#### A

Mini Project Report On

MEASURE MATE

**For**

Partial fulfillment of award of the B.Tech. Degree

in Information Technology



**2024-25**

**Mr. Adityan Gupta**

(Assistant Professor)

**Team Members**

Shyam Sunder Sharma (2301920130180) Shreyansh Singh Rana (2301920130173) Shivansh Mishra (2301920130167)

**Department of Information Technology**

G. L. Bajaj Institute of Technology and Management Plot No 2, Knowledge Park-III, Greater Noida-201306

2024-2025



Department of Information Technology

# Declaration

I/We here with declare that the project work conferred during this report entitled “**Measure-mate**”, in partial fulfillment of the necessity for the award of the degree of Bachelor of Technology in Information Technology, submitted to A.P.J. Abdul Kalam Pradesh Technical University, Uttar Pradesh, is an authentic record of my/our own work distributed in Department of Information Technology & Engineering, G.L. Bajaj Institute of Technology & Management, Greater Noida. It contains no material antecedently printed or written by another person except wherever due acknowledgement has been created within the text. The project work reported during this report has not been submitted by me/us for award of the other degree or certification.

Signature:

Name: Shreyansh Singh Rana Roll No :2301920130173

Signature:

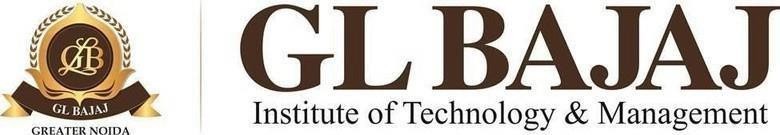
Name: Shivansh Singh Mishra Roll No:2301920130167

Signature:

Name: Shyam Sunder Sharma Roll No :2301920130180

Date:

Place: Greater Noida



Department of Information Technology

# Certificate

This is to certify that Project Report entitled “**Measure-mate**” that is submitted by **Shyam Sunder Sharma, Shreyansh Singh Rana and Shivansh Mishra** in partial fulfillment of the necessity for the award of degree B. Tech. in Department of Information Technology of Abdul Kalam Technical University, are record of the candidate own work distributed by him below my/our oversight. The matter embodied during this report is original and has not been submitted for the award of the other degree.

Adityan Gupta Dr. P C Vashist

(Assistant Professor) Head of Department





Department of Information Technology

# Acknowledgement

We would like to express our sincere thanks to our project supervisor **Mr.Adityan** Gupta and our Head of department **Dr. P.C Vashist** for their invaluable guidance and suggestions. This project enriches our knowledge and experience of working in a team and a live project. Also, we would like to express gratitude to **Mr.Adityan** Gupta for his/her help in preparation and overview of our project. Lastly, we would like to thank all the faculties for providing their valuable time whenever needed for helping us carry on with our project.

**TABLE OF CONTENTS**

[ABSTRACT IV](#_bookmark0)

ABBREVATION USED V

[CHAPTER-1: INTRODUCTION TO UNIT CONVERTER 1](#_bookmark1)

* 1. INTRODUCTION 1
  2. BACKGROUND 2
  3. [MOTIVATION 3](#_TOC_250002)
  4. PURPOSE AND NEEDS 4
  5. TARGET AUDIENCE………………………………………………………..7
  6. PROJECT SCOPE…………………………………………………………….8

CHAPTER-2: EXISTING TECHNOLOGIES 9

* 1. FRONT-END TECHNOLOGIES 9
  2. BACK-END TECHNOLOGIES 9

[CHAPTER-3: METHODOLOGY 10](#_bookmark2)

CHAPTER-4: CONCLUSION 16

CHAPTER-5: FUTURE SCOPE 17

[REFERENCES 18](#_bookmark3)

## ABSTRACT

Unit and Currency Converter is a practical tool designed to facilitate the conversion of various units of measurement and currencies. With the increasing need for global interactions, this tool plays a crucial role in simplifying the process of converting between different units (e.g., length, weight, temperature, volume) and currencies (e.g., USD, EUR, INR). The converter relies on predefined mathematical formulas for unit conversions and real-time exchange rate data for currency conversions, ensuring accuracy and reliability.

