**SIMATS ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CHENNAI-602105**

**CSA08-PYTHON PROGRAMMING**

**Course Leaning Outcome**

1. Ability to utilize the syntax and semantics of Python Programming
2. Apply the knowledge in use of control flow statement, handling of strings and functions
3. Create and manipulate lists, dictionaries, tuples and sets
4. Implement the operations of file systems, Modules and Packages to develop solutions to data analysis problems.
5. Employ python code to control and document data science processes
6. Ability to integrate knowledge acquired throughout the course and apply the gained knowledge related to Python Programming
7. Ability to communicate on problem-solving and logical thinking skills to solve real time problems.
8. Ability to interact with peers on Python Programming.

**Course Learning Outcome (CLOs) Vs Capstone Project Mapping**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

**CAP STONE PROJECT**

**REVIEW 1 MARKS RUBRICS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Components** | **Marks(100)** |
| 1 | Novelty of the project | 20 |
| 2 | Communication | 20 |
| 3 | Concept clarity | 20 |
| 4 | Presentation | 20 |
| 5 | Viva | 20 |
|  | **Total** | **100** |

**FINAL REVIEW MARK RUBRICS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Components** | **Marks(100)** |
| **1** | **Problem Statement** | **5** |
| **2** | **Proposed Design Work**   1. Identifying Key Components 2. Functionality 3. Architectural Design | **20**  (10)  (5)  (5) |
| **3** | **UI Design 1. Layout Design**  Flexible Layout  User Friendly  Color Selection   1. **Feasible Elements Used**   Elements Positioning  Accessibility   1. **Elements Function** | **30**  **(15)**  (5) (5)  (5)  **(10)**  (5) (5)  **(5)** |
| **4** | **Project Demonstration**   1. Communication Skills 2. Execution of the project | **40**  (20)  (20) |
| **5** | **Conclusion** | **5** |
|  | **Total** | **100** |





Table for Content

|  |  |  |
| --- | --- | --- |
| **1** | Abstract | **4** |
| **2** | Introduction | **5** |
| **3** | About Python | **5** |
| **4** | Flow Chart | **6** |
| **5** | Code | **7-8** |
| **6** | Output Screen | **9** |
| **7** | Conclusion | **10** |

# [STUDENT NAME]

Abstract:

The CPA (Cost Per Action) Calculator Project aims to provide a robust tool for

advertisers and marketers to evaluate the effectiveness and efficiency of their digital marketing campaigns. As businesses increasingly rely on data-driven decision-making, understanding the cost efficiency of various marketing actions becomes crucial. This project focuses on developing a user-friendly, web-based CPA calculator that enables users to input key metrics such as total cost and number of actions, generating precise CPA values to facilitate informed decision-making.

The CPA Calculator leverages modern web technologies to offer a seamless and intuitive interface, ensuring that users, regardless of technical expertise, can easily access and interpret their marketing performance data. Key features include real-time calculation, historical data tracking, and the ability to compare different marketing strategies. Additionally, the calculator integrates with popular data sources and analytics platforms to streamline data entry and improve accuracy.

By offering actionable insights into marketing expenditure, the CPA Calculator

empowers businesses to optimize their advertising strategies, enhance budget allocation, and maximize return on investment. This project underscores the importance of efficient cost management in achieving marketing success and aims to contribute to more strategic and data-informed marketing practices.

INRODUCTION:

-**>** Welcome to the documentation CGPA Calculator, a Python application developed using the Tkinter library. This tool provides a straightforward and interactive way for users to calculate their Cumulative Grade Point Average (CGPA) based on grades in multiple subjects.

**-> PURPOSE:**

The CGPA Calculator serves as a practical solution for students to assess their academic performance efficiently. By offering a user-friendly interface and real-time calculation, users can quickly determine their CGPA with ease.

