Department of Information Technology

Report on Course Based Project

CURRENCY CONVERTER

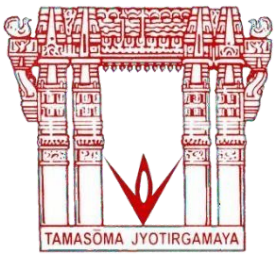
A Course Based Project Report Submitted in partial fulfillment of
the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY IN CSE (CYBERSECURITY)

Submitted by

C. S. Puneet Teja 22075A6201

Under the guidance of
Dr.LALITHA
(Assistant Professor, Department of CSE-CYS, VNR VJIET)



DEPARTMENT OF CSE (CYBERSECURITY)

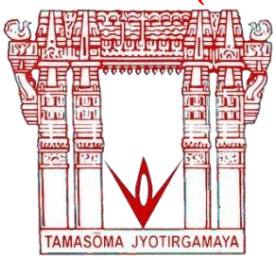
VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute, Accredited by NAAC with 'A++' Grade NBA)
Bachupally, Pragati Nagar, Nizampet (S.O), Hyderabad – 500 090, TS,
India

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute, Accredited by NAAC with 'A++' Grade NBA)
Bachupally, Pragati Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

DEPARTMENT OF CSE (CYBERSECURITY)



CERTIFICATE

This is to certify that the project report entitled "**CURRENCY CONVERTER"** is a bonafide work done under our supervision and is being submitted by **22075A6201 - C.S.PUNEET TEJA** in partial fulfillment for the award of the degree of Bachelor of Technology in CSE (CYBERSECURITY), of the VNRVJIET, Hyderabad during the academic year 2022- 2023. Certified further that to the best of our knowledge the work presented in this thesis has not been submitted to any other University or Institute for the award of any Degree or Diploma.

Project Guide

Dr.LALITHA
Professor,
Dept of CYS,
VNRVJIET
Hyderabad.

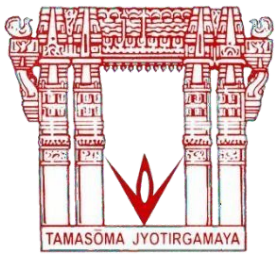
Head of the Department

Dr. RAJASHEKHAR
Head of the dept,
Dept of CYS,
VNRVJIET
Hyderabad

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute, Accredited by NAAC with 'A++' Grade NBA)
Bachupally, Pragati Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

DEPARTMENT OF CSE (CYBERSECURTIY)



DECLARATION

We declare that the major project work entitled "**Currency converter**" submitted in the department of CSE-CYS, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **CSE-CYS** is a bonafide record of our own work carried out under the supervision of Dr. LALITHA. **,Assistant Professor**. Also, we declare that the matter embodied in this thesis has not been submitted by us in full or in any part thereof for the award of any degree/diploma of any other institution or university previously.

Place: Hyderabad

C.S.PUNEET TEJA (22075A6201)

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved Chairman, Shri. D.Suresh Babu, VNR Vignana Jyothi Institute of Engineering &Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our beloved Principal, Dr.C.D.Naidu for permitting us to carry out this project.

We express our deep sense of gratitude to Dr. Rajashekar, Associate Professor and Head, Department of Cyber security, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad for the valuable guidance and suggestions, keen interest and through encouragement extended throughout period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved Guide Lalitha, Professor in Cyber security, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, for his valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion of our project work.

22075A6201 - C.S.PUNEET TEJA

AIM:

To create a program that can give us real time currency exchange rates.

PROBLEM STATEMENT:

The problem is to develop a user-friendly, reliable, and accessible real-time currency exchange rates application that provides accurate currency rates information to users on multiple platforms.

SOFTWARE REQUIREMENT SPECIFICATION (SRS):

S. No	Software Requirement Specification
1.0	Introduction
1.1	Purpose
1.2	Scope
1.3	Technologies to be used
1.4	Tools to be used
1.5	Overview
1.6	Product Perspective
1.7	Interface
1.8	System Functions
1.9	User Characteristics
2.0	Future Scope
2.1	UML Diagrams(Activity Diagram)
2.2	Source Code & Output

1.0 INTRODUCTION

A currency converter is an application that helps with the quick and easy conversion of currencies on the basis of exchange rates. In simple language, a currency converter helps in converting an amount from one currency to another currency. A currency converter is generally used while traveling abroad, by businessmen, and while exporting and importing trades.

