**Universidad de Oriente**

**Facultad de Ingeniería en Telecomunicaciones, Informática y Biomédica**

**Departamento de Informática**

Informe de la

Práctica Profesional

Título: **“Prototipo de sistema informático para el análisis y toma de decisiones sobre el deterioro de la línea costera de áreas de la región oriental, como parte de un proyecto de investigación de la empresa Geocuba.”**

Autor: Dariel Enmanuel Cabrera López

Tutor: Dr. Dionis López Ramos

Santiago de Cuba, 2023

**Resumen**

El presente Informe de la Práctica profesional se plantea un primer prototipo de sistema informático que permita a la empresa Geocuba la informatización de tareas como: cálculo, búsqueda, almacenamiento de las mediciones realizadas en las líneas costeras, para el realización de este sistema es utilizado la metodología XP (Programación Extrema) ,el lenguaje Python y PyQt para la crear la Interfaz.

Durante el Capítulo 1, se explora las diferentes tecnologías existentes que pueden darle solución a la problemática. Se detallas las herramientas, lenguajes de programación y tecnologías de desarrollo para la implementación del prototipo. Se explica las características y las fases de la metodología empleada.

A lo largo del Capítulo 2, se muestra las particularidades de la construcción de la aplicación de escritorio:

Se definen los actores del Negocio, Los requisitos funcionales y no funcionales, las historias de usuario

Summary

This Professional Practice Report proposes a first prototype of a computer system that allows the company Geocuba to computerize tasks such as: calculation, search, storage of the measurements made on the coastlines, for the realization of this system the XP methodology (Extreme Programming), the Python language and PyQt are used to create the Interface.

Chapter 1 explores the different existing technologies that can solve the problem. The tools, programming languages and development technologies for the implementation of the prototype are detailed. The characteristics and phases of the methodology used are explained.

Throughout Chapter 2, the particularities of building the desktop application are shown: Business actors, functional and non-functional requirements, user stories are defined

Índice

**Introducción 3**

**Capítulo 1: Marco Referencial5**

1.1 Descripción del Negocio 5

1.2 Estados del Arte 5

1.3 Herramientas y Tecnologías de Desarrollo 11

1.4 Metodología de Desarrollo13

**Capítulo 2: . Planificación y Diseño** 1**5**

2.1 Actores del Sistema15

2.2 Funcionalidad del Sistema15

2.3 Historias de Usuario16

2.4 Arquitectura del Sistema21

2.5 Prototipo de Interfaz23

**Conclusiones**27

**Anexos**28

**Referencias Bibliográficas**29

**Introducción**

El cambio climático ha sido causado por las actividades antrópicas que durante cientos de años se han desarrollado a nivel mundial, estas han ocasionado un impacto negativo y considerable como el aumento de la temperatura de la Tierra relacionado directamente con la emisión de gases de efecto invernadero (GEI) generados por las actividades productivas, sistemas de transporte con uso de combustibles fósiles, inadecuada gestión de residuos sólidos y en general, por el uso desmesurado de recursos naturales y pocas prácticas de sostenibilidad ambiental. (Medina, s. f.)

Este es una amenaza real para las costas de todo el mundo. El aumento del nivel del mar, la erosión costera y la acidificación de los océanos son algunos de los efectos más graves del cambio climático en las costas

En Cuba la erosión costera es un problema grave. (Ver Anexo 1).

La Tarea Vida que es un plan del estado cubano para el enfrentamiento del cambio climático. (Ver Anexo 2)

Una de las empresas que está realizando aportes a la tarea Vida es la empresa Geocuba (Ver Anexo 3)

**Situación Problémica**

La empresa GeoCuba para el cálculo de transporte de sedimentos emplea una ecuación y recoge datos de las mediciones de las costas a través de métodos tradicionales, los cuales pueden tener un deterioro en el tiempo, errores en los cálculos realizados y morosidad en la búsqueda de información. Para ello se necesita guardar la información de una manera segura, calcular los datos con mayor precisión y buscar los datos con mayor rapidez.

**Objetivo**

Desarrollar una aplicación informática que permita calcular, gestionar y almacenar los datos para el cálculo del transporte de sedimentos en las costas cubanas.

**Desarrollo**

**Capítulo 1. Marco Referencial**

En este capítulo se explicanlos principales aspectos teóricos, los conceptos de las tecnologías y la caracterización de las herramientas computacionales utilizadas

* 1. **Descripción del Negocio**

La dinámica sedimentaria es un proceso clave que permite cuantificar los cambios que se producen en las zonas costeras, dando lugar a fenómenos como la erosión y la sedimentación o azolvamiento. El conocimiento de la complejidad de este proceso data de tiempos pasados, convirtiéndose en una de las tareas más importantes dentro de la ingeniería de costa definiéndose a través del cálculo del transporte litoral de los sedimentos.

En análisis de los procesos costeros uno de los factores que se considera es el cálculo del transporte longitudinal de arena, existiendo para tal propósito una serie de fórmulas de carácter empírico que predicen dicho transporte. Estas fórmulas contienen constantes de calibración probadas en ensayos de laboratorio y aplicados en diferentes playas.

Para el cálculo existen diferentes métodos de formulación, el Método del Flujo de Energía, que está basado en la energía de la ola, es conocida en la actualidad como la fórmula de CERC.

En los términos de volumen la ecuación de CERC expresa:

Ecuación 1:

Q=K ()sen(2α)

Donde ρ Densidad del agua del mar

ρs, densidad de la arena

n, coeficiente de porosidad

α,ángulo de rompiente

k, índice de rompiente

g, aceleración gravitacional

Hb, altura

K, puede tener 2 valores; el recomendado por Komar e Inman (K=0.77), por ser el de uso más generalizado en trabajos de investigación; y el coeficiente K propuesto por Del Valle,Medina y Losada:

K= 1.4

D50(grano medio) = 0.48 mm

* 1. **Estados del Arte**

**** Figura 1.1 Logo de Tkinter

Tkinter es uno de los paquetes de Python más populares y también es la forma estándar para crear interfaces Gráficas de usuario de escritorio en Python.

Tkinter es la combinación de Tk; un toolkit open source para crear Widgets de elementos como botones, menús, Canvas, textos; y así más código de Python que te permite controlar estas interfaces.

Aunque tkinter solo permite crear interfaces básicas, y sus widgets pueden llegar a ser difíciles de de debuggear

No posee ningún tipo de software para diseñar interfaces visualmente, y aunque hay un proyecto llamado Tkinter Designer que convierte diseños hechos de Figma a código de Tkinter, aun así, no están fácil personalizar los proyectos grandes. (*Frameworks de Python para Desktop | Fazt Web*, 2022).

