|  |  |  |
| --- | --- | --- |
| zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz  **Tech Saksham**  Final Project Report |  |  |

**“CURRENCY CONVERTER”**

**“RAYALASEEMA UNIVERSITY COLLEGE OF ENGINEERING”**

|  |  |
| --- | --- |
| **ROLL NO** | **NAME** |
| 20RU1A0551 | TARIGOPULA LAVANYA |
| 20RU1A0546 | SANDRA POOJITHA |
| 20RU1A0507 | B0JJA PRATHYUSHA |
| 20RU1A0517 | GULAM MOHIDDIN KHAJAVALI |
|  |  |

|  |  |
| --- | --- |
|  |  |
|  | Hrishikesh Mahure  Edunet Trainer |

**ABSTRACT**

We all know that every country use different currency system, so when person is travel from one place to another or when person transfer the money from one country to another country then everyone needs to convert their currency, by keeping this thought on mind, I develop the currency converter system in html, css, and languages. It is very simple to calculate like application. This application is so convenivent that’s why anybody can use it. Most probably this application is used in the share market. Business strategy and many more. The currency converter has a simple interface in which user can enter the amount and choose the changes by dropdown list.

Different countries use different currency, and there is daily variation in these currencies relative to one another. Those who transfer money from one country to another (one currency to another) must be updated with the latest currency exchange rates in the market.

Currency converter mini project is built keeping this thing in mind. It is simply a calculator-like app developed using html, css web features. In this application, there is regular update about currency of every country by which it displays present currency market value and conversion rate.

Such web can be used by any user, but it is mainly useful for business, shares, and finance related areas where money transfer and currency exchange takes place on a daily basis.

In this currency converter are provided with an option to select the type of conversion, i.e. from “this” currency to “that” currency. This simple feature allows users to enter amount to be converted (say currency in Dollars), and display the converted amount (say currency in Euro)

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Table of Contents** | **Page No.** |
| 1 | Chapter 1: Introduction |  |
| 2 | Chapter 2: Services and Tools Required |  |
| 3 | Chapter 3: Project Architecture |  |
| 4 | Chapter 4: Architecture Blocks Detail Working |  |
| 5 | Conclusion |  |
| 6 | References |  |
| 7 | Code |  |

**CHAPTER 1**

**INTRODUCTION**

In our increasingly interconnected world, the need for currency conversion is a fundamental aspect of international trade, travel, and finance. Whether you're a business dealing with international clients, a traveler exploring foreign lands, or an investor diversifying your portfolio, the ability to quickly and accurately convert one currency to another is invaluable. To meet this demand and simplify the process, we have developed a Currency Converter Project

* 1. **Overview**

The Currency Converter Project is a versatile and user-friendly tool designed to provide real-time currency exchange rates and facilitate seamless conversions between various currencies. This project aims to address the challenges associated with currency exchange, such as fluctuating rates, complex calculations, and the need for up-to-date information.

**1.2 Feature**

**1. \*Real-Time Exchange Rates: \*** Our Currency Converter Project relies on real-time data sources, ensuring that users always have access to the latest exchange rates. This feature is crucial for making informed financial decisions or planning international trips.

**2. \*Multiple Currencies: \*** The project supports a wide range of currencies from around the world, making it a valuable resource for individuals and businesses engaged in global transactions.

**3. \*User-Friendly Interface:** \* We've designed the user interface to be intuitive and easy to navigate, ensuring that users of all levels of technical proficiency can use the converter with ease.

**4. \*Conversion History: \*** Users can track their conversion history, helping them keep a record of past transactions or financial activities. This feature is especially useful for budgeting and financial planning.

**5. \*Customization:** \* The project allows users to customize their experience by selecting their preferred base currency and setting up personalized watchlists of frequently used currencies.

**1.3 Advantages**

**1. \*Accurate Exchange Rates: \*** Currency converters provide real-time or up-to-date exchange rates, ensuring that users have access to accurate and reliable information. This accuracy is crucial for making informed financial decisions.

**2. \*Saves Time: \*** Currency converters automate the process of converting one currency into another, eliminating the need for manual calculations. This saves users time and reduces the risk of errors in currency conversion.

**3. \*Global Accessibility: \*** Currency converters are typically available online or as mobile apps, making them accessible from anywhere with an internet connection. This global accessibility is especially valuable for travelers and international businesses.

**4. \*Currency Diversification: \*** Investors can use currency converters to assess the value of assets and investments in different currencies. This helps with portfolio diversification and risk management.