This tool is developed to be user-friendly, with a simple interface that allows quick input and output of conversions. The currency converter, in particular, fetches up-to-date exchange rates from reliable sources, providing users with the most accurate and current financial data. The unit converter supports a wide range of units and makes calculations fast and efficient, reducing the likelihood of errors in manual conversions.

The project demonstrates the application of basic programming concepts such as API integration, real-time data processing, and user interface design. It serves as a valuable utility for personal, educational, and professional use in sectors such as travel, finance, engineering, and science.

In conclusion, the Unit and Currency Converter is an essential tool that enhances efficiency

and accuracy in conversions, offering users a seamless experience in dealing with various

measurement systems and currencies.

## ABBREVIATION USED

**HTML**: HyperText Markup Language.

**CSS**: Cascading Style Sheets.

**HTTP**: HyperText Transfer Protocol **API**: Application Programming Interface **JS**: JavaScript

#### 

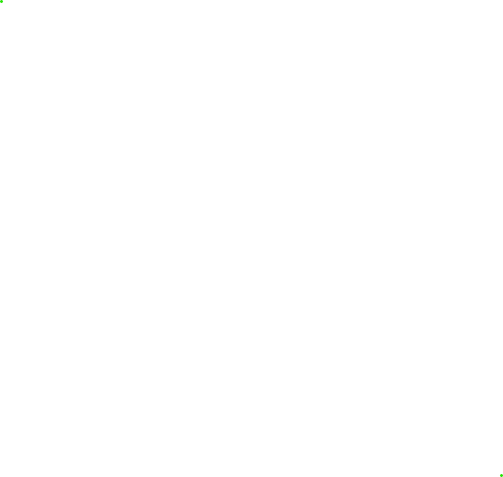
#### **CHAPTER****-1 INTRODUCTION TO UNIT CONVERTER**

This chapter introduces the concept of unit and currency conversion, outlining its importance, scope, and the objectives of creating a versatile, user-friendly converter. It also briefly discusses the challenges and limitations faced in traditional conversion methods and sets the context for the modern technologies explored in later chapter.

* 1. **INTRODUCTION**

In today’s interconnected and highly digital world, converting units and currencies efficiently and is essential across various industries, professions, and everyday scenarios. Whether for personal use, scientific research, engineering projects, or global business operations, the need for seamless unit and currency conversion has become increasingly critical. Unit and currency conversion tools simplify complex calculations, reduce errors, and save time, making them indispensable in various applications.

The Unit Converter and Currency Converter project is designed to simplify the process of converting measurements and currencies. In everyday life, people often encounter the need to convert between various units of measurement, such as length, weight, temperature, and more.

Similarly, for international trade, travel, or financial transactions, converting between different currencies with up-to-date exchange rates is essential. This project aims to create an efficient tool that allows users to perform both unit and currency conversions in one convenient platform.

The Unit Converter handles conversions for physical quantities such as length (meters to feet), weight (kilograms to pounds), temperature (Celsius to Fahrenheit), and others. It uses pre- defined conversion factors and formulas to ensure accurate results across various unit types.

The Currency Converter helps users convert amounts between international currencies, utilizing real-time exchange rates sourced from financial APIs. This is particularly useful for travelers, businesses, and financial analysts who need to quickly convert values based on current market rates.

By providing these two key functionalities, the project enables users to easily manage both unit conversion and currency exchanges in a simple, reliable, and user-friendly interface.

1

## 1.2 BACKGROUND

The Unit Converter and Currency Converter have become essential tools in modern life, addressing the need for accurate conversions in science, business, travel, and everyday activities. Their development is rooted in the history of measurement systems and the globalization of economies, which has made standardizing and converting units and currencies a practical necessity.

### 

### Background of Unit Converter

Throughout history, different civilizations used varying systems of measurement, often leading to confusion and inconsistency. For example, the Imperial system was widely used in Britain and its former colonies, while the Metric system, introduced by France during the 18th century, became the standard for most of the world. As globalization advanced, the need to easily convert between these systems became clear.

In science and engineering, unit conversions are crucial for ensuring that measurements align with the desired standards. For example, engineers might need to convert kilometers to miles or liters to gallons depending on the region or context they are working in. This need for accuracy and standardization in diverse industries led to the development of unit conversion tools. These tools help users convert between different measurement units, such as length (meters to feet), weight (grams to pounds), and temperature (Celsius to Fahrenheit), making it easier to communicate and work across various systems.

