



Universidad Nacional Autónoma de
México



Facultad de Ingeniería

Estructuras de Datos y Algoritmos I

Proyecto Final

Sistema de Cobranza en Python

Elaborado por: Arciga Guzmán Fernando.
Valenzuela Vigil Ángel David.

Profesor: Marco Antonio Martínez Quintana.

Fecha de Entrega: 27/05/2020

Semestre: 2°

Calificación:

Objetivo del proyecto:

Crear un programa utilizando lenguaje de programación Python combinado con SQLite que permita llevar un registro de pagos y gestionar elementos como usuarios, contraseñas, productos y precios.

Alcance del proyecto:

Este proyecto busca crear una herramienta de uso común que sea intuitiva para la zona metropolitana de la Ciudad de México. Su interfaz gráfica permite que sea amigable con el usuario y el sistema de bases de datos ofrece una alternativa más segura para guardar esta información de manera local, ya que se requiere un nombre de usuario y contraseña determinados para poder acceder a los datos. Este programa incluye versiones de uso libre GNU. Siendo un código abierto, cualquier persona puede modificar el programa.

Introducción:

En México, no existe una implementación real de un sistema que permita llevar un registro de datos de forma local y segura. El pequeño empresario debe acudir a software para industrias que puede resultar abrumador, por lo que este proyecto busca crear una herramienta intuitiva y mejorable, destinada a aquellos emprendedores o *freelancers* que desean gestionar los datos de cobranza de su empresa o negocio.

Desarrollo:

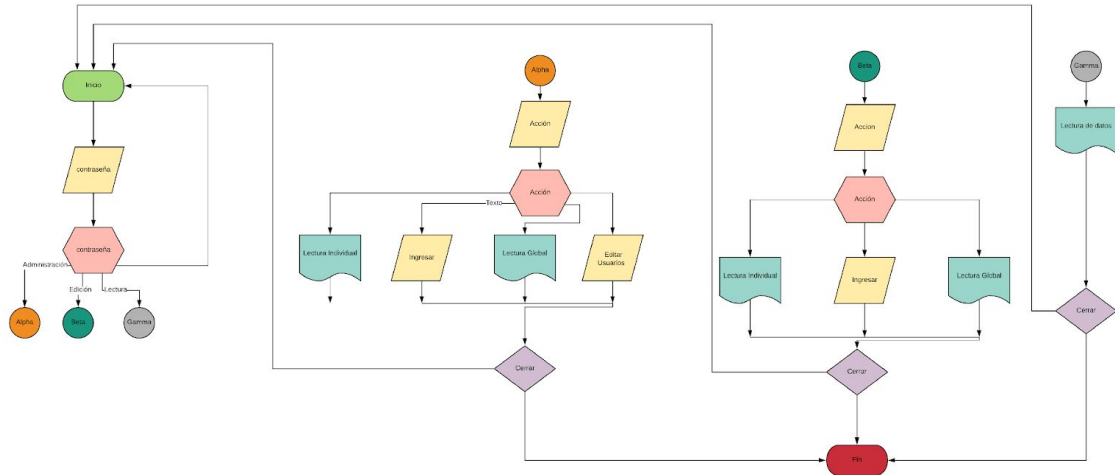
Algoritmo

1. Inicio
2. Preguntar por el tipo de usuario que usará el sistema
 - 2.1. Si usuario == Admin, entonces
 - 2.1.1. Comprobar la contraseña
 - 2.1.1.1. Si desea modificar el programa
 - 2.1.1.1.1. Si desea cambiar los usuarios
 - 2.1.1.1.1.1. Cambiar usuarios que lo utilizarán y sus respectivas contraseñas y privilegios (máximo de 3 usuarios en total)
 - 2.1.1.1.2. Si desea cambiar la estructura del programa

