

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Lenguajes formales y de programación

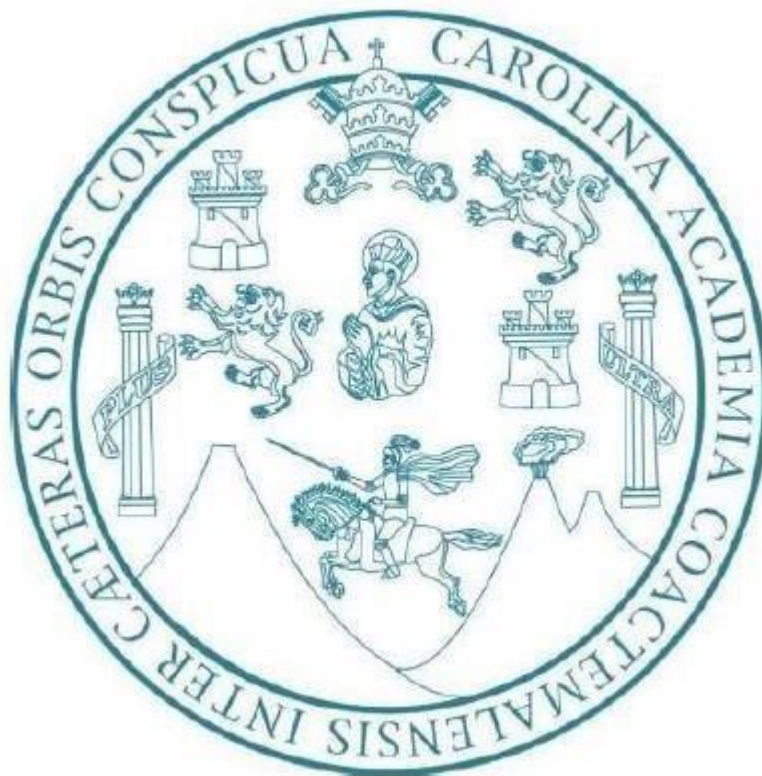
Primer semestre 2022 2021

Catedrático: Ing. Otto Amilcar Rodriguez Acosta

Auxiliar: Fernando Feliciano Chajon del Cid

Proyecto 1

Manual técnico



Nombre: Josue Daniel Solis Osorio

Carnet: 2020001574

Fecha: 20/03/2022

Introducción

El presente documento describe los aspectos técnicos informáticos del sistema de información. El documento familiariza las persona que utilicen el programa para su correcta ejecución.

Objetivo de los procedimientos

La elaboración de una aplicación que sea capaz de generar formularios a partir de una lectura de un archivo.form median el uso de autómatas

Especificaciones técnicas

El sistema operativo con el que se trabajó con Windows 10, utilizando el lenguaje de programación denominado Python

Requisitos mínimos

Windows

- Windows 10 (8u51 y superiores)
- Windows 8.x (escritorio)
- Windows 7 SP1
- Windows Vista SP2
- Windows Server 2008 R2 SP1 (64 bits)
- Windows Server 2012 y 2012 R2 (64 bits)
- RAM: 128 MB
- Espacio en disco: 124 MB para JRE; 2 MB para Java Update
- Procesador: Mínimo Pentium 2 a 266 MHz
- Exploradores: Internet Explorer 9 y superior, Firefox

Mac OS X

- Mac con Intel que ejecuta Mac OS X 10.8.3+, 10.9+
- Privilegios de administrador para la instalación
- Explorador de 64 bits
- Se requiere un explorador de 64 bits (Safari, por ejemplo) para ejecutar Oracle Java en Mac.