**5. \*Exchange Rate Trends:** \* Some currency converters provide historical exchange rate data and trends, helping users analyze how exchange rates have fluctuated over time. This information can be valuable for decision-making.

**6. \*User-Friendly Interfaces**: \* Many currency converter apps and websites are designed with user-friendly interfaces, making them accessible to individuals with varying levels of technical expertise.

**7. \*Customization:** \* Users can often customize currency converters by selecting their preferred base currency and setting up watchlists of frequently used currencies. This personalization enhances the user experience.

**8. \*Conversion History:** \* Currency converters often include a feature that allows users to save their conversion history. This is useful for keeping records of financial transactions and reviewing past conversions.

**1.4 Scope**

The scope of a currency converter project can vary depending on its objectives and target audience. Here are some key aspects to consider when defining the scope of a currency converter project:

**1. \*Supported Currencies:** \* Determine which currencies the currency converter will support. You can choose to include major world currencies, less common currencies, or even cryptocurrencies. Consider whether the project will include historical exchange rate data for analysis.

**2. \*Real-Time Data: \***Decide if the currency converter will provide real-time exchange rate data or rely on periodic updates. Select a reliable data source or API to fetch the exchange rates.

**3. \*User Interface: \***Define the user interface (UI) design, including the layout, color scheme, and user interactions. Ensure that the UI is user-friendly and accessible to a wide range of users, including those with disabilities.

**4. \*Conversion Features:** \*Specify the types of conversions the currency converter will support, such as single currency conversions, multiple currency conversions, or cross-currency conversions. Consider adding features like currency conversion history, saved conversions, and favorite currencies.

**5. \*Customization Options:** \*Decide if users can customize their experience by selecting a preferred base currency. Allow users to create watchlists of frequently used currencies for quick access.

**6. \*Platform and Devices:** \* Determine whether the project will be a web application, a mobile app, or both. Ensure that the currency converter is responsive and compatible with various devices and screen sizes.

**7. \*Security and Privacy:** \* Implement security measures to protect user data an transactions. Address privacy concerns and compliance with data protection regulations if applicable.

**8. \*Data Accuracy:** \*Ensure that the project uses reliable data sources to provide accurate exchange rates.

**1.5 Future Work**

The future work for a currency converter project can involve several enhancements and expansions to improve its functionality, user experience, and relevance in an ever-changing financial landscape. Here are some potential areas of future work for a currency converter project

**CHAPTER 2**

**SERVICES AND TOOLS REQUIRED**

**2.1 Services Used**

When developing a currency converter HTML/CSS project, you primarily focus on the client-side development of the user interface. Here are some common services, libraries, and resources you can use in an HTML/CSS currency converter project:

**1. \*Currency Exchange Rate API: \***Utilize a currency exchange rate API, such as Open Exchange Rates, Fixer.io, Currency Layer, or free APIs provided by central banks, to fetch real-time or periodic exchange rate data.

**2. \*JavaScript:** \*JavaScript is essential for adding interactivity to your currency converter. You can use it to perform currency conversions, fetch data from APIs, and update the UI dynamically.

**3. \*HTML and CSS:** \*The core of your project, HTML is used for structuring the content, while CSS is employed for styling and layout.

**4. \*Responsive Design Frameworks**: \*Frameworks like Bootstrap or Foundation can simplify the process of creating a responsive and visually appealing user interface.

**5. \*Front-End JavaScript Libraries:** \* Libraries like jQuery can make DOM manipulation and AJAX requests more convenient.

**6. \*Font Icons:** \*Consider using font icon libraries like Font Awesome for currency symbols and other icons in your UI.

**7. \*Code Editors:** Choose a code editor that suits your preferences, such as Visual Studio Code.

**8.\*Security Services:** \*Implement security best practices in your HTML/CSS project, including secure data transmission (HTTPS) and protection against common web vulnerabilities.

**2.2 Tools and Softwares Used:**

Creating a currency converter project involves a combination of front-end and back-end development, data retrieval, and user interface design. Here's a list of tools and software commonly used for different aspects of a currency converter project:

**\*Front-End Development: \***

**1.\*HTML:** \* For structuring the project's content.

**2. \*CSS: \*** For styling the user interface and making it visually appealing.

**3. \*JavaScript:** \* To add interactivity and perform currency conversion calculations.

**4. \*Code Editor:** \* A code editor like Visual Studio Code, Sublime Text, or Atom for writing and editing HTML, CSS, and JavaScript code.