Today, digital tools for unit conversion use mathematical formulas and predefined conversion factors to automate the process, ensuring that users can quickly and accurately switch between units without manual calculations.

### Background of Currency Converter

Before the establishment of standard international exchange mechanisms, barter systems or locally recognized currencies were used. With the rise of foreign trade and international business, the need for a method to convert one country’s currency into another became crucial.

During the 19th and 20th centuries, standardized currency systems like the Gold Standard and, later, the Bretton Woods system helped stabilize exchange rates. However, after the collapse of these systems in the 1970s, the foreign exchange (Forex) market was created, where currencies began to float freely. As exchange rates began to fluctuate based on supply and demand, currency conversion became a constantly changing process, making real-time exchange rate data essential for accurate currency conversion.

Modern currency converters use financial APIs to access current exchange rates from the Forex market or central banks, allowing users to convert between currencies accurately. Whether for travelers, businesses, or online shoppers, currency converters enable individuals and organizations to calculate the equivalent value of a sum of money in another currency, considering real -time market fluctuations.

**1.3 MOTIVATION**

### Motivation in Unit Converter:

The motivation behind creating a unit converter stems from the need to simplify complex conversions between various units of measurement, making everyday tasks more efficient and accurate for individuals and businesses alike. Here are some key motivations driving the development of a unit converter:

* **Simplifies Complex Conversions:** Converts measurements between various units (e.g., length, weight, temperature) with ease.
* **Reduces Errors**: Automates the conversion process, minimizing mistakes.
* **Increases Efficiency:** Saves time in fields like science, engineering, and everyday use.

### Motivation in Currency Converter:

The motivation behind developing a currency converter is driven by the need for fast, accurate, and convenient currency exchange solutions in an increasingly globalized world. Here are the key motivations for creating a currency converter:

* **Real-Time Exchange Rates:** Provides up-to-date currency conversions for accurate financial decisions.
* **Facilitates International Transactions:** Useful for travelers, businesses, and online shoppers handling multiple currencies.
* **Simplifies Financial Planning:** Helps users easily understand foreign currency values and manage budgets.

## 1.4 PURPOSE AND NEEDS

### Unit Converter

**Objective** : Build a tool that converts measurements across various categories such as length, weight, volume, temperature, etc.

###### Categories of Units:

* + **Length**: meter, kilometer, foot, inch, mile, etc.
  + **Weight**: gram, kilogram, pound, ounce, ton, etc.
  + **Volume**: liter, milliliter, gallon, pint, etc.
  + **Temperature**: Celsius, Fahrenheit, Kelvin.
  + **Time**: second, minute, hour, day, etc.
  + **Digital Storage**: byte, kilobyte, megabyte, gigabyte, etc.

###### Conversion Logic:

* + Implement formulas or conversion factors for units within each category. For **example**:

**Length**: 1 meter = 3.28084 feet

**Weight**: 1 kilogram = 2.20462 pounds

###### User Interface:

* + Allow users to input the value and select the unit to convert from and to.
  + Display the converted result in real-time.

###### Extensibility:

* + Make the converter flexible to add more units or categories in the future.
  + Include error handling for invalid inputs or conversions.

### Currency Converter

**Objective**: Create a tool that converts currency values based on real-time exchange

rates.

###### Supported Currencies:

* + Support major global currencies: USD, EUR, JPY, AUD, CAD, INR, CNY, etc.

###### API Integration:

* + Integrate with an external API (e.g., Open Exchange Rates or Currency Layer) to

get up-to-date exchange rates.

* + Schedule API calls periodically to refresh rates or trigger based on user requests.

###### Conversion Logic:

* + Convert from one currency to another using the formula:

Amount in Target Currency = Amount in Base Currency × Exchange Rate

###### User Interface:

* + Let users input the amount, select the base and target currency.
  + Display conversion results instantly.

###### Error Handling:

* + Handle API failures, network issues, and invalid currency inputs gracefully with error messages.

###### Extensibility:

* + Allow the addition of more currencies and support historical rate conversions if needed.