1.1 PURPOSE

Currency conversion is of practical use to tourists who travel abroad, to businesses who do business overseas or are involved in imports and exports, and to FX traders. A currency converter is an web application that allows for the quick conversion of any currency into any other currency. One can use a converter to perform entire tasks of calculations. Apart from performing calculations, this tool will always give you regular updates on currency rates. It would be very difficult for investors to convert different currencies without using a currency converter.

1.2 SCOPE

1.2.1 **Currency Exchange Rates Collection**: Gathering currency exchange rates data from various countries through API.

1.2.2 **User Interface:** Developing a user-friendly interface that allows users to easily convert currency into desired country's currency.

1.2.3 **User Support:** Providing support for users, including troubleshooting, answering questions, and addressing concerns

1.3 TECHNOLOGIES TO BE USED

1.3.1 Python

1.3.2 Tkinter

1.3.3 API

1.4 TOOLS TO BE USED

1.4.1 Windows 7/8/10/11 versions

1.4.2 VS Code

1.5 OVERVIEW

The currency converter application called Currency Converter, converts amounts from one currency to two others. You can choose to display different currencies like euros, yen, rupees, dollars, etc and enter a value of currency to be converted into the other selected currencies.

1.6 PRODUCT PERSPECTIVE

User research and testing: Conducting user research and testing to understand user needs and preferences, and to validate product features and functionality.

1.7 INTERFACE

1.7.1 **GUI:** Outputs the weather report using tkinter.

1.8 SYSTEM FUNCTIONS

A real-time currency exchange rates application is a software solution designed to provide up-to-date and accurate currency information of different countries to users. This type of application is essential for individuals and businesses who need to make decisions based on current currency rates**.**

1.9 USER CHARACTERISTICS

Location: To provide currency exchange rates information specific to the user's interests.
Device type: The type of device being used, such as a smartphone, tablet, or desktop computer, can affect the presentation and functionality of the currency converter application.
Demographic information: This can include gender, income, education level, and other demographic information that may influence preferences for weather information and how it is presented.
Businesses: Currency converters help international import and export businesses by helping them determine the selling and buying profits of different products.
Usage patterns: How often the user checks the weather and at what times can affect the presentation and functionality of the weather application.

2.0 FUTURE SCOPE

2.1 **Incorporating machine learning algorithms:** Predict future exchange rates and provide more accurate conversion rates.

2.2 **Integration with other financial services:** could be integrated with online payment platforms or investment apps.

2.3 **Adding more currencies and crypto currencies:** to include more currencies and crypto currencies.

2.4 **Automating conversion process:** Automatically converting currencies in online shopping carts or in international wire transfers.

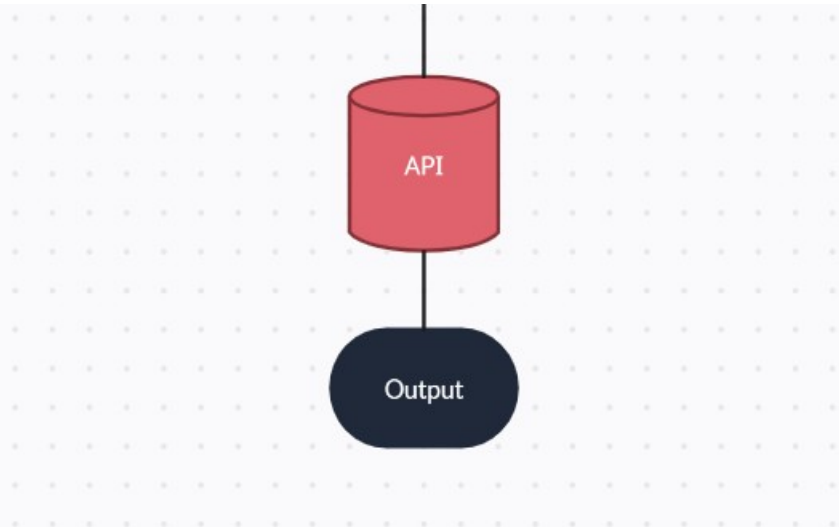
2.5 **Implementing a mobile application and Offering a subscription service:** A subscription service with premium features such as real-time exchange rate alerts or historical exchange rate data.