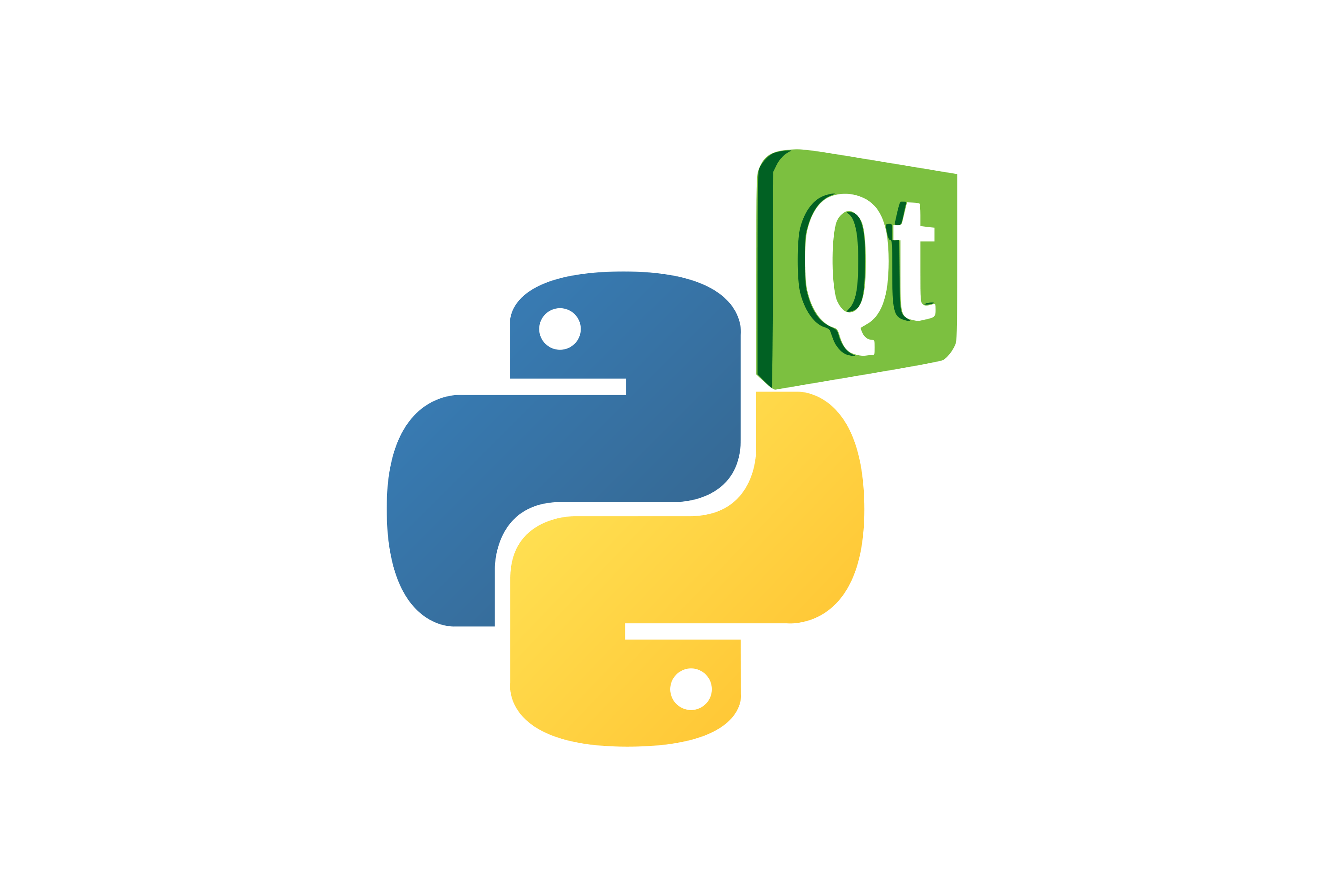
Ventajas:

* Es una biblioteca GUI incorporada estándar para Python, lo que significa que es fácil de instalar y usar.
* Utiliza como una sintaxis simple y proporciona muchos widgets comunes, como etiquetas, botones, cuadros de texto, etc.
* Utiliza widgets nativos en Mac y Windows, lo que significa que puede combinarse bien con la apariencia del sistema.
* Es rápido y ligero, lo que significa que puede funcionar sin problemas en dispositivos de gama baja.

Desventajas:

* No es muy atractivo ni tiene un aspecto moderno, a diferencia de otros frameworks que ofrecen más personalización y temas.
* A veces es difícil depurar o controlar errores, especialmente cuando se utilizan varios subprocesos o procesos.
* No es adecuado para desarrollar aplicaciones móviles o multitáctiles, ya que no admite eventos táctiles ni gestos. (*Codificación BREX*, s. f.)

**PyQt**

**** Figura 1.2 Logo de PyQt

PyQt es una de las opciones más robustas para crear aplicaciones de escritorio avanzadas en Python, está basada en el framework Qt (desarrollado por Nokia), el cual es multiplataforma, lo que significa que puedes desarrollar apps para Windows, Linux, Mac y Raspberry, usando el mismo código base, además de ser orientado a Objetos. Qt Ofrece principalmente widgets desarrollados en C++, pero ofrece bibliotecas de otros lenguajes para interactuar con estos widgets, de aquí es donde sale PyQt.

### Ventajas de usar PyQt

1. Flexibilidad de codificación - La programación GUI con Qt está diseñada alrededor del concepto de [señales y slots](https://pythonspot.com/pyqt5-signals-and-slots/) para establecer comunicación entre objetos. Esto permite flexibilidad cuando se trata de eventos GUI y resulta en una base de código más fluida.
2. Más que un marco de trabajo - Qt utiliza una amplia gama de APIs de plataformas nativas con el propósito de redes, creación de bases de datos y mucho más. Ofrece acceso primario a ellos a través de una API única.
3. Varios componentes de la interfaz de usuario - Qt ofrece varios widgets, como botones o menús, todos ellos diseñados con una apariencia básica en todas las plataformas soportadas.
4. Varios recursos de aprendizaje: dado que PyQt es uno de los marcos de trabajo de interfaz de usuario más utilizado para Python, puede acceder fácilmente a una amplia gama de documentación.

### Desventajas de usar PyQt

1. Falta de documentación específica de Python para las clases de PyQt5
2. Requiere mucho tiempo para entender todos los detalles de PyQt, lo que significa que es una curva de aprendizaje bastante empinada.

**Kivy**

**** Figura 1.3: Logo de Kivy

Kivy es un framework Open Source (OpenGL ES2), para crear aplicaciones multiplataforma tanto en Desktop (Windows, Linux, Mac), móviles (iOS, Android) en incluso Raspberry Pi. Todo con una sola base de código. Todo el framework está desarrollado combinado Cpython y Python. Aunque sus módulos son inestables dando errores difícil de manejar algunas veces.

Y aunque este framework te permita crear apps móviles, muchos desarrolladores actualmente no están muy interesados en crear apps usando Python, por lo que este framework tampoco es ampliamente utilizado en ese aspecto.

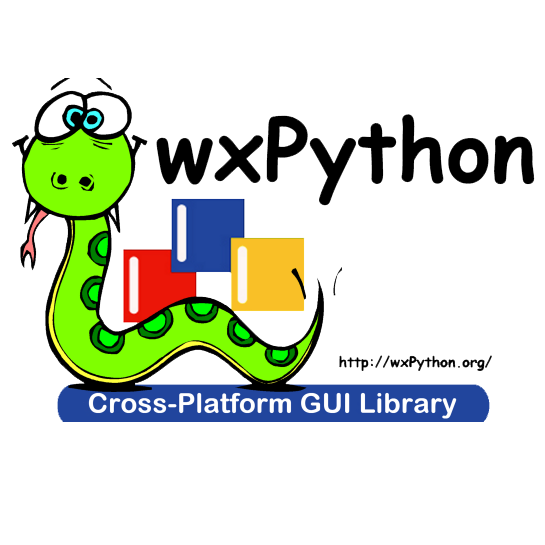
Ventajas:

* Es un marco de código abierto que permite el desarrollo multiplataforma para Windows, Mac, Linux, iOS y Raspberry Pi. Esta flexibilidad facilita a los desarrolladores la creación de aplicaciones multiplataforma sin tener que reescribir el código específicamente para cada plataforma.
* Ofrece un amplio conjunto de widgets y funciones para crear interfaces de usuario innovadoras y dinámicas, como animaciones transiciones, gráficos, etc.
* Admite eventos y gestos multitáctiles, lo que lo hace ideal para desarrollar aplicaciones móviles o táctiles.
* Tiene un lenguaje declarativo llamado ky que permite la creación de prototipos y la personalización de los elementos de la interfaz de usuario de forma fácil y rápida. (*Codificación BREX*, s. f.)
* Sintaxis simple e intuitiva: Kivy por su sintaxis simple e intuitiva, facilita la codificación. La naturaleza sencilla de la sintaxis permite a los usuarios crear rápidamente widgets y escribir código sin ninguna confusión.
* Buen apoyo de la comunidad: Numerosos revisores han destacado la disponibilidad de un sólido sistema de apoyo de la comunidad como un aspecto positivo del uso de Kivy. Este sistema de soporte incluye foros y canales de IRC donde los usuarios pueden buscar ayuda, compartir ideas y colaborar con otros miembros de la comunidad. La amabilidad y la amabilidad de la comunidad en línea han sido elogiadas por los usuarios. (*Pros and Cons of Kivy 2024*, 2023)

Desventajas:

* Requiere una instalación y dependencias separadas, lo que puede ser complicado o llevar mucho tiempo para algunos usuarios.
* Tiene una curva de aprendizaje empinada, ya que utiliza un enfoque y una sintaxis diferentes a los de otros marcos.
* No utiliza widgets nativos. (*Codificación BREX*, s. f.)