**Needs :**

* **Unit Converter Needs**
* **Conversion Formulas:** Database or logic containing conversion factors for various units.
* **User Interface (UI):** Simple UI/UX allowing users to input values, select units
* **Testing:** Ensure accurate conversion through unit testing for each category and conversion type**.**
* **Localization:** Support for various measurement systems (e.g., metric vs. imperial).

### Currency Converter Needs

* + **API Access**: Subscription to a currency conversion API for real-time exchange rates.
  + **Database**: Store supported currency codes and possibly historical rates.
  + **Security**: Secure API communication and user data protection.
  + **Testing**: End-to-end testing to ensure reliable conversion and handling of API errors or downtime.

### Common Needs:

**Programming Languages/Frameworks**: Select appropriate technologies such as:

* + **Backend**: Python, Node.js, or similar.
  + **Frontend**: HTML/CSS/JavaScript (React, Vue, or Angular for dynamic UI).

**Version Control**: Use Git or another version control system for tracking Changes.

**Documentation**: Clear documentation of the conversion formulas and API integrations.

**Deployment:** Host the application (e.g., on Heroku, AWS, or Digital Ocean).

## TARGET AUDIENCE

A unit converter can serve a broad range of users across various fields. Depending on the types of units included and the complexity of the converter, the target audience can vary from general users to professionals in specific industries. Here's a breakdown of the potential target audiences for a unit converter:

### Students and Educators

**Description**:

Students studying subjects that require unit conversions and educators who need tools to demonstrate concepts.

Examples:

* + **Science Students**: Need to convert units in physics, chemistry, and biology class

(e.g., converting meters to kilometers, milliliters to liters, or Celsius to Kelvin).

* + **Math Students**: Converting between different measurement systems
  + **Educators**: Teachers can use unit converters to illustrate concepts during lesson

### Engineers and Technical Professionals

**Description:**

Professionals in technical fields that frequently involve unit conversions in their work.

**Examples**:

##### **Mechanical Engineers**: Need to convert units related to force, pressure ,

##### dimensions.

##### **Civil Engineers**: Need to convert between units of length, area, volume

##### (e.g., meters to feet, cubic meters to gallons, kilonewtons to pounds-force).

##### **Electrical Engineers**: Convert units related to voltage, resistance, power, and

##### energy (e.g., watts, kilowatt-hours, amperes).

### International Business Professionals

##### **Description:** Business people who work across international borders and frequently with different units measurement and currencies.

* **Examples:**
* **Importers/Exporters:** Converting units related to shipping weights, volumes, dimensions, as well as currency conversions.
* **Sales and Marketing Teams:** Need to convert prices, quantities, and dimension match international markets**.**

### Designers and Builders (e.g., Architects, Interior Designers)

##### **Description**: Professionals who work in design, construction, and architecture, often

converting units related to dimensions, areas, and materials.

##### **Examples**:

##### **Architects**: Converting between metric and imperial systems for building desi

##### (e.g., meters to feet, square meters to square feet).

##### **Interior Designers**: Converting fabric measurements or room dimensions.

* **Builders and Contractors**: Converting measurements for materials like wood,

metal, and concrete (e.g., cubic meters to cubic yards).

### IT Professionals and Data Scientists

**Description**: People working with data storage, bandwidth, and computing resources who need to convert between different digital units.

**Examples**:

* + **IT Managers:** Converting data units such as gigabytes, terabytes, and petabytes**.**
  + **Network Engineers:** Converting bandwidth or data transfer rates (e.g., from megabits per second to gigabits per second).
  + **Data Scientists:** Handling conversions related to memory usage, computing power, and storage.

## 

## PROJECT SCOPE

The project scope for a unit converter outlines the key features, functionalities, and deliverables to ensure that the final product meets the user's needs and expectations. Here's a detailed breakdown.

Develop a unit conversion tool that allows users to convert values between various unit systems (metric, imperial, etc.) across different categories.

Ensure the tool is user-friendly, responsive, and accurate, catering to both general users and professionals in fields like science, engineering, and business.