2.6 **Use of block chain technology:** The project could be integrated with block chain technology to allow for decentralized and secure currency exchange.

2.7 **More functionality for businesses:** The ability to generate invoices in different currencies or track expenses in multiple currencies.

2.1 UML DIAGRAMS

Activity Diagram

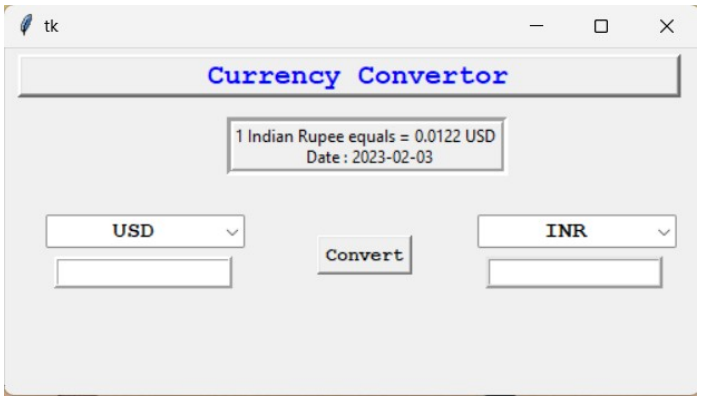


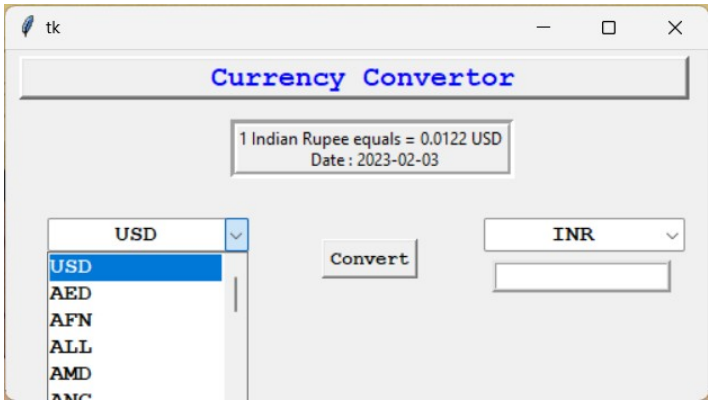
2.2 SOURCE CODE:

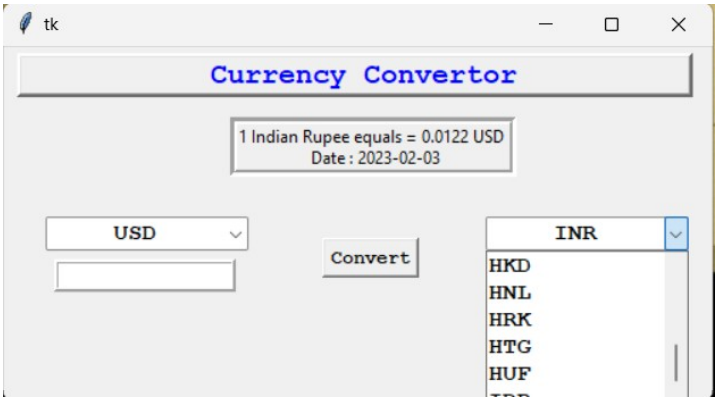
```
import requests
from tkinter import *
import tkinter as tk
from tkinter import ttk


class RealTimeCurrencyConverter():
    def __init__(self,url):
            self.data = requests.get(url).json()
            self.currencies = self.data['rates']

    def convert(self, from_currency, to_currency, amount):
        initial_amount = amount
        if from_currency != 'USD' :
            amount = amount / self.currencies[from_currency]

        # limiting the precision to 4 decimal places
        amount = round(amount * self.currencies[to_currency], 4)
        return amount

class App(tk.Tk):

    def __init__(self, converter):
        tk.Tk.__init__(self)
        self.title = 'Currency Converter'
        self.currency_converter = converter
        self.geometry("500x250")

        # Label
        self.intro_label = Label(self, text = '           Currency Convertor
',  fg = 'blue', relief = tk.RAISED, borderwidth = 3)
        self.intro_label.config(font = ('Courier',15,'bold'))

        self.date_label = Label(self, text = f"1 Indian Rupee equals =
{self.currency_converter.convert('INR','USD',1)} USD \n Date :
{self.currency_converter.data['date']}", relief = tk.GROOVE, borderwidth = 5)

        self.intro_label.place(x = 10 , y = 5)
        self.date_label.place(x = 160, y= 50)

        # Entry box
        valid = (self.register(self.restrictNumberOnly), '%d', '%P')
        self.amount_field = Entry(self,bd = 3, relief = tk.RIDGE, justify =
tk.CENTER,validate='key', validatecommand=valid)
        self.converted_amount_field_label = Label(self, text = '', fg = 'black',
bg = 'white', relief = tk.RIDGE, justify = tk.CENTER, width = 17, borderwidth = 3)

        # dropdown
        self.from_currency_variable = StringVar(self)
        self.from_currency_variable.set("USD") # default value
        self.to_currency_variable = StringVar(self)
```