**WxPython**

**** Figura 1.4 Logo de WxPython

De forma similar como la biblioteca PyQt permite usar widgets de Qt en Python, WxPython te permite usar widgets de una biblioteca llamada WxWidget, desarrollado con C++, en Python.

Con WxPython puedes crear aplicaciones multiplataforma (Windows, Linux, Mac) que tienen el Look and Feel de componentes nativos, y sin tener que reescribir código.

* Ventajas:
  + Completo conjunto de elementos gráficos (listados, arboles, grillas, etc.)
  + Flexible control del comportamiento de la interface
  + Rápido y de Apariencia nativa (diseñado para utilizar funciones nativas de cada plataforma)
  + "Baterías Incluidas": más de 12 librerías y utilitarios complementarios.
  + Independencia: no está orientado a ningún entorno, ni QT ni GTK, hay una capa más que agrega un grado de libertad adicional
  + No se cierra en el mínimo denominador común; soporta las características comunes de Windows, y las emula en Linux/Mac OS cuando no se pueden hacer nativamente (y viceversa).
  + Es más "pitónico", por ej. espacio de nombres más claro, sin referencias a C/C++, etc.
  + Permite separar completamente el diseño de la interface en XML del código python (XRC)
  + Es fácil armar componentes personalizados, tanto que incorpora widgets que no están en wxWidgets mismo, ya que están escritos en Python ([AGW](http://xoomer.virgilio.it/infinity77/main/freeware.html)).
  + Documentación completa y ejemplos extensivos.
  + Su lista oficial de usuarios ([wxpython-users](http://groups.google.com/group/wxpython-users)) es muy activa y amigable, donde participan los desarrolladores principales del proyecto.
* Desventajas:
  + No viene preinstalado con python, se debe instalar un paquete (wxPython en Windows y Mac OS, wxWidgets+wxPython en Linux, aunque en este último caso está generalmente está fácilmente disponible en los repositorios).
  + Relativamente más complejo de aprender
  + Al tener un desarrollo bastante rápido y sostenido, se liberan versiones frecuentemente, lo que en la práctica le confiere cierto nivel de "volatilidad" y problemas de compatibilidad si se deben mantener varias versiones de wx para el mismo código.
  + Es una capa más sobre el toolkit gráfico que se usa debajo (ej: GTK).
  + Las características emuladas de otras plataformas no siempre se ven bien.
  + Hacer interfaces multiplataformas que se vean bien requiere conocimiento del toolkit subyacente (win32, gtk).
  + En proyectos medianos/grandes, puede ser inestable y difícil de debuggear: en windows es muy fácil segfaultear si se pasan parámetros incorrectos. (Argentina, 2023)

**DearPyGUI**

**** Figura 1.5: Logo de DearPyGUI

DearPyGUI es un framework para crear desde aplicaciones de escritorios pequeñas hasta muy grandes, es Open Source, y la mayor parte del código está escrito en C++, además no usa los componentes nativos del sistema operativo, sino que están desarrollados encima de APIs de Gráficos como OpenGL.

DearPyGUI ofrece componentes o Widgets comunes como botones, ventanas y cajas de texto (Inputs), pero también componentes avanzados como gráficos estadísticos, gráficos de nodos, Canvas, renderizado 3D, y mucho más. Además de soportar ejecución asíncrona de código. Haciéndolo una de las opciones más robutas para crear aplicaciones Desktop multiplataformas y profesionales.

Además DearPyGui es un kit de herramientas de interfaz gráfica de usuario (GUI) dinámico, acelerado por GPU y fácil de usar para Python. Está "construido con" Dear ImGui. Las características incluyen elementos tradicionales de la interfaz gráfica de usuario, como botones, botones de opción, menús y varios métodos para crear un diseño funcional. Además, DPG tiene una increíble variedad de gráficos dinámicos, tablas, dibujos, depurador y múltiples recursos Espectadores.

DPG es muy adecuado para crear interfaces de usuario simples, así como para desarrollar interfaces gráficas complejas y exigentes.

DPG ofrece un marco sólido para el desarrollo de aplicaciones científicas, de ingeniería, de juegos, de ciencia de datos y de otro tipo que requieren interfaces rápidas e interactivas. (*pdf.pdf*, s. f.)

Ventajas:

* Representación de GPU: DearPyGUI dibuja widgets utilizando la GPU de la computadora, lo que resulta en una interfaz de usuario más rápida y fluida.
* Compatibilidad con las funciones asincrónicas incorporadas simples: DearPyGUI es compatible con funciones asincrónicas incorporadas simples, lo que permite a los desarrolladores crear interfaces de usuario más dinámicas.
* Control completo de tema y estilo: DearPyGUI proporciona un control completo sobre el tema y el estilo de la interfaz de usuario, lo que permite a los desarrolladores personalizar la apariencia de su aplicación.
* Ventana de registro incorporada simple: DearPyGUI proporciona una ventana de registro incorporada simple que facilita la depuración de aplicaciones.
* DearPyGUI proporciona una amplia variedad de widgets para elegir, lo que permite a los desarrolladores crear interfaces de usuario personalizadas.

Desventajas:

* No es compatible con todos los sistemas operativos: DearPyGUI no es compatible con todos los sistemas operativos, lo que puede limitar su uso en algunos casos.
* Curva de aprendizaje empinada: DearPyGUI tiene una curva de aprendizaje empinada, lo que puede hacer que sea difícil para los desarrolladores principiantes comenzar a usarlo.
* Documentación Limitada: La Documentación de DearPyGUI es limitada en comparación con otros marcos de interfaz de usuario de Python, lo que puede dificultar la solución de problemas y la creación de aplicaciones.

**Flet**

**** Figura 1.6: Logo de Flet

Flet es un framework relativamente nuevo, permite crear aplicaciones web y desktop usando Python junto a interface de usuario llamado Flet Controls que están desarrolladas en Flutter (un framework de interfaces desarrollado en Dart, es decir al igual WxPython o PyQt, te permite manipular widgets de otro lenguaje pero en Python.

En la práctica usar esta biblioteca se le parece a usar tkinter, debido a su facilidad de instalación y uso. Su documentación aún es pequeña, pero tiene ejemplos muy sencillos, además que es más fácil de entender que los de los frameworks anteriores.

El framework ya posee módulos de autenticación, animaciones, almacenaje local (sesiones y memoria), Hot Reload, además de otras características.

Y para ir a la producción te da la posibilidad de convertir tu aplicación a una app de escritorio con módulos como PyInstaller o desplegar tus proyectos en servicios de la nube como flu.io

**1.2: Herramientas y tecnologías de desarrollo**

Las **Herramientas**, son [programas](https://www.ecured.cu/Programas), [aplicaciones](https://www.ecured.cu/Aplicaciones) o simplemente [instrucciones](https://www.ecured.cu/index.php?title=Instrucciones&action=edit&redlink=1) usadas para efectuar otras tareas de modo más sencillo. En un sentido amplio del término, podemos decir que una herramienta es cualquier programa o instrucción que facilita una tarea.(*Herramientas informáticas - EcuRed*, s. f.) La Tecnología es el [conjunto](https://concepto.de/que-es-un-conjunto/) de nociones y [conocimientos científicos](https://concepto.de/conocimiento-cientifico/)**que el**[ser humano](https://concepto.de/ser-humano/)**utiliza para lograr un**[objetivo](https://concepto.de/objetivo/)**preciso,** que puede ser la solución de un [problema](https://concepto.de/problema/) especifico del individuo o la satisfacción de alguna de sus necesidades.(*Tecnología - Concepto, tipos, ejemplos, evolución, características*, s. f.)

