Manual Técnico - Calculadora de Raíces

1. Introducción

Este documento detalla la arquitectura, componentes y funcionamiento interno del proyecto web desarrollado con Django para el cálculo de raíces de funciones polinómicas.

2. Arquitectura General

- \*\*Framework:\*\* Django (seguimiento del patrón MTV: Modelo - Template - Vista).

- \*\*Frontend:\*\* HTML5, CSS3, Bootstrap.

- \*\*Backend:\*\* Python 3 con lógica en archivos separados.

- \*\*Gráficas:\*\* Matplotlib.

3. Componentes Clave

3.1 `views.py`

- Contiene la vista principal `calcular`, que maneja el formulario, ejecuta el método numérico y renderiza resultados.

3.2 `forms.py`

- Define `MetodoForm`, que recoge:

- Funciones ingresadas como texto (ej: "x\*\*2 - 4")

- Valores iniciales (`a`, `b`, `x0`)

- Tolerancia (`tol`)

- Número máximo de iteraciones

- Método seleccionado

3.3 `metodos.py`

- Implementa:

- `biseccion(...)`

- `newton\_raphson(...)`

- `newton\_raphson\_modificado(...)`

- Cada función devuelve:

- Lista de iteraciones (dicts con `x`, `f(x)`, `error`, etc.)

- Mensaje de estado (convergencia, error)

3.4 `graficas.py`

- Utiliza Matplotlib para graficar la función en el intervalo dado.

- Guarda la imagen generada para ser mostrada en la plantilla HTML.

3.5 `templates/raiz/calcular.html`

- Contiene:

- Formulario de entrada

- Tabla con iteraciones

- Imagen con gráfica

- Mensajes de error y resultados

3.6 `urls.py`

- Define la URL principal `/` para acceder a la vista `calcular`.

4. Flujo de Ejecución

1. Usuario accede a `/`
2. Envía el formulario con los datos
3. La vista valida el formulario
4. Se ejecuta el método seleccionado
5. Se generan:
   1. Tabla de resultados
   2. Imagen de la gráfica
6. Todo se renderiza en `calcular.html`

5. Validaciones

- Las funciones se evalúan con `eval` pero en un entorno restringido (`math`, `x`).

- Se manejan errores comunes como:

- Intervalos mal definidos

- División por cero

- Derivadas nulas

- No convergencia

6. Recomendaciones de Seguridad

- Aunque el entorno de evaluación está parcialmente restringido, idealmente debería reemplazarse `eval()` por un parser como `sympy` o `numexpr` para producción.

7. Dependencias

- Python 3.x

- Django

- Matplotlib

8. Estructura de Carpetas Relevante

raiz/

|-- metodos.py

|-- graficas.py

|-- forms.py

|-- views.py

|-- templates/raiz/calcular.html

9. Tecnologías Utilizadas

- Python 3

- Django

- Matplotlib

- HTML5 y CSS3

- Bootstrap (para diseño responsivo)

10. Autor

- Wendy Carolina Tomás Tubac