USD Currency Converter Application

Introduction

The USD Currency Converter Application is a user-friendly tool designed to convert amounts from USD to various currencies using real-time exchange rates. This project provides an efficient solution for travelers and businesses needing accurate currency conversion.

Objectives

- Allow users to input a USD amount and select a target currency.

- Fetch real-time exchange rates from a reliable API.

- Display the converted amount in an intuitive graphical user interface (GUI).

- Enhance user experience with straightforward navigation.

Technology Stack

- Python: Primary programming language.

- tkinter: For creating the graphical user interface.

- requests: To handle API requests and fetch exchange rates.

System Design

- User Input: An input field for entering the USD amount.

- Currency Selection: A dropdown for selecting the target currency.

- Conversion Logic: When the 'Convert' button is clicked, the app retrieves the latest exchange rate and calculates the converted amount.

- Output Display: The conversion result is shown clearly in the GUI.

Implementation

1. The user enters the USD amount.

2. The user selects the target currency.

3. The user clicks 'Convert,' triggering the application to fetch the latest exchange rate.

4. The converted amount is displayed.

Testing

- Unit Testing: For individual functions like conversion logic.

- Integration Testing: To ensure smooth interactions between components.

- User Testing: Feedback was gathered to identify and resolve usability issues.

Conclusion

The USD Currency Converter Application effectively meets its objectives, providing a quick and reliable tool for currency conversion with real-time rates. This project enhances skills in software development, particularly in GUI design and API integration.

Future Enhancements

- Support for additional currencies.

- Offline functionality to cache exchange rates.

- Ability to choose both source and target currencies.

Python Code for the Application

```python
import tkinter as tk

from tkinter import ttk

import requests


def fetch_rates():

    url = "https://v6.exchangerate-api.com/v6/e95a18d93e37e78f9875c5ad/latest/USD"

    response = requests.get(url)

    data = response.json()

    return data['conversion_rates']


class CurrencyConverter:

    def __init__(self, root):

        self.root = root

        self.root.title("USD Currency Converter")


        # Create UI Elements
```

```python
        self.create_widgets()

    def create_widgets(self):
        # USD Amount Input
        tk.Label(self.root, text="Enter amount in USD:").grid(row=0, column=0, padx=10,
pady=10)

        self.usd_amount = tk.Entry(self.root)

        self.usd_amount.grid(row=0, column=1, padx=10, pady=10)


        # Target Currency Selection
        tk.Label(self.root, text="Select target currency:").grid(row=1, column=0,
padx=10, pady=10)

        self.target_currency = ttk.Combobox(self.root, values=["EUR", "GBP", "INR",
"AUD", "CAD"])

        self.target_currency.grid(row=1, column=1, padx=10, pady=10)


        # Convert Button
        self.convert_button = tk.Button(self.root, text="Convert",
command=self.convert_currency)

        self.convert_button.grid(row=2, column=0, columnspan=2, pady=20)


        # Result Display
        self.result_label = tk.Label(self.root, text="", font=("Arial", 14))

        self.result_label.grid(row=3, column=0, columnspan=2, pady=10)

    def convert_currency(self):
        try:
```

```python
        # Get input values

        amount = float(self.usd_amount.get())

        currency = self.target_currency.get()



        # Fetch exchange rates

        rates = fetch_rates()



        # Convert currency

        converted_amount = amount * rates[currency]



        # Display the result

            self.result_label.config(text=f"{amount} USD = {converted_amount:.2f} {currency}")

    except KeyError:

        self.result_label.config(text="Invalid currency selection.")

    except ValueError:

        self.result_label.config(text="Invalid USD amount.")


if __name__ == "__main__":

    root = tk.Tk()

    app = CurrencyConverter(root)

    root.mainloop()
```

Final Notes

Ensure you have the 'requests' library installed:

```bash
pip install requests
```

Run the application, enter the USD amount, select a currency, and click 'Convert' to see the result.