**Canva**

Canva es una herramienta relacionada con la transmisión, procesamiento y almacenamiento digitalizado de la información. Un aliado del emprendimiento, tanto en nuevos conceptos como en lo tradicional. Es realmente espectacular para diseñar y crear contenido web de todo tipo. Con Canva podemos crear carteles, posters, infografías, documentos, tarjetas de visita, gráficos, etc. de forma muy sencilla. (Sanchez Chavez, 2020).Se utilizó para diseño de prototipos de interfaces de la herramienta informática.

**PyQt**

[PyQt en su versión 6](https://www.pythonguis.com/pyqt6-tutorial/) es la unión de la biblioteca gráfica de Qt y el lenguaje de programación Python, en otras palabras es un conjunto de herramientas que permiten que se trabaje el diseño gráfico en Qt y se ensamble con la programación del lenguaje Python, permitiendo que se ejecuten en la mayoría de las plataformas. (Rojas Desales, 2022) Se utilizó para el diseño de la interfaz de la herramienta informática.

**SQLite**

SQLite es una librería compacta y autocontenida de código abierto y distribuida bajo dominio público que implementa un gestor de bases de datos SQL embebido, sin configuración y transaccional. Los usuarios más conocidos que la utilizan actualmente en sus aplicaciones son: Adobe, Apple, Mozilla, Google, McAfee, Microsoft, Philips, Sun y Toshiba, entre otros.

Se empleó este lenguaje de base de datos porque es fácil de usar, ya que no se utiliza una comunicación cliente-servidor para las consultas, ya que se comunica con un archivo que es la base de datos y que puede ser autogenerado por la propia aplicación. Es muy ligero, integrándose con la propia aplicación.

**Python**

Python en su versión 3.12.0 es un lenguaje de programación interpretado y multiplataforma, de propósito general, que ha existido desde hace un tiempo. Un lenguaje interpretado, esto se refiere a que el programa se interpreta por el código que lo conforma. Además de usarse para el desarrollo Web, también es útil para desarrollar Interfaces de escritorio y crear videojuegos.

Guido van Rossum, es el creador de Python, lo comenzó a desarrollar en 1990, por lo que es un lenguaje estable y maduro actualmente. Python también es considerado como un lenguaje dinámico, esto significa que la compilación se realiza en tiempo de ejecución. Es orientado a objetos, significa que se ejecuta en las principales plataformas de Hardware y de los distintos sistemas operativos, por lo cual no está restringido. Programar con Python es muy productivo, ya que en cada fase del desarrollo ofrece un análisis, diseño, creación, ajustes, documentación, implementación y mantenimiento (*Python in a Nutshell - Alex Martelli, Anna Martelli Ravenscroft, Steve Holden, Paul McGuire - Google Libros*, s. f.)

En resumen, Python tiene las siguientes características:

* De alto nivel, esto indica que leer y escribir en Python es sencillo
* Interpretado, es que no necesita un compilador para escribir y ejecutar Python, lo cual permite
* Ahorrar tiempo en desarrollo y no es necesario compilar y enlazar.
* Orientado a Objetos, esto permite a los usuarios manipular estructuras denominadas objetos para construir y ejecutar los programas.
* Python ofrece mucho mayor prueba de errores comparado con el lenguaje C.
* Python tiene incorporado arreglos de tamaño flexible y diccionarios.
* Permite trabajar los programas por módulos, estos mismos se pueden reutilizar en otros programas. (Rojas Desales, 2022)

Se escogió este lenguaje de programación ya posee una sintaxis clara y legible, que es fácil de entender, facilita la lectura y la escritura del código, lo acelera el desarrollo; además de poseer una amplia biblioteca estándar extensa que cubre diversas áreas.

**Visual Studio Code**

Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma, está disponible para Windows, GNU/Linux y macOS. VS Code tiene una buena integración con Git, cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación. (*Qué es Visual Studio Code y qué ventajas ofrece*, 2022)

Se escogió este EDI ya que es el más usado para desarrollar aplicaciones en cualquier lenguaje, es bastante completo, además de las extensiones que tienen que la ayudan mucho a la otra de codificar

# **Epígrafe 1.3: Metodologías de Desarrollo Programación Extrema (XP)**

La programación extrema es una metodología de desarrollo ligera (o ágil) basada en una serie de valores y de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas. Este modelo de programación se basa en una serie de metodologías de desarrollo de software en la que se da prioridad a los trabajos que dan un resultado directo y que reducen la burocracia que hay alrededor de la programación. Una de las características principales de este método de programación, es que sus ingredientes son conocidos desde el principio de la informática. Los autores de XP han seleccionado aquellos que han considerado mejores y han profundizado en sus relaciones y en cómo se refuerzan los unos con los otros. El resultado de esta selección ha sido esta metodología única y compacta. Por esto, aunque no está basada en principios nuevos, sí que el resultado es una nueva manera de ver el desarrollo de software. El objetivo que se perseguía en el momento de crear esta metodología era la búsqueda de un método que hiciera que los desarrollos fueran más sencillos. Aplicando el sentido común.(*Programación Extrema - PDF Descargar libre*, s. f.)

Para desarrollar un proyecto de software, la metodología propone cuatro fases:

**Planeación**: La actividad de planeación comienza escuchando —actividad para recabar requerimientos que permite que los miembros técnicos del equipo XP entiendan el contexto del negocio para el software y adquieran la sensibilidad de la salida y características principales y funcionalidad que se requieren—. Escuchar lleva a la creación de algunas historias del usuario que describen la salida necesaria, características y funcionalidad del software que se va a elaborar. Cada historia es escrita por el cliente y colocada en una tarjeta indizada. El cliente asigna un valor (es decir, una prioridad) a la historia con base en el valor general de la característica o función para el negocio.

**Diseño**: El diseño XP sigue rigurosamente el principio MS (mantenlo sencillo). Un diseño sencillo siempre se prefiere sobre una representación más compleja. Además, el diseño guía la implementación de una historia conforme se escribe: nada más y nada menos. Se desalienta el diseño de funcionalidad adicional porque el desarrollador supone que se requerirá después. XP estimula el uso de las tarjetas CRC como un mecanismo eficaz para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC (clase-responsabilidad-colaborador) identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual de software.

**Codificación**: Después de que las historias han sido desarrolladas y de que se ha hecho el trabajo de diseño preliminar, el equipo no inicia la codificación, sino que desarrolla una serie de pruebas unitarias a cada una de las historias que se van a incluir en la entrega en curso (incremento de software).Una vez creada la prueba unitaria, el desarrollador está mejor capacitado para centrarse en lo que debe implementarse para pasar la prueba. No se agrega nada extraño (MS). Una vez que el código está terminado, se le aplica de inmediato una prueba unitaria, con lo que se obtiene retroalimentación instantánea para los desarrolladores. Un concepto clave durante la actividad de codificación (y uno de los aspectos del que más se habla en la XP) es la programación por parejas. XP recomienda que dos personas trabajen juntas en una estación de trabajo con el objeto de crear código para una historia. Esto da un mecanismo para la solución de problemas en tiempo real (es frecuente que dos cabezas piensen más que una) y para

el aseguramiento de la calidad también en tiempo real (el código se revisa conforme se crea). También mantiene a los desarrolladores centrados en el problema de que se trate.

**Pruebas**: Ya se dijo que la creación de pruebas unitarias antes de que comience la codificación es un elemento clave del enfoque de XP. Las pruebas unitarias que se crean deben implementarse con el uso de una estructura que permita automatizarlas (de modo que puedan ejecutarse en repetidas veces y con facilidad). Esto estimula una estrategia de pruebas de regresión, siempre que se modifique el código (lo que ocurre con frecuencia, dada la filosofía del rediseño en XP). A medida que se organizan las pruebas unitarias individuales en un “grupo de prueba universal], las pruebas de la integración y validación del sistema pueden efectuarse a diario. Esto da al equipo XP una indicación continua del avance y también lanza señales de alerta si las cosas marchan mal. Las pruebas de aceptación XP, también llamadas pruebas del cliente, son especificadas por el cliente y se centran en las características y funcionalidad generales del sistema que son visibles y revisables por parte del cliente. Las pruebas de aceptación se derivan de las historias de los usuarios que se han implementado como parte de la liberación del software ROGER, S. P. (2012).