**5. \*Version Control: \*** Git and platforms like GitHub or GitLab for version control, collaboration, and code sharing.

**\*Back-End Development (Optional, if required): \***

**6. \*Server-Side Language:** \* A server-side language like Java, or Node.js to handle API requests or database interactions.

**7. \*Web Framework:** \* A web framework such as Express.js (Node.js), for building server-side logic.

**8. \*API Integration:** \* To fetch real-time currency exchange rate data, you can use APIs like Open Exchange Rates, Fixer.io, or a central bank's API.

**\*Data Storage (Optional, if required): \***

**9. \*Database:** \* A database system (e.g., MySQL, PostgreSQL, MongoDB) for storing historical exchange rate data or user-specific information.

**\*APIs and Data Sources: \***

**10. \*Currency Exchange Rate API: \*** An API service like Open Exchange Rates, Fixer.io, or Currency Layer to fetch real-time exchange rate data.

**2.2.1 CSS**

CSS (Cascading Style Sheets) plays a crucial role in the Currency Converter project, as it's responsible for styling the user interface and making it visually appealing. Here's how you can use CSS effectively in your currency converter project:

**\*1. External Stylesheet:** \*Create an external CSS file (e.g., `styles.css`) to separate your styles from the HTML code. Link this stylesheet to your HTML document using the `<link>` tag in the `<head>` section.

html

<link rel="stylesheet" type="text/css" href="styles.css">

**\*2. Maintain a Clean and Organized Structure**: \*Organize your CSS code by grouping styles based on their purpose. Use comments to label different sections of your stylesheet. Follow a consistent naming convention for CSS classes and IDs.

**\*3. Selectors**: \*Use various CSS selectors to target HTML elements. For example: Element selectors: `body`, `h1`, `div`, etc. Class selectors: `.button`, `.input-field`, `.currency-dropdown`, etc.

**\*4. Colors:** \*Set the color scheme for your project using properties like `color`, `background-color`, and `border-color`. Consider using variables (CSS custom properties) for consistent color management.

**\*5. Typography**: \*Define font styles for text elements like headings, paragraphs, and buttons using the `font-family`, `font-size`, `font-weight`, and `font-style` properties.

**\*6. Layout and Positioning**: \*Use CSS for layout control. Employ properties like `width`, `height`, `margin`, `padding`, `display`, and `position` to control the positioning and sizing of elements. Implement responsive design using media queries to adapt your layout for different screen sizes.

**\*7. Transitions and Animations:** \*Apply transitions and animations to enhance the user experience, such as smooth hover effects or subtle animations for currency conversions.

**\*8. Customization and Theming**: \*Allow users to customize the theme or appearance of the currency converter using CSS variables and user-selectable themes.

CSS is a powerful tool for creating visually appealing and responsive user interfaces. By using it effectively in your currency converter project, you can provide users with a pleasant and user-friendly experience.

**2.2.2 HTML**

Creating the HTML structure for a currency converter project involves defining the necessary elements and layout for your user interface. Here's a simple HTML template to get you started. You can then customize and expand it to suit your project's specific requirements:

The HTML document is structured with a <head> section for metadata and a <body> section for the content.

The <header> contains the project's title.

The <main> section includes a form for currency conversion. It consists of dropdowns for selecting the source and target currencies, an input field for entering the amount, and an input field to display the result. The "Convert" button triggers the conversion process.

The <footer> includes a copyright notice.

Start by creating the basic HTML structure for your currency converter. You will need an input field for the amount to be converted, two dropdown menus for selecting the currencies, and a button to perform the conversion.

* + 1. **JAVASCRIPT**

**1.Event Handling:** JavaScript is used to handle user interactions such as button clicks or dropdown selections. Event listeners are added to HTML elements (e.g., buttons) to trigger specific actions when events occur.

**2. DOM Manipulation**: JavaScript is used to interact with the Document Object Model (DOM) to read and modify HTML elements. You can access and update input values, dropdown selections, and result displays using JavaScript.

**3. Data Storage:** JavaScript can store exchange rates and currency data either manually as objects or by fetching them from an external API. Arrays or objects are used to store currency conversion data.

**4. Calculations:** JavaScript is responsible for performing the currency conversion calculations based on user input and exchange rates. You'll use mathematical operations to convert the currency amount from one currency to another.