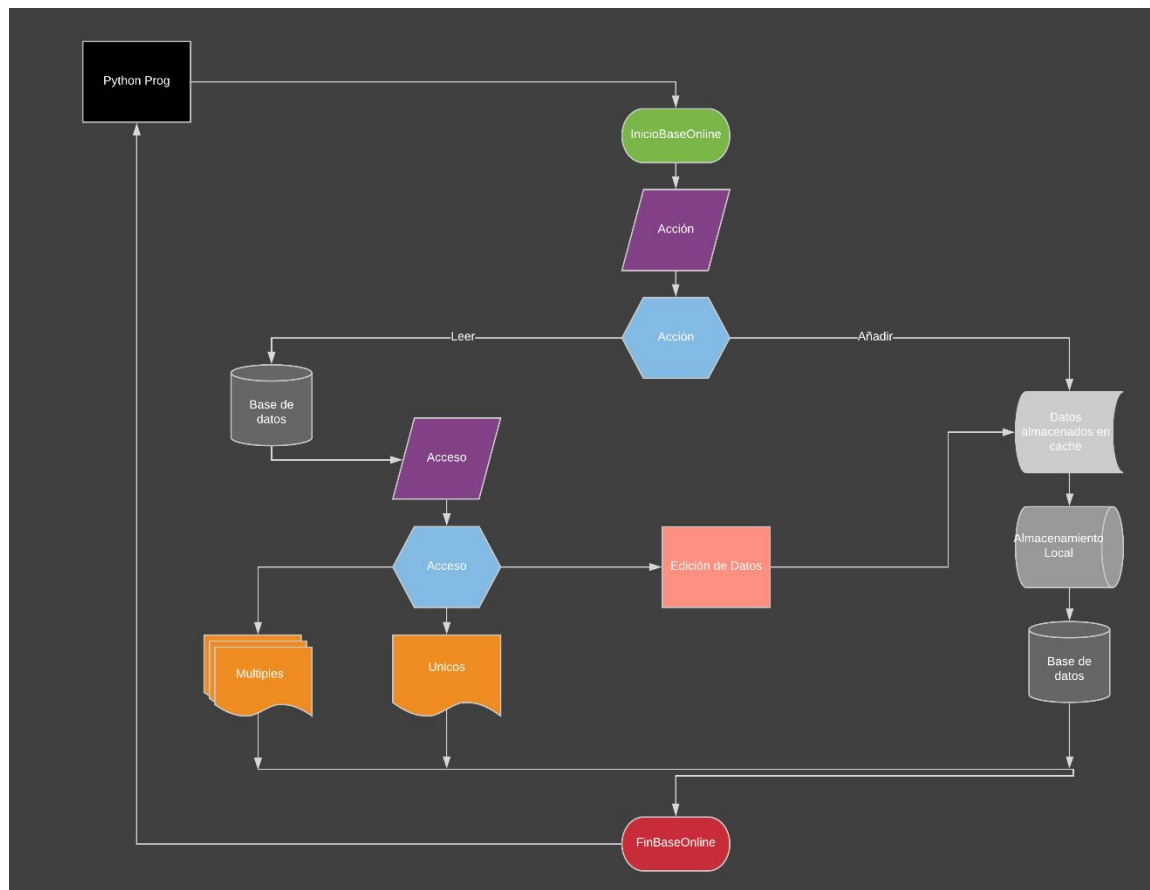
- 2.1.1.1.2.1. Definir la forma que tendrá el programa (clientes/productos)
 - 2.1.1.1.3. Crear archivos residuales con los usuarios y los productos
 - 2.1.1.1.4. Mostrar el resultado
 - 2.1.1.2. Si desea leer o editar los (productos/clientes)
 - 2.1.1.2.1. Mostrar un diccionario con formato en modo escritura
 - 2.1.1.2.1.1. Si edita datos
 - 2.1.1.2.1.1.1. Mostrar el menú
 - 2.1.1.2.1.1.1.1. Eliminar
 - 2.1.1.2.1.1.1.2. Editar
 - 2.1.1.2.1.1.1.3. Guardar
 - 2.1.1.3. Si desea salir
 - 2.1.1.3.1. Fin
- 2.2. Si usuario == Complementario, entonces
 - 2.2.1. Comprobar la contraseña
 - 2.2.1.1. Si desea leer o editar los (productos/clientes)
 - 2.2.1.1.1. Mostrar un diccionario con formato en modo escritura
 - 2.2.1.1.1.1. Si edita datos
 - 2.2.1.1.1.1.1. Mostrar el menú
 - 2.2.1.1.1.1.1.1. Eliminar
 - 2.2.1.1.1.1.1.2. Editar
 - 2.2.1.1.1.1.1.3. Guardar
 - 2.2.1.2. Si desea salir
 - 2.2.1.2.1. Fin
- 2.3. Si usuario == Externo, entonces
 - 2.3.1. Iniciar el programa en modo lectura
 - 2.3.1.1. Mostrar los productos y precios o clientes y montos (modo lectura)
 - 2.3.1.2. Si desea salir
 - 2.3.1.2.1. Fin
3. Fin