**Capítulo 2. Planificación y Diseño**

En este capítulo se documentan los aspectos más relevantes que fueron identificados durante el estudio del contexto del software y el sistema a partir de las necesidades encontradas. Se detallan los requisitos del sistema mediante las Historias de Usuario, así como aspectos relacionados con el diseño de software, tales como: diagrama de clases y diagrama de entidad- relación.

**2.1: Actores del Sistema**

Los Actores del Sistema son los antiguos trabajadores del negocio; así como los actores del negocio que interactúen con el sistema.

Tabla 2.1: Actores del Sistema

|  |  |
| --- | --- |
| Actor | Justificación |
| Usuario | Podrá realizar las operaciones comunes para lo que fue diseñada la herramienta, dichas operaciones se realizan desde la interfaz gráfica de la misma. |
| Administrador | Es el encargado de crear Usuarios, tendrá acceso a revisar las trazas del Sistema y podrá realizar cambios en el Sistema |

# **2.2: Funcionalidades del Sistema**

Las funcionalidades del sistema se refieren al conjunto de funciones que satisfagan las necesidades implícitas o explícitas de los usuarios, al ser utilizado bajo condiciones específicas. Se relaciona directamente con aquello que el software hace para satisfacer necesidades, mientras que las demás características se refieren al cómo y al cuándo. Evalúa el cumplimiento de requerimientos, la exactitud de los resultados, la seguridad del producto y la interacción con otros sistemas. («Funcionalidad», 2015)

# **Requisitos Funcionales**:

Requisitos Funcionales (Capacidades): Describe las funciones que lleva a cabo el software; como debe reaccionar éste ante ciertas entradas y como debe comportarse en situaciones particulares.

F1: Autenticarse: Permitirá al usuario acceder en el Sistema

Entrada: Usuario y Contraseña

Salida: El usuario accede al sistema

F2: Gestionar Usuario: El administrador creará, editará y eliminará usuarios del sistema

Entrada: Datos

Salida: Usuarios Creados, Editados o Eliminados

F3: Registrar Cálculo de Transporte Longitudinal de Arena: Tanto el administrador como el usuario crearán, editarán y eliminaran cálculos

Entrada: Datos

Salida: Registro de Cálculos

F4: Gestionar Trazas: El administrador podrá visualizar y eliminar las Trazas.

**Requisitos No Funcionales:**

Requisitos  No Funcionales (Restricciones, requisitos de calidad): Restricciones sobre las funciones o servicios ofrecidos por el sistema.

* **Requisito de interfaz**

El sistema debe tener una interfaz sencilla y simple de usar.

* **Requerimiento de Usabilidad**

El sistema deber ser fácil de usar y comprender. El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.

* **Requerimiento de Seguridad**

El sistema debe estar protegido contra el acceso no autorizado. Dará acceso a la información a los usuarios autorizados sin ocultar o retrasar a los usuarios

* **Requerimiento de Fiabilidad**

El sistema debe ser confiable y cumplir con los requisitos del usuario

* **Requerimiento de Mantenimiento**

El sistema debe ser fácil de mantener y actualizar.

**2.3 Historias de Usuarios:**

Las Historias de Usuario representan una breve descripción del comportamiento del sistema, emplea terminología del cliente sin lenguaje técnico, se realiza una por cada característica principal del sistema, se emplean para hacer estimaciones de tiempo y para el plan de lanzamientos, reemplazan un gran documento de requisitos y preceden la creación de las pruebas de aceptación.

* **HU1: Iniciar Sesión**
* **HU2: Cálculo de Variación de Costa**
* **HU3: Nuevo Cálculo**
* **HU4: Editar Cálculo**
* **HU5: Gestión de Trazas**
* **HU6: Gestión de Usuarios**
* **HU7: Nuevo Usuario**
* **HU8: Editar Usuario**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Historia de Usuario | | |
| Número: 1 | | **Nombre de la Historia de Usuario: Iniciar Sesión** | |
|  | **Modificación o Extensión de Historia de Usuario y Número: Ninguna** | | |
| Usuario: Administrador y Cliente | | | **Iteración asignada:** |
| Prioridad del Negocio: (Alta/Media/ Baja)  Alta | | | **Puntos Estimados:** 1 semana |
| Riesgo en Desarrollo: (Alta/Media/ Baja)  Baja | | | **Puntos Reales:** 1 semana |
|  | **Programador responsable:** Dariel Cabrera | | |
|  | **Descripción:** El módulo de autenticación debe mostrar un formulario que está compuesto por dos campos obligatorios, usuario y contraseña. Cuando el usuario se autentique hará el rol de administrador o de cliente. | | |
|  | **Observaciones:** En caso que los datos de entradas no estén correctos, se lanza un mensaje alertando que las credenciales son incorrectas. | | |
|  |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Historia de Usuario | | |
| Número: 2 | | **Nombre de la Historia de Usuario: Cálculo de Variación de Costa** | |
|  | **Modificación o Extensión de Historia de Usuario y Número: Ninguna** | | |
| Usuario: Administrador y Cliente | | | **Iteración asignada:** |
| Prioridad del Negocio: (Alta/Media/ Baja)  Alta | | | **Puntos Estimados:** 2 semana |
| Riesgo en Desarrollo: (Alta/Media/ Baja)  Baja | | | **Puntos Reales:** |
|  | **Programador responsable:** Dariel Cabrera | | |
|  | **Descripción:** El módulo de Cálculo de Variación de Costa tendrá campos obligatorios para que el usuario realice la entrada de datos | | |
|  | **Observaciones:** En caso que los datos de entradas no estén correctos o no entre unos de los datos solicitados, se lanza un mensaje alertando que las credenciales son incorrectas. | | |
|  |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Historia de Usuario | | |
| Número: 3 | | **Nombre de la Historia de Usuario: Gestión de Trazas** | |
|  | **Modificación o Extensión de Historia de Usuario y Número: Ninguna** | | |
| Usuario: Administrador | | | **Iteración asignada:** ,,, |
| Prioridad del Negocio: (Alta/Media/ Baja)  Alta | | | **Puntos Estimados:** 1 semana |
| Riesgo en Desarrollo: (Alta/Media/ Baja)  Baja | | | **Puntos Reales:** 1 semana |
|  | **Programador responsable:** Dariel Cabrera | | |
|  | **Descripción:** El módulo de gestión de Trazas solo tendrá acceso el administrador el cual podrá ver y eliminar las trazas del sistema | | |
|  | **Observaciones** | | |
|  |  | | |

**2.4 Arquitectura del Sistema**

Diagrama de Entidad-Relación

Un diagrama de entidad-relación (ER) es una representación gráfica utilizada en el diseño de bases de datos. Este tipo de diagrama es una herramienta fundamental en el modelado de datos, ya que permite visualizar las entidades relevantes en un sistema, sus atributos y las relaciones entre ellas.

