

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS – UNIMINUTO

Informe – Proyecto metodos Python

Estudiantes:

Jon Edilson Castillo Balvuela

Metodología de la Investigación

NRC 10-73392

Nombre del tutor: ALONSO GUEVARA PEREZ

Contenido

Introducción:	3
Análisis	3
Diseño	4
Codificación.....	5

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS – UNIMINUTO

Introducción:

En este informe se explicará la forma en la que se desarrolló el proyecto en el taller planteado de programación orientada a objetos siguiendo la modelo vista controlador. La aplicación tiene como fin calcular funciones matemáticas definidas en el taller (seno, coseno etc) mediante la implementación de series de taylor.

La aplicación tiene como objetivo afianzar los conceptos básicos de programación en Python utilizando la modelo vista controlador planteado en clase y tiene como objetivo resolver problemas matemáticos sin usar funciones integradas del lenguaje Python ni librerías externas que ya realicen el proceso lo que permite entender el funcionamiento interno del porqué de las cosas y profundizar en la lógica detrás las operaciones que se realizan.

Este informe documenta el proceso de análisis y desarrolla desde el planteamiento del problema hasta la implementación con el diseño.

Análisis

Se realizó un planteamiento del problema en el cual se llegó al entendimiento siguiente, crear un programa de funciones totalmente nativas sin usar librerías integradas del lenguaje Python y que propone desarrollar una calculadora capaz de realizar las siguientes operaciones:

- ex
- $\sin(x)$
- $\cos(x)$
- $\sinh(x)$
- $\cosh(x)$

El programa debe estar disponible tanto como para ambiente CLI como web

Requisitos

funcionales:

El sistema debe calcular las funciones expuestas anteriormente

El Sistema debe permitir al usuario que ingrese un parámetro de entrada para el valor de x

El sistema debe manejar conexiones web

El sistema debe ser modular y funcional para CLI y web

Requisitos no funcionales:

Simulación matemática con el modelo planteado en el taller

Portabilidad el código debe ser subido a un controlador de versionamiento

Documentación del código.

Se hace una idea de cómo empezar, por lo cual se debería empezar a construir el paquete modelo vista controlador para posteriormente subirlo a un controlador de versionamiento una vez realizado dicho proceso se procede con la modificación y creación de la lógica del Core del programa.

Diseño

Se plantea un diseño MVC modelo vista controlador para aplicarlo en el programa lo que hace que el código sea más organizado y permite separar la lógica de la funcionalidad cálculos, la lógica de presentación o gui, y el flujo del programa todo esto dentro de una estructura de carpetas ya definidas lo que facilita el mantenimiento en futuros mantenimientos del programa

Estructura:

Metodos_Python/

|

```
|— app/

| |— controllers/

| | |— matematicas_controller.py # Coordina entre vista y modelo

| |

| |— models/

| | |— matematicas_model.py      # Contiene las funciones matemáticas

| |

| |— views/

| | |— matematicas_view.py      # Interfaz en terminal (texto)

| | |— templates/

| | |— index.html              # Interfaz web HTML

| |

|— metodos.py                  # Punto de entrada para la app web (Flask)

|— main.py                     # Punto de entrada para modo consola

|— requirements.txt            # Dependencias del proyecto

|— docs/

    |— informe
```

Codificación

En la codificación del proyecto se llevó a cabo distintas practicas aplicadas en los lenguajes y estructuras de control, el código fue desarrollado en python 3.12 sin usar funciones integradas a esta, es decir desde cero usando tipo de datos primitivos int, float, operadores matemáticos básicos, estructuras de control if else for while.

Se implementaron las funciones siguiendo el modelo de las series de taylor con la integración del MVC.

Validacion de entradas, se estableció una gestión en las entradas o funciones no soportadas devolviendo un mensaje de error desde el controlador lo que mejora el sistema y su capacidad alertas.

Ejemplo de uso de la aplicación.

Desde la terminal ejecutar:

Python main.py

Se iniciará una conexión web y se debe ingresar a <http://127.0.0.1:5000/>