## CHAPTER-2

## EXISTING TECHONOLOGY

In this chapter, we explore the current state of technology related to unit converters. Understanding the tools, libraries, APIs, and systems that are widely used today can help provide a solid foundation for building and enhancing a unit conversion tool. These existing technologies serve different purposes ranging from basic conversions to real-time data integration for highly dynamic units like currencies.

* **HTML** (Hyper Text Markup Language): The backbone of the website, HTML structures the content such as headings, paragraphs, links, and buttons. It forms the basic framework of the unit converter, including the blog posts, navigation, and footer.
* **CSS** (Cascading Style Sheets): CSS is used to style the HTML content, making the unit conversion visually appealing. It defines the layout, colors, fonts, spacing, and positioning. With CSS, the unit conversion’s design is made user-friendly, responsive, and attractive.
* **JavaScript**: JavaScript plays a crucial role in making unit converters interactive, responsive, and dynamic. It handles the logic for converting units, processes user input, and updates the UI in real time. Below is an overview of how JavaScript is typically used in a unit converter.

**APIs:**

**OpenExchangeRates**: Provides real-time exchange rates for global currencies. It supports historical data, time-series analysis, and provides reliable conversions.

**Currency layer**: A simple REST API for real-time and historical exchange rates with currency conversions.

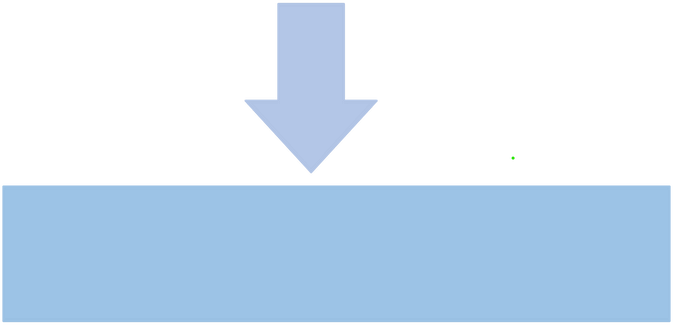
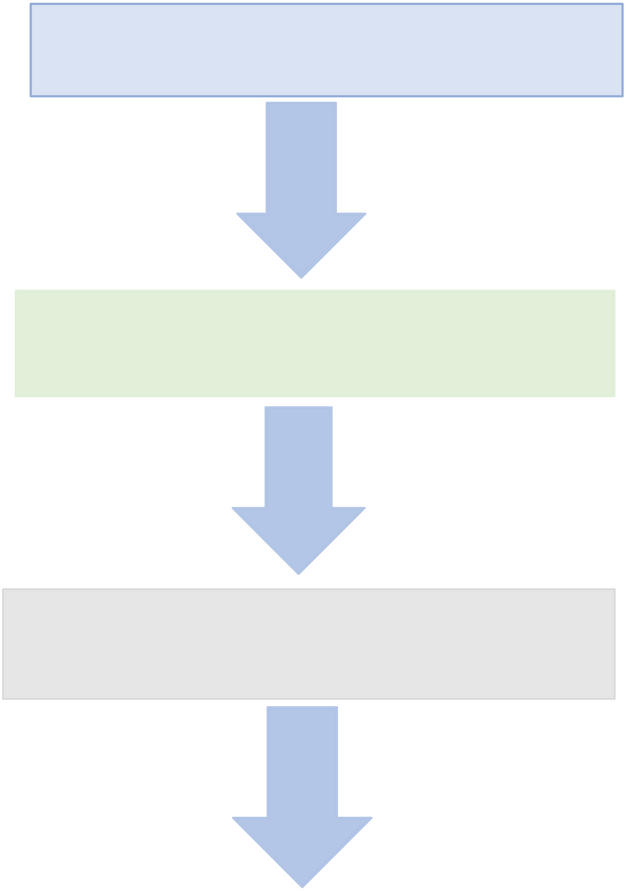
**XE Currency Data API**: Provides accurate exchange rates and supports real-time currency conversions.

**JavaScript Libraries**:

**math.js**: A versatile JavaScript library that offers a broad range of mathematical functions and unit conversion capabilities, from length and mass to more complex compound units. Convert.js: A lightweight library designed specifically for handling simple unit conversions such as length, volume, and temperature.