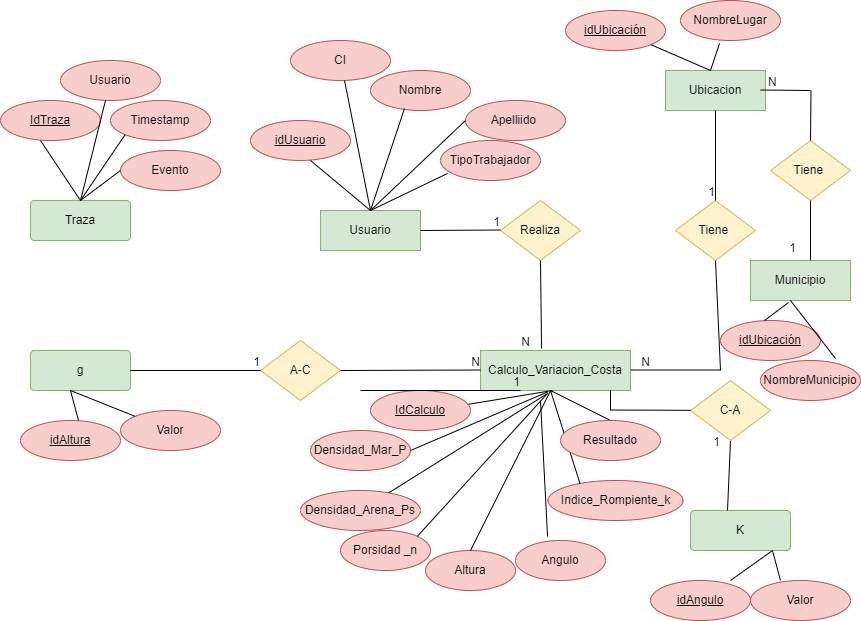
****

Figura 2.1 La Entidad Usuario va contener los datos del trabajador que accede al sistema, Este usuario hace muchos Cálculos (Es otra entidad), Calculos\_Variacion\_Costa va a contener todas las variables que no varíen durante un periodo corto de tiempo. La entidad K y la entidad g se van guardar los datos de estas mediciones (Son variables empíricas que puedes ser modificadas) estas dos entidades tienen muchos cálculos. La entidad Municipio va a contener muchas Ubicaciones, asi la entidad ubicación que va a contener el nombre del lugar donde se van a realizar las mediciones,. En una ubicación se realizan muchos Calculos\_Variacion\_Costa.

**Diagrama de Clases**

[Un **diagrama de clases** es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos](https://es.wikipedia.org/wiki/Diagrama_de_clases)

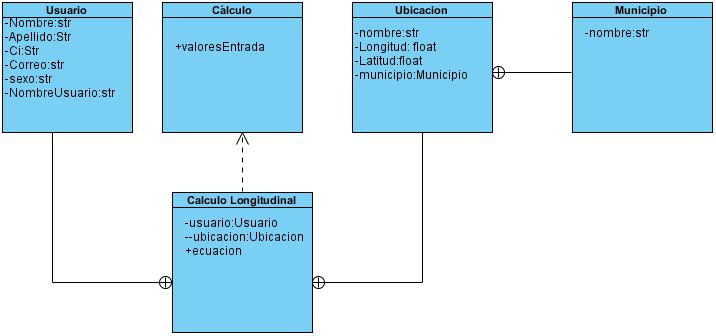
****

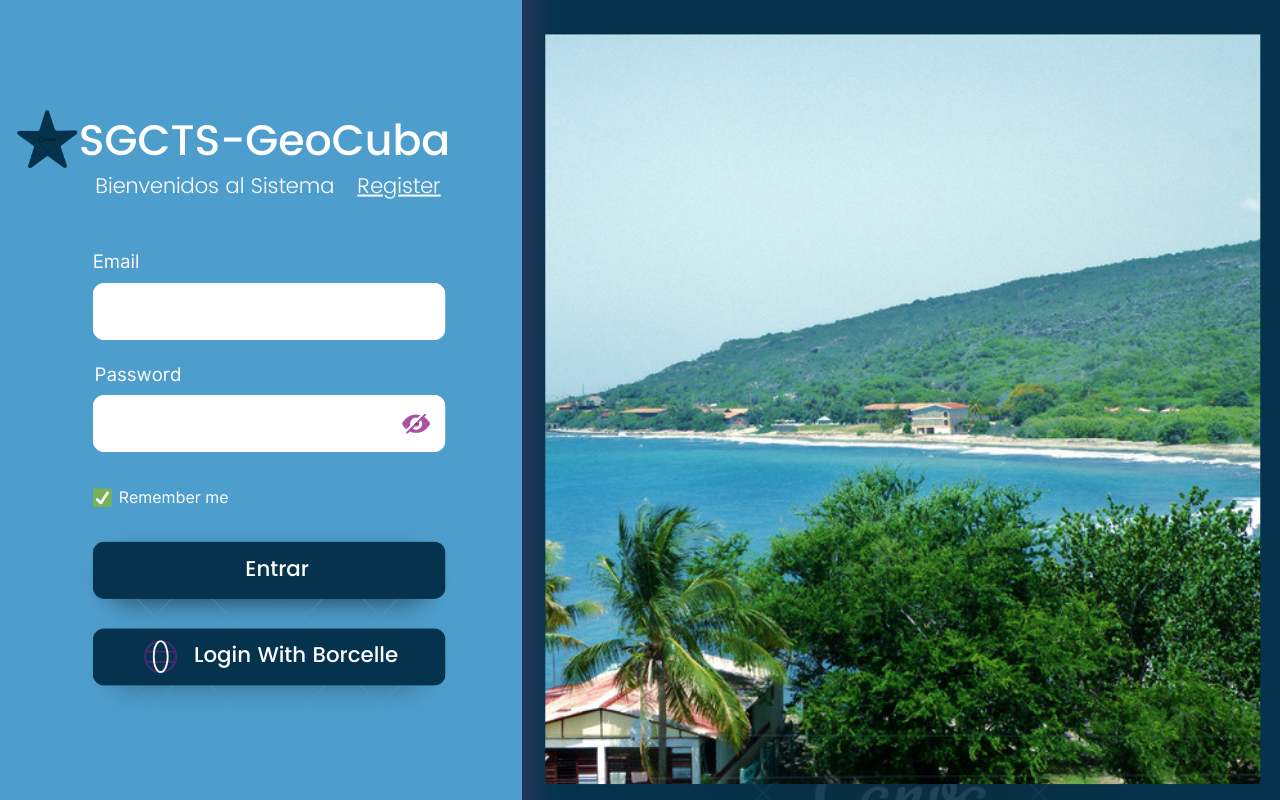
Figura 2.2 La clase usuario va a realizar muchos cálculos longitudinales, los cuales heredan de la clase Cálculos, estos cálculos ser realizan en una ubicación, en una ubicación se realiza muchos cálculos, en un Municipio hay muchas ubicaciones

**2.5: Prototipos de Interfaz**

Prototipo de interfaz usuario: Presentación de la interfaz del producto que representa la funcionalidad contenida en los casos de uso; de manera que permita que el usuario verifique que el sistema va a satisfacer sus necesidades.