->**KEY FEATURES:**

**Dynamic Grade Selection**: Users can input grades for each subject through a dynamic popup interface.

**Real-time CGPA Calculation**: The application provides instant feedback on the overall CGPA as grades are entered.

User-Friendly Design: With simplicity in mind, the interface is designed to be intuitive and easy to navigate.In the rapidly evolving digital marketing landscape, optimizing advertising spend to achieve the highest return on investment (ROI) is a critical goal for businesses of all sizes. One of the key metrics in assessing the effectiveness of marketing campaigns is Cost Per Action (CPA), which measures the cost associated with a specific user action resulting from an ad, such as a purchase, sign-up, or click. Understanding and managing CPA is essential for marketers to ensure their campaigns are both cost-effective and profitable.The CPA Calculator Project is designed to address the need for a practical, accessible tool that simplifies the process of calculating and analyzing CPA values. This project aims to develop a web-based application that allows users to easily input essential data—such as total campaign cost and the number of actions taken—and obtain precise CPA figures. By offering a clear and user-friendly interface, the CPA Calculator will enable users to make informed decisions about their marketing strategies and budget allocation.In addition to basic CPA calculation, the tool will provide features such as real-time updates, historical data tracking, and comparative analysis of different marketing campaigns. This functionality will not only help users track their advertising efficiency but also enable them to identify trends, measure performance over time, and adjust their strategies for optimal results.The CPA Calculator Project is a response to the growing demand for sophisticated yet accessible marketing tools that support data-driven decision-making. By equipping businesses with a reliable and easy-to-use calculator, this project aims to enhance their ability to evaluate and optimize their advertising investments, ultimately contributing to more effective and efficient marketing efforts.

ABOUT PYTHON :

-> Python is a high-level, general-purpose, and interpreted programming language used in various sectors including machine learning, artificial intelligence, data analysis, web

development, and many more.

-> Python is known for its ease of use,powerful standard library, and dynamic semantics. It also has a large community of developers who keep on contributing towards its growth.

-> The major focus behind creating it is making it easier for developers to read and understand, also reducing thelines of code.

-> Python has plenty of features that make it the most demanding and more popular. Let’s read about a few of the best features that Python has:

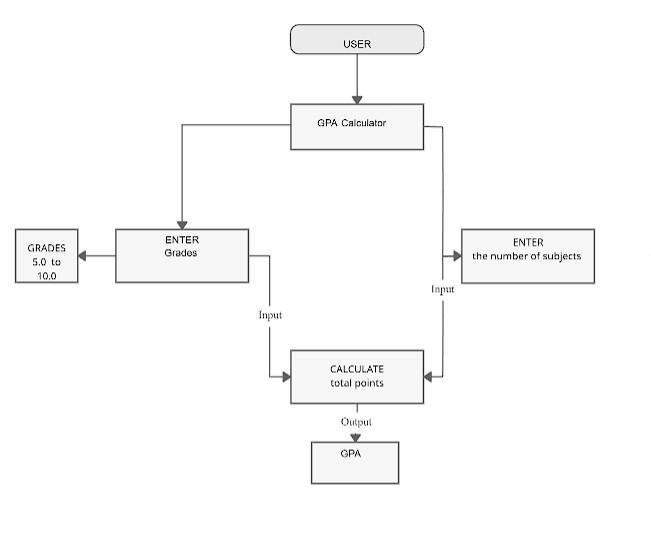
* Easy to read and understand
* Interpreted language
* Object-oriented programming language
* Versatile and Extensible
* Hundreds of libraries and frameworks.
* Flexible, supports GUI
* Dynamically typed

Problem Statement:

In today’s competitive digital marketing environment, businesses are increasingly

tasked with maximizing the effectiveness of their advertising campaigns while managing their budgets efficiently. A critical metric in assessing advertising performance is Cost Per Action (CPA), which measures the cost associated with each specific user action, such as a purchase, sign-up, or click, resulting from an ad.However, many businesses and marketers struggle with the following challenges: Complex Calculation Processes: Accurately calculating CPA often involves complex and error-prone manual calculations or reliance on disparate tools and spreadsheets, which can lead to inconsistencies and inefficiencies. Lack of Real-Time Analysis: Traditional methods for calculating CPA may not provide real-time insights, making it difficult for marketers to make timely adjustments to their campaigns and budget allocations. Limited Integration: Existing tools may not integrate seamlessly with various data sources and analytics platforms, leading to cumbersome data entry and potential inaccuracies in CPA calculations. Insufficient Comparative Analysis: Marketers often lack the ability to easily compare CPA values across different campaigns, channels, or time periods, which hinders their ability to identify trends and optimize their advertising strategies. The CPA Calculator Project seeks to address these issues by developing a user-friendly, web-based tool that simplifies the CPA calculation process, provides real-time data analysis, and integrates with popular data sources. This tool will enable users to calculate CPA accurately, track historical data, and compare performance across different marketing efforts, ultimately leading to more informed decision-making and enhanced advertising efficiency.