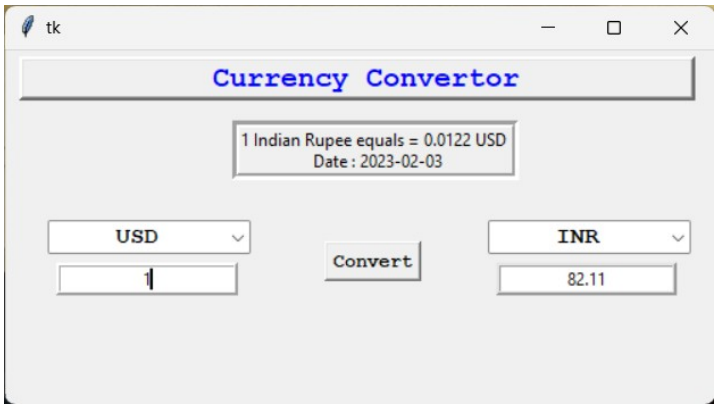
```
        self.to_currency_variable.set("INR") # default value

        font = ("Courier", 12, "bold")
        self.option_add('*TCombobox*Listbox.font', font)
        self.from_currency_dropdown = ttk.Combobox(self,
textvariable=self.from_currency_variable,values=list(self.currency_converter.curre
ncies.keys()), font = font, state = 'readonly', width = 12, justify = tk.CENTER)
        self.to_currency_dropdown = ttk.Combobox(self,
textvariable=self.to_currency_variable,values=list(self.currency_converter.currenc
ies.keys()), font = font, state = 'readonly', width = 12, justify = tk.CENTER)

        # placing
        self.from_currency_dropdown.place(x = 30, y= 120)
        self.amount_field.place(x = 36, y = 150)
        self.to_currency_dropdown.place(x = 340, y= 120)
        self.converted_amount_field_label.place(x = 346, y = 150)

        # Convert button
        self.convert_button = Button(self, text = "Convert", fg = "black", command
= self.perform)
        self.convert_button.config(font=('Courier', 10, 'bold'))
        self.convert_button.place(x = 225, y = 135)

    def perform(self):
        amount = float(self.amount_field.get())
        from_curr = self.from_currency_variable.get()
        to_curr = self.to_currency_variable.get()

        converted_amount =
self.currency_converter.convert(from_curr,to_curr,amount)
        converted_amount = round(converted_amount, 2)

        self.converted_amount_field_label.config(text = str(converted_amount))

    def restrictNumberOnly(self, action, string):
        regex = re.compile(r"[0-9,]*?(\.)?[0-9,]*$")
        result = regex.match(string)
        return (string == "" or (string.count('.') <= 1 and result is not None))

if __name__ == '__main__':
    url = 'https://api.exchangerate-api.com/v4/latest/USD'
    converter = RealTimeCurrencyConverter(url)

    App(converter)
    mainloop()
```

Output:


tk
Currency Convertor
1 Indian Rupee equals = 0.0122 USD
Date : 2023-02-03
USD
Convert
INR


tk
Currency Convertor
1 Indian Rupee equals = 0.0122 USD
Date : 2023-02-03
USD
USD
AED
AFN
ALL
AMD
Convert
INR


tk
Currency Convertor
1 Indian Rupee equals = 0.0122 USD
Date : 2023-02-03
USD
Convert
INR
HKD
HNL
HRK
HTG
HUF


tk
Currency Convertor
1 Indian Rupee equals = 0.0122 USD
Date : 2023-02-03
USD
1
Convert
INR
82.11