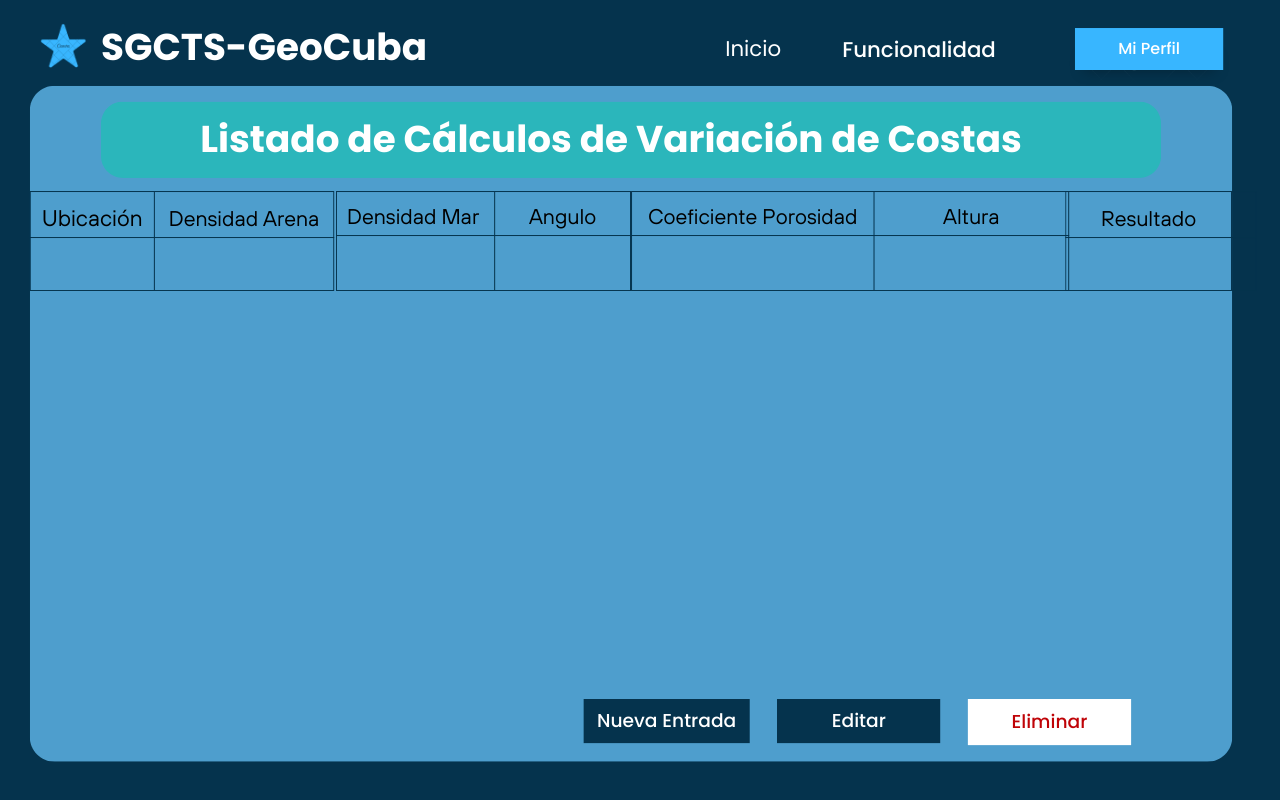
**Figura 2.3** **Interfaz Previa**



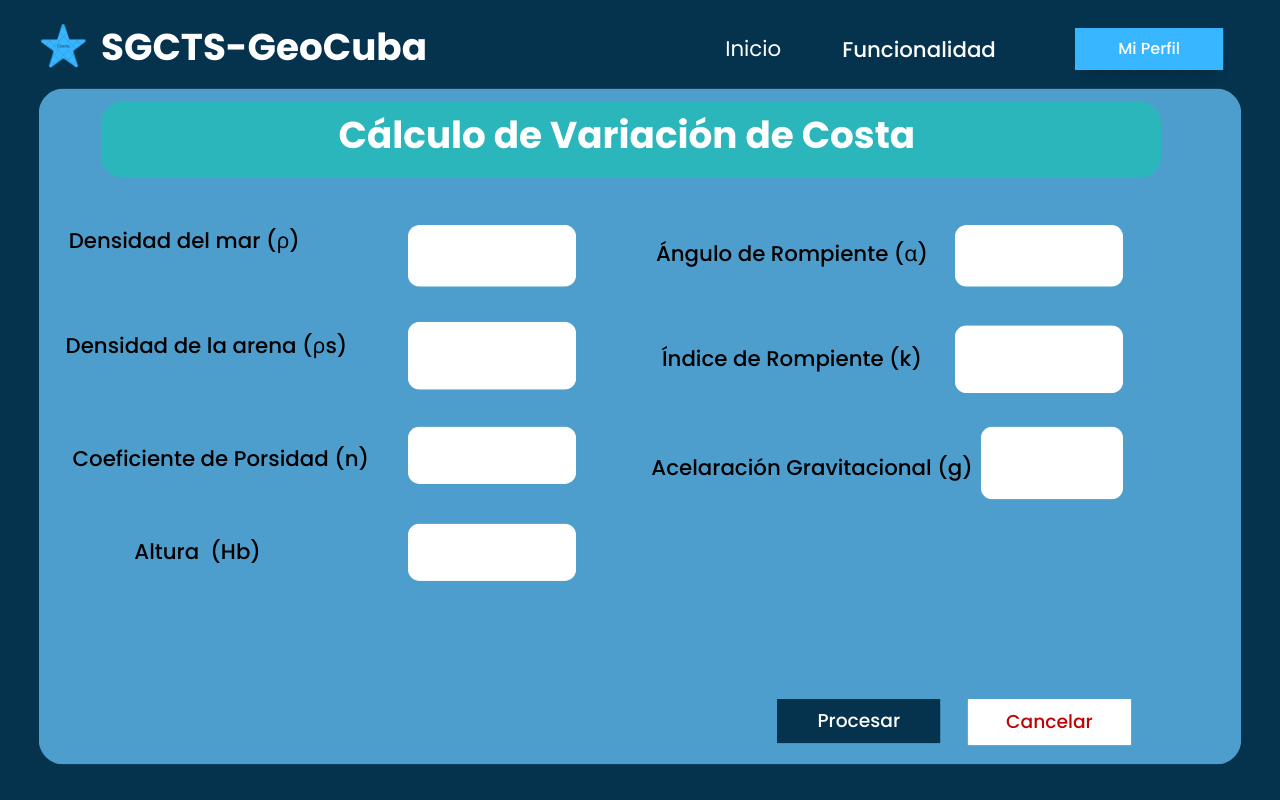
**Figura 2.4 Interfaz de Autenticación**

****

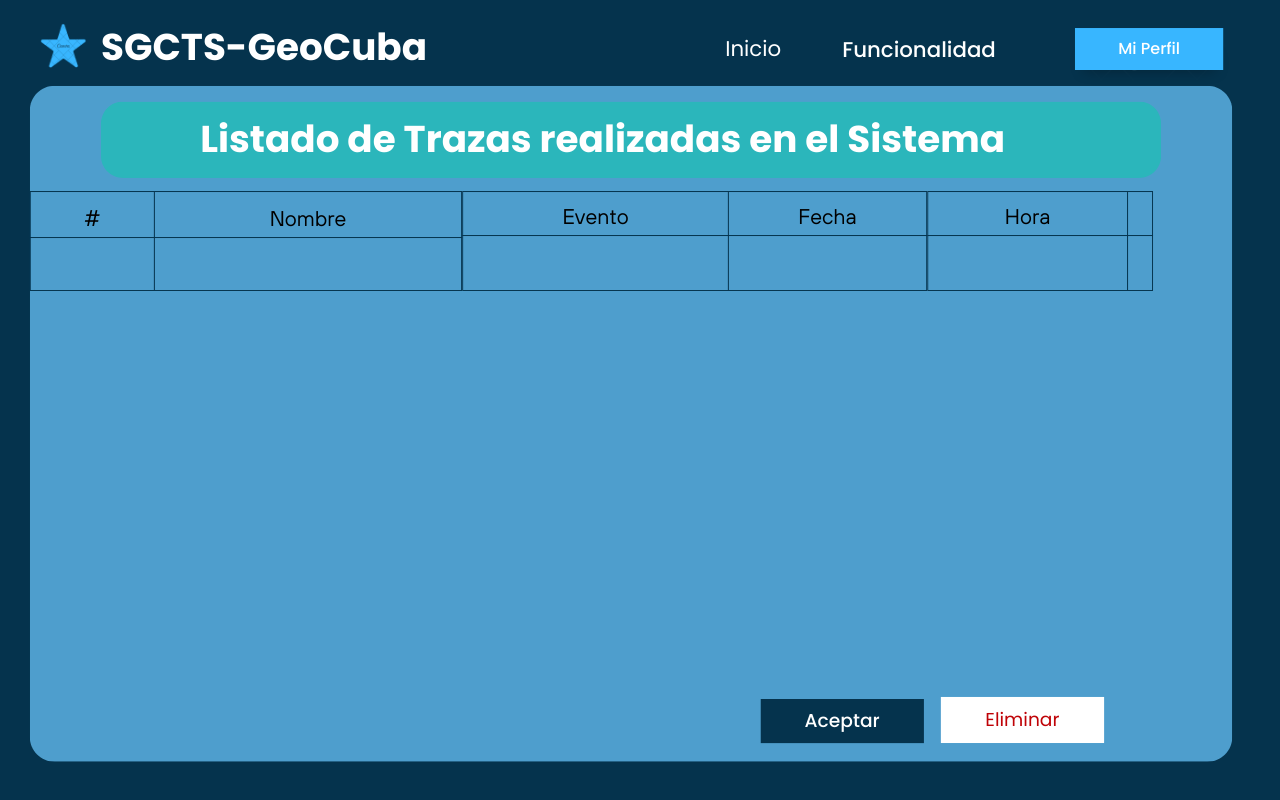
**Figura 2.5 Interfaz de Inicio**, Contiene una Botón Funcionalidades que permitirá acceder a las demás opciones del sistema (Listado de Cálculo de Variación de Costa, Listado de Trazas (solo el administrador podrá acceder) el botón mi perfil permite al usuario ver los datos de su perfil podrá revisarlos, editarlo y cerrar sección