FLOW CHART :



CODE:

import tkinter as tk

def select\_grade(grade, popup):

global grades grades.append(grade) if len(grades) == num\_subjects: calculate\_cgpa() popup.destroy()

def calculate\_cgpa():

global grades

scount = grades.count('S') acount = grades.count('A') bcount = grades.count('B') ccount =

grades.count('C') dcount = grades.count('D') ecount = grades.count('E')

total\_grade\_points = (10 \* scount) + (9 \* acount) + (8 \* bcount) + (7 \* ccount) + (6 \* dcount) + (5 \*

ecount) res = total\_grade\_points / num\_subjects result\_label.config(text=f"CGPA is: {res:.2f}")

def create\_popup():

popup = tk.Toplevel(root) popup.title("Select Grade") popup.geometry("200x150")

btn\_frame = tk.Frame(popup) btn\_frame.pack(pady=20)

for grade in ['S', 'A', 'B', 'C', 'D', 'E']:

tk.Button(btn\_frame, text=grade, command=lambda g=grade, p=popup: select\_grade(g, p)).pack(side=tk.LEFT, padx=5) root = tk.Tk()

root.title("CGPA Calculator")

grades = [] num\_subjects = 0

main\_frame = tk.Frame(root) main\_frame.pack(pady=20)

num\_subjects\_label = tk.Label(main\_frame, text="Enter the number of subjects:") num\_subjects\_label.pack()

num\_subjects\_entry = tk.Entry(main\_frame) num\_subjects\_entry.pack()

def create\_subject\_popup():

global num\_subjects

num\_subjects = int(num\_subjects\_entry.get()) for \_ in

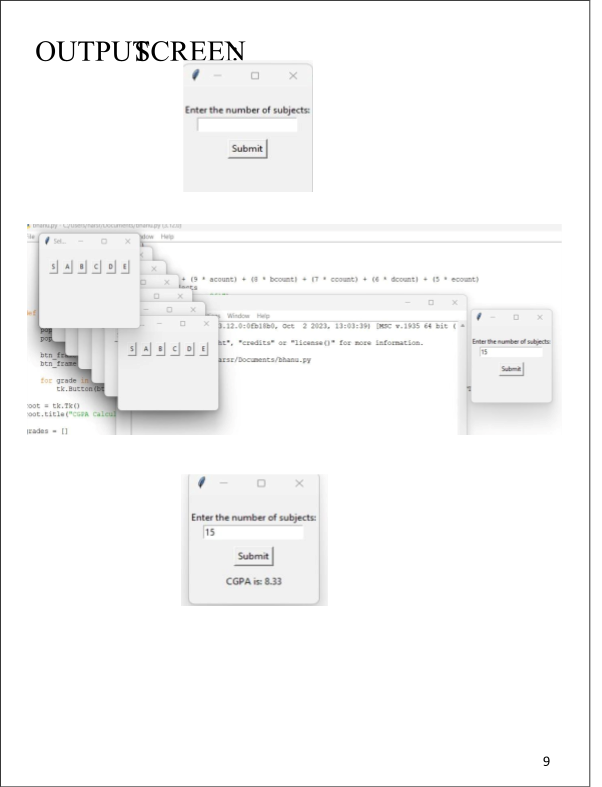
range(num\_subjects):

create\_popup()

submit\_btn = tk.Button(main\_frame, text="Submit", command=create\_subject\_popup) submit\_btn.pack(pady=10)

result\_label = tk.Label(main\_frame) result\_label.pack()

root.mainloop()



Conclusion:

-> The CGPA Calculator provides a simple and efficient solution for

students to calculate their CGPA with ease. Whether you're a student or a developer looking for an example of a Tkinter-based application, this project serves as a valuable resource.

-> Feel free to explore the code, customize it to suit your needs, or use it

as a starting point for your own projects. If you have any questions or feedback, please refer to the code comments or reach out to the developer.