**CHAPTER-3**

## METHODOLOGY



PROJECTPLANNINGANDANALYSIS

SYSTEMDESIGN

DEVELOPMENTPHASE

TESTINGPHASE

Figure 3.1: Flow chart

This chapter presents the methodology adopted in developing the unit and currency converter. It covers the systematic process followed, including data collection, system design, implementation, and testing phases, all aimed at ensuring the accuracy and efficiency of the converter. Which is explained in a flowchart in Figure 3.1

###### Planning Analysis

The planning and analysis phase is critical in the development of any application, especially one involving real-time data, such as a unit and currency converter. This phase helps define the scope, requirements, and architecture of the project, ensuring that the final product meets user expectations and functions effectively. Below is a detailed breakdown of the planning and analysis for a unit and currency converter project.

**Goal**: Identify the specific requirements for the unit converter.

###### User Requirements:

* The converter should allow users to input a value and choose both the unit to convert from and the unit to convert to.
* Support for different categories of units, such as length, weight, volume, temperature, etc.
* Real-time or button-triggered conversion.
* An intuitive user interface (UI) with clear input fields and dropdowns.
* Option for mobile responsiveness.

###### Functional Requirements:

* Conversion formulas for each unit category.
* Error handling for invalid input (e.g., non-numeric values).
* Performance optimization for fast conversions.

###### Non-functional Requirements:

* Responsive design for different screen sizes.
* A simple and clean user interface.
* Usability and accessibility.

###### Design and Planning

In this section, we will cover the design and planning aspects of the unit and currency converter, including system architecture, user interface design, and project scheduling. Careful planning and an efficient design are crucial for the development of a reliable, scalable, and user-friendly application.

**Goal**: Create a blueprint for the project, including user interface design and system architecture.

**UI/UX Design**:

* **Wireframes**: Create simple wireframes for the layout. This might include input fields for entering values, dropdown menus for selecting units, and a conversion result display.
* **User Flow**: Design the interaction flow (e.g., how the user enters data, selects units, and sees results).

**System Design**:

* **Front-end:** Plan the use of technologies like HTML, CSS, and JavaScript for the UI and conversion logic.
* **Database (Optional):** If you plan to save historical conversions or support more complex calculations, consider a simple database design (e.g., localStorage).

**Technology Stack**:

* **Frontend**: HTML, CSS, JavaScript (React.js, Vue.js, or plain JS depending on project size).
* **Conversion Logic**: Implement conversion formulas using JavaScript**.**
* **Libraries (optional):** Consider using convert-units or other similar libraries to simplify conversion logic.

###### Implementation (Development)

**Goal**: Code the actual functionality of the unit converter according to the design and requirements.

Step-by-Step Breakdown:

###### Set up the HTML Structure:

* Create input fields for the user to enter values.
* Add dropdowns for selecting the unit categories (length, weight, temperature, etc.).
* Provide an area to display the conversion result.

###### Styling (CSS):

* Style the form and the page to ensure that the layout is responsive and visually appealing.
* Use CSS frameworks (e.g., Bootstrap, Tailwind) or custom styles to enhance the UI.

###### Write Conversion Logic (JavaScript):

* Implement the logic to convert values between units. You can define conversion factors for each unit.
* Use functions to handle conversions (e.g., converting meters to feet or Celsius to Fahrenheit).
* Add error handling (e.g., what happens if the user inputs an invalid value or selects the same unit for both input and output**).**

###### Interactivity:

Add event listeners to handle user actions, such as clicking the "Convert" button or selecting a unit from a dropdown. Ensure that the result is dynamically updated based on the user input and selected units.

###### Testing and Validation

**Goal**: Test the unit converter to ensure it works as expected, is user-friendly, and provides accurate results.

**Unit Testing**:

* Test individual functions to ensure that the correct conversions are performed (e.g., meters to feet, kilograms to pounds, etc.).
* Write test cases to validate the input handling and conversion output.

**Integration Testing**:

* + Test how different parts of the application work together (e.g., input handling, conversion logic, and displaying the results).

**User Testing**:

* Allow potential users to test the unit converter and provide feedback on usability, responsiveness, and accuracy.

**Bug Fixes**:

* Identify and resolve bugs or issues found during testing (e.g., incorrect conversion results or UI glitches on different devices).