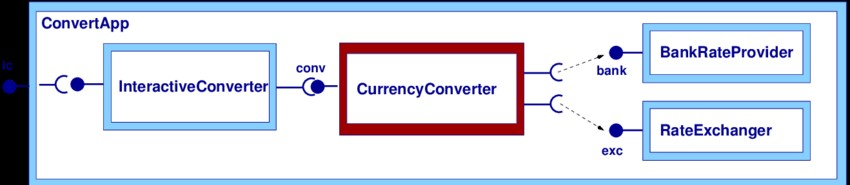
**5. Displaying Results**: JavaScript updates the HTML to display the converted amount to the user. The result is typically shown dynamically on the webpage after the conversion is performed.

**6. Error Handling:** JavaScript code can include error handling to ensure that user input is valid and that the conversion

**CHAPTER 3**

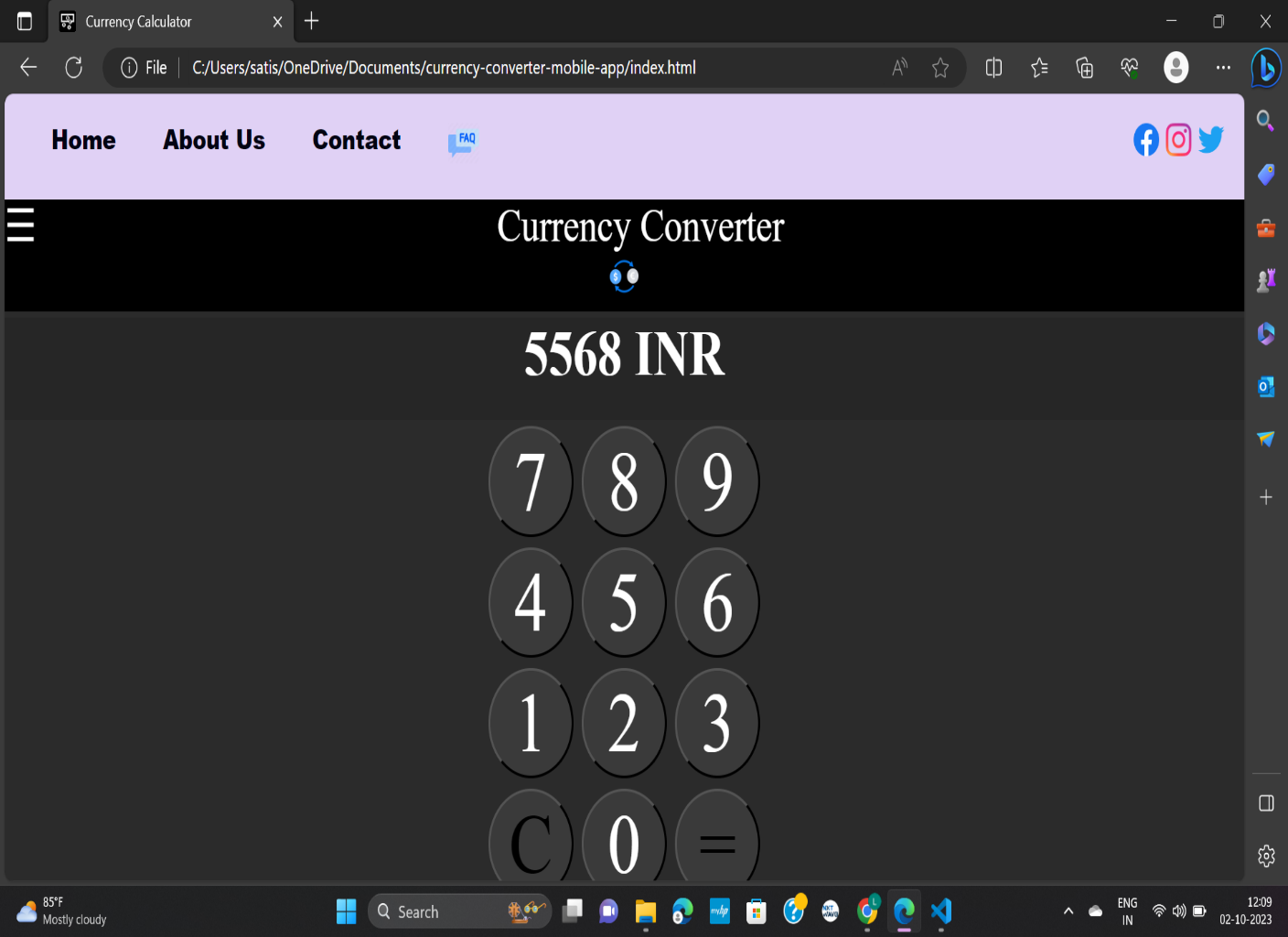
**PROJECT ARCHITECTURE**

**3.1 Architecture**



**3.2 Implementation**

© Edunet Foundation. All rights reserved.