**Figura 2.6 Interfaz Listado de Cálculos de Variación de Costa.** El usuario podrá ver los datos almacenados en la Base de Datos, podrá editarlos, eliminarlos y el botón Nueva Entrada lo llevará a la interfaz **Cálculo de Variación de Costa**. Con el botón inicio podrá regresar a la **interfaz Inicio**



**Figura 2.7 Interfaz Cálculo de Variación de Costa** el usuario podrá entrar los datos para el cálculo en cuestión. A través del botón procesar el sistema le debe realizar los cálculos y guardarlos en la Base de Datos. Con el botón Cancelar podrá regresar a la interfaz anterior. Con el botón inicio podrá regresar a la **interfaz Inicio**



**Figura 2.8 Interfaz de Trazas** solo el administrador del sistema podrá acceder a esta donde podrá ver las trazas de los usuarios del sistema y podrá eliminar. Con el botón inicio podrá regresar a la interfaz Inicio

**Conclusiones**

* Se realizó un estudio del estado del arte de los marcos para crear aplicaciones de escritorio, lo que permitió usar el más apto y óptimo para realizar el trabajo de investigación, el framework PyQt fue el que resultó elegido.
* Se diseñó el prototipo de la herramienta informática que permite el cálculo, almacenamiento y búsqueda de las mediciones realizadas en las líneas costeras.
* Se definieron las herramientas de desarrollo y se implementó la herramienta diseñada.

**Anexos**

Anexo 1:

Según en el Segundo Periodo Ordinario de Sesiones de la IX Legislatura de la Asamblea Nacional del 2018 se estimó que e**l 85% de las playas arenosas cubanas presentan indicios de erosión: 23 con erosión intensa y 195 con erosión moderada** (*Cubadebate*, s. f.-a)**. De 505 playas, se han evaluado 247 y se ha comprobado que 208 (84.2 %) presentan indicios de erosión.** “Se ratifica que el ritmo de la erosión de las playas arenosas en el archipiélago cubano, que se estima en 1.2 metros de retroceso de la línea de la costa como promedio cada año”, así expuso la viceministra del CITMA Adianez Taboada Zamora, en la sesión de trabajo de la Comisión de Educación, Cultura, Ciencia, Tecnología y Medio Ambiente de 2023 (*Cubadebate*, s. f.-b).Para enfrentar estos efectos la isla aprobó en 2017

Anexo 2:

En cuanto al deterioro de las costas, Tarea Vida tiene como objetivo detener el deterioro, rehabilitar y conservar los arrecifes de coral en todo el archipiélago, con prioridad en las crestas que bordean la plataforma insular y protegen playas urbanizadas de uso turístico. Además de esto el gobierno cubano aprobó el Decreto-Ley 77 de las Costas y su Reglamento, este posee elementos nuevos que ha aportado la ciencia así como el necesario reconocimiento a la protección de los asentamientos costeros, donde inciden factores de índole cultural.[4]

Anexo 3:

[GEOCUBA es un grupo empresarial cubano que se dedica a la elaboración y comercialización de información, tecnologías, productos y servicios en las esferas de la Geodesia, la Fotogrametría, la Teledetección, la Hidrografía, los Estudios Marinos, la Cartografía, los Estudios Medioambientales, las Artes Gráficas y la Ayuda a la Navegación Marítima](https://www.ecured.cu/GEOCUBA)(*GEOCUBA - EcuRed*, s. f.)

**Referencias Bibliográficas**

Argentina, P. (2023, noviembre 23). *Interfaces Gráficas (GUI)*. Python Argentina Wiki. https://wiki.python.org.ar/interfacesgraficas/

*Codificación BREX*. (s. f.). Recuperado 7 de enero de 2024, de https://www.brexcoding.com/kivy-vs-tkinter-pros-and-cons/

*Cubadebate*. (s. f.-a). Recuperado 7 de enero de 2024, de http://www.cubadebate.cu/noticias/2018/12/16/tarea-vida-a-debate-en-la-asamblea-nacional-cuba-ante-las-realidades-del-cambio-climatico/

*Cubadebate*. (s. f.-b). Recuperado 7 de enero de 2024, de http://www.cubadebate.cu/noticias/2023/07/18/tarea-vida-en-la-mira-de-los-diputados-hay-una-transicion-en-el-clima-de-cuba/

*Frameworks de Python para Desktop | Fazt Web*. (2022, junio 11). faztweb.com. https://faztweb.com/contenido/python-frameworks-desktop

Funcionalidad. (2015, junio 7). *gestion de la calidad del software - norma iso-9126*. https://diplomadogestioncalidadsoftware2015.wordpress.com/norma-iso-9126/calidad-interna-y-externa/funcionalidad/

*GEOCUBA - EcuRed*. (s. f.). Recuperado 9 de enero de 2024, de https://www.ecured.cu/GEOCUBA

*Herramientas informáticas—EcuRed*. (s. f.). Recuperado 7 de enero de 2024, de https://www.ecured.cu/Herramientas\_inform%C3%A1ticas

Medina, N. M. C. (s. f.). *EL CAMBIO CLIMÁTICO Y SUS EFECTOS SOBRE LA SALUD HUMANA*.

*Pdf.pdf*. (s. f.). Recuperado 8 de enero de 2024, de https://dearpygui.readthedocs.io/\_/downloads/en/latest/pdf/

*Programación Extrema—PDF Descargar libre*. (s. f.). Recuperado 7 de enero de 2024, de https://docplayer.es/1647643-Programacion-extrema.html

*Pros and Cons of Kivy 2024*. (2023, agosto 29). https://www.trustradius.com/products/kivy/reviews?qs=pros-and-cons

*Python in a Nutshell—Alex Martelli, Anna Martelli Ravenscroft, Steve Holden, Paul McGuire—Google Libros*. (s. f.). Recuperado 7 de enero de 2024, de https://books.google.com.ar/books?hl=es&lr=&id=2WSmEAAAQBAJ&oi=fnd&pg=PT28&dq=A.+Martelli,+A.+Ravenscroft,+and+S.Holden.+Python+in+a+Nutshell:+A+Desktop+Quick+Reference.+O%E2%80%99+Reilly+Media,2017&ots=oVi\_FYTB50&sig=81PZeDg8DjpHozbBXUNGSr2ijgU#v=onepage&q&f=false

*Qué es Visual Studio Code y qué ventajas ofrece*. (2022, julio 22). OpenWebinars.net. https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/

Rojas Desales, L. E. (2022). *DESARROLLO DE APLICACIONES DE ESCRITORIO CON PYTHON Y LA BIBLIOTECA QT*. http://ri.uaemex.mx/handle/20.500.11799/112894

Sanchez Chavez, M. Y. (2020). *Herramienta Canva para mejorar la creatividad en estudiantes de primer año en informática en la I.E. Simón Bolívar*. https://repositorio.usil.edu.pe/entities/publication/5078d6f8-4d42-4438-af21-4dffffe48496

*Tecnología—Concepto, tipos, ejemplos, evolución, características*. (s. f.). Recuperado 7 de enero de 2024, de https://concepto.de/tecnologia/#ixzz8O8i9AU6q