###### Deployment

**Goal**: Make the unit converter accessible to users.

**Hosting**:

* + Choose a deployment platform for hosting the project.
  + Options include: GitHub Pages (for static sites).
* Ensure the project is fully tested before going live.

###### Maintenance and Updates

**Goal**: Maintain the project by fixing bugs, improving performance, or adding new features.

###### Feedback Loop:

* Collect feedback from users regarding additional units they might want or suggestions for improving usability.

**New Features**:

* Consider adding more categories of units (e.g., time, speed, or data storage).
* Implement additional features like unit preferences or conversion history.

**CHAPTER-4**

## CONCLUSION

In conclusion, the unit and currency converter project highlight the importance of providing users with a seamless way to convert between different units of measurement and currencies.

This tool is highly practical in everyday life, especially in globalized settings where international transactions and varied measurement systems are common.

**Key takeaways**:

1. **Efficiency**: Converters save time and reduce errors by instantly performing complex conversions, making them valuable in various sectors, from science to finance.
2. **Accuracy**: For currency conversion, ensuring accurate and up-to-date exchange rates is crucial, while unit conversion relies on consistent mathematical formulas or predefined standards.
3. **User Experience**: A well-designed interface is essential for user satisfaction. Clear inputs, outputs, and intuitive navigation contribute to an effective tool.

**Learning and Implementation**: Developing such a tool involves integrating various technical aspects, such as working with APIs (for currency rates), building a simple UI, and implementing conversion logic, providing valuable hands-on learning experiences.

Overall, a unit and currency converter is a powerful tool that addresses everyday needs and demonstrates key software development concepts. It offers practical applications in numerous domains and can be expanded to include additional features, such as historical data or advanced conversion types, for even greater utility.

#### **CHAPTER-5**

**FUTURE SCOPE**

While the basic unit and currency converter is highly functional, there are several opportunities for enhancing its features:

**Integration with More Units**: Expanding the range of units for conversion, such as volume, speed, energy, and pressure, could increase the tool's usefulness across different domains.

1. **Real-Time Currency Exchange Rates:** For currency conversion, integrating more sources of real-time exchange rates and providing historical data could make the tool even more reliable and dynamic.
2. **Multi-Language and Multi-Currency Support**: Introducing multi-language options and support for a wider range of currencies will make the tool more accessible to global users.
3. **Mobile and Web Apps:** Developing mobile applications or integrating the converter into websites or e-commerce platforms can increase accessibility and convenience, especially for users on the go**.**
4. **User Customization:** Allowing users to save frequently used conversions or customize the interface can enhance the user experience**.**
5. **Machine Learning Integration**: For currency conversions, machine learning models could predict trends or suggest the best times for conversion based on historical data.
6. **Voice Recognition:** Adding voice command functionality could improve accessibility for users with disabilities or those seeking hands-free interaction**.**
7. **Offline Functionality:** Implementing offline functionality for unit conversions (without the need for an internet connection) could be beneficial, especially in areas with limited connectivity**.**

Overall, the future scope of the unit and currency converter lies in expanding its features,

improving accessibility, and integrating emerging technologies to provide an even more user-friendly and comprehensive tool for a global audience.

**REFERENCES**

###### For unit converter we took help of :-

[https://www.nist.gov/pml/owm/metric-si/si-unit](https://www.nist.gov/pml/owm/metric-si/si-units)s

###### For currency converter we took help of :-

<https://www.xe.com/currency/>

###### For API we used api-key :-

[974d238937731029fcabbe7](http://974d238937731029fcabbe7/)

###### For CSS we took help from :-

<https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Functions>

###### For JS :-

<https://www.geeksforgeeks.org/currency-converter-in-javascript/>

###### [For HTML :-](https://webdesign.tutsplus.com/currency-converter-with-html-css-and-vanilla-javascript--cms-108362t)

[https://webdesign.tutsplus.com/currency-converter-with-html-css-and-vanilla-](https://webdesign.tutsplus.com/currency-converter-with-html-css-and-vanilla-javascript--cms-108362t) [javascript--cms-108362t](https://webdesign.tutsplus.com/currency-converter-with-html-css-and-vanilla-javascript--cms-108362t)