Diagrama de flujo reducido

- Programa Principal



- Base de datos



Pseudocódigo

- Programa Principal:

```
Funcion Alpha()
  Leer Acción;
  Switch(Acción)
    Caso 0:
      Escribir Lectura Individual;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
    Caso 1:
      Leer Ingresar;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
    Caso 2:
      Escribir Lectura Global;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
    Caso 3:
      Leer Editar Usuarios;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
  FinFuncion
```

```
Funcion Beta()
  Leer Acción;
  Switch(Acción)
    Caso 0:
      Escribir Lectura Individual;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
    Caso 1:
      Leer Ingresar;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
    Caso 2:
      Escribir Lectura Global;
      Switch(Cerrar)
        Caso 1:
          ProgramaPrincipal();
        Caso 2:
          Fin
  FinFuncion
```

```
Funcion Gamma()
  Escribir Lectura de datos;
  Switch(Cerrar)
    Caso 1:
      ProgramaPrincipal();
    Caso 2:
      Fin
  FinFuncion
```

1. Inicio
2. Leer contraseña;
3. Si contraseña == Administración, entonces Alpha();
4. Si contraseña == Edición, entonces Beta();
5. Si contraseña == Lectura, entonces Gamma();

- Base de Datos:

```
1. InicioBaseOnline
2. Leer Acción;
3. Switch(Acción)
Caso Leer:
Base de Datos();
Leer Acceso;
Switch(Acceso)
Caso 0:
Escribir Multiples;
FinBaseOnline
Caso 1:
Escribir Unicos;
FinBaseOnline
Caso 2:
Hacer Edición de Datos;
Datos almacenados en cache();
Almacenamiento Local();
Base de datos();
FinBaseOnline
Caso Añadir:
Datos almacenados en cache();
Almacenamiento Local();
Base de datos();
FinBaseOnline
```

Código

GitHub: <https://github.com/ghero56/Proyecto-Final-EDA1/tree/master/C%C3%B3digo>

Conclusiones:

El uso de un sistema de cobranza, con una base de datos en línea, crea una posibilidad enorme de guardar datos de manera más rápida y eficiente.

En México, se le ha restado importancia al uso de bases de datos de forma local y por eso sabemos que este proyecto será de gran utilidad para todo aquel que lo necesite, en caso de que no utilice software similar. El proyecto queda abierto para que cualquiera que lo desee pueda contribuir en su desarrollo, ya sea optimizando los procesos, actualizando las versiones de desarrollo (Python, SQL) o implementando nuevos sistemas.

Utilizar este programa facilita realizar tareas necesarias para gestionar de forma eficiente un negocio o empresa independientes. Buscamos facilitar esta tarea para optimizar el rendimiento de las microempresas de la ciudad y fomentar su desarrollo en un campo empresarial altamente competitivo.

Referencias:

pildorasinformaticas. (2017, 24 de enero). *Curso Python desde 0* [Lista de reproducción]. YouTube
<https://www.youtube.com/playlist?list=PLU8oAIHdN5BlvPxziopYZRd55pdqFwkeS>

codigofacilito. (2013, 15 de julio). *Curso Bases de Datos en MySQL* [Lista de reproducción]. YouTube
https://www.youtube.com/watch?v=PrS4RYiYVi4&list=PLbJUM5_cl8xM_uCOyStpOlYciDw1oCia

Python conexión con MySQL – Bytes [Video]. YouTube
<https://www.youtube.com/watch?v=LhsGA9PJtG0&t=608s>