****

**CHAPTER 4**

**ARCHITECTURE BLOCKS DETAIL WORKING**

**4.1 Blocks**

**1. \*User Interface (UI): \***

UI is the user-facing part of the application where users input the amount to convert and select the source and target currencies.

It displays the result of the currency conversion and provides a user-friendly experience.

User interactions trigger events that are processed by JavaScript.

**2. \*HTML/CSS/JavaScript: \***

HTML is used to structure the UI elements, including input fields, dropdowns, buttons, and result displays.

CSS styles the UI elements to create an appealing and consistent visual design.

JavaScript adds interactivity to the UI, handling user input and triggering currency conversion calculations.

**3. \*Currency Data: \***

Currency data is essential for performing accurate conversions. It includes exchange rates between different currencies.

This data can be obtained from an external currency exchange rate API or, in some cases, from a local database.

**4. \*AJAX/HTTP Requests: \***

JavaScript makes asynchronous requests to fetch currency exchange rate data.

These requests are sent to either a back-end server or external currency exchange rate API.

**5. \*Back-End:** \*

A back-end server may be used to handle requests, especially if the project requires server-side logic (e.g., user authentication, data caching, or rate limiting).

The back-end can integrate with external APIs or databases to fetch or store currency data.

**CONCLUSION**

In conclusion, a currency converter is a valuable tool that simplifies international financial transactions by providing real-time exchange rate information and enabling users to convert one currency into another quickly and accurately. Whether for personal travel, business transactions, or investment decisions, currency converters play a crucial role in the global economy, facilitating seamless currency conversions and financial planning across borders.

when person transfer the money from one country to another country then everyone needs to convert their currency, by keeping this thought on mind, I develop the currency converter system, so it is very simple to calculate like application. The currency converter has a simple interface in which user can enter the amount and choose the changes.

Very simply, a currency converter uses exchange rates to show users how the values of two currencies are related. The exchange rate is the cost of money from one currency to another.

Currency can be converted using an online currency exchange, or it can be performed manually. To use either method, you must first look up the exchange rate using an online exchange rate calculator or by contacting your bank.

finance, an exchange rate is the rate at which one currency will be exchanged for another currency.

The architecture of a currency converter project can vary in complexity based on its requirements and scale. However, I'll provide a high-level overview of a typical architecture for a currency converter web application. This architecture involves both front-end and back-end components to provide real-time currency conversion services.

In our increasingly interconnected world, the need for currency conversion is a fundamental aspect of international trade, travel, and finance. Whether you're a business dealing with international clients, a traveler exploring foreign lands, or an investor diversifying your portfolio, the ability to quickly and accurately convert one currency to another is invaluable. To meet this demand and simplify the process, we have developed a Currency Converter Process.

**REFERENCES**

Open Exchange Rates: Open Exchange Rates provides a simple API to access exchange rate data. You can sign up for an API key and use their endpoint to fetch exchange rates.

**Rates Website: Open Exchange**

**Exchange Rate-API:** Exchange Rate-API is a free and simple API to get exchange rate and currency conversion data. It offers a free tier with limited requests per month.

**Website: Exchange Rate-API**

**Fixer.io**: Fixer.io is an open-source API that provides exchange rate data. It's free to use and offers both a free and a paid plan.

**GitHub Repository: Fixer.io**

**Currency Layer:** Currency Layer offers a simple REST API with real-time and historical exchange rate data. They have both free and paid plans.

**Website: Currency Layer**

**Xe.com (Xe Currency Data**): Xe.com provides currency data through their API. They offer various plans, including a free plan with limited features.

**Website: Xe Currency Data**

**Open Rates**: Open Rates is another free and open API for currency exchange rates. It provides JSON data for various currencies.

**Website: Open Rates**

To implement a currency converter, you'll typically make HTTP requests to these APIs with the source currency, target currency, and the amount you want to convert. The API will respond with the conversion result based on the current exchange rate.

**CODE**

[Samplereddygithub/currency\_converter](https://github.com/Samplereddygithub/currency_converter)