleshoot issues, and share knowledge and expertise with fellow developers.

##### Project Inspirations:

##### 1.XE.com Currency Converter:

##### An influential online currency conversion tool known for its accuracy, reliability, and user-friendly interface, serving as a reference for the Currency Converter project's functionality and design.

##### 2. OANDA Currency Converter:

##### Another prominent currency conversion platform offering comprehensive features and real-time exchange rate data, inspiring aspects of the Currency Converter project's implementation and user experience.

##### These references provided invaluable guidance, inspiration, and resources throughout the development lifecycle of the Currency Converter project, contributing to its success and effectiveness in meeting user needs and expectations