Linux

- Oracle Linux 5.5+¹
- Oracle Linux 6.x (32 bits), 6.x (64 bits)²
- Oracle Linux 7.x (64 bits)² (8u20 y superiores)
- Red Hat Enterprise Linux 5.5+¹ 6.x (32 bits), 6.x (64 bits)²
- Red Hat Enterprise Linux 7.x (64 bits)² (8u20 y superiores)
- Suse Linux Enterprise Server 10 SP2+, 11.x
- Suse Linux Enterprise Server 12.x (64 bits)² (8u31 y superiores)
- Ubuntu Linux 12.04 LTS, 13.x
- Ubuntu Linux 14.x (8u25 y superiores)
- Ubuntu Linux 15.04 (8u45 y superiores)
- Ubuntu Linux 15.10 (8u65 y superiores)
- Exploradores: Firefox

Lógica del programa

Clase: Programa

Menu :Esta función se encarga de generar un HTML a través de los datos cargados

```
def menu(self): ...
```

reportes: Esta función se encarga de generar el listado errores y tokens atravez de u HTML

```
def reportes(self): ...
```

Clase: analizar_form

Leer: esta función se encarga de cargar el archivo .form

```
def leer(self): ...
```

Analizador: esta función se encarga de leer archivo. Form y separarlos por sus respectivos tokens

```
def analizador(self): ...
```

Estas funciones se encargar de realizar la labor de automata para posteriormente su identificación como tokens

```
def leer_letra(self): ...  
def letras_s0(self): ...  
def quitar_primera_letra(self): ...  
def letras_s1(self): ...  
def comillas_s0(self): ...  
def comillas_s1(self): ...
```


Clase: Producto

Estas funciones se encargan de recibir y dar los atributos de la “Producto”

```
@Override
public String toString() { ...3 lines }

public int getCodigo() { ...3 lines }

public void setCodigo(int codigo) { ...3 lines }

public String getNombre() { ...3 lines }

public void setNombre(String nombre) { ...3 lines }

public String getDescripcion() { ...3 lines }

public void setDescripcion(String descripcion) { ...3 lines }

public int getCantidad() { ...3 lines }

public void setCantidad(int cantidad) { ...3 lines }

public double getPrecio() { ...3 lines }

public void setPrecio(double precio) { ...3 lines }

public int getNumVentas() { ...3 lines }

public void setNumVentas(int numVentas) { ...3 lines }
```

Clase: Sucursales

Estas funciones se encargan de recibir y dar los atributos de la “Sucursales”

```
public int getCodigo() { ...3 lines }

public void setCodigo(int codigo) { ...3 lines }

public String getNombre() { ...3 lines }

public void setNombre(String nombre) { ...3 lines }

public String getDireccion() { ...3 lines }

public void setDireccion(String direccion) { ...3 line

public String getCorreo() { ...3 lines }

public void setCorreo(String correo) { ...3 lines }

public int getTelefono() { ...3 lines }

public void setTelefono(int telefono) { ...3 lines }
```

Clase: Vendedores

Estas funciones se encargan de recibir y dar los atributos de la “Vendedores”

```
public int getCodigo() {...3 lines }

public void setCodigo(int codigo) {...3 lines }

public String getNombre() {...3 lines }

public void setNombre(String nombre) {...3 lines }

public int getCaja() {...3 lines }

public void setCaja(int caja) {...3 lines }

public int getVentas() {...3 lines }

public void setVentas(int ventas) {...3 lines }

public String getGenero() {...3 lines }

public void setGenero(String genero) {...3 lines }

public String getPassword() {...3 lines }

public void setPassword(String password) {...3 lines }
```

Clase: LogicaCliente

DescargarDatos: Esta función se encarga de descargar los datos de la clase cliente

```
public void descargarDatos(String nombre, Object guardar) {...9 lines }
```

Exportar listado Pdf: esta función se encarga de hacer el exportado a pdf

```
public void exportarListadoPdf() {...45 lines }
```

CargaMasiva: Esta función se encarga de hacer la carga masiva

```
public void cargaMasiva() {...37 lines }
```

Único: Esta función se encarga de que los códigos no se repitan

```
public boolean unico(int codigo) {...12 lines }
```

Crear: Esta función se encarga de la parte lógica del botón “crear”

```
public boolean crear(int codigo, String nombre, int nit, String correo, String genero) {...16 lines }
```

Actualizar: Esta función se encarga de la parte lógica del botón “Actualizar”

```
public boolean actualizar(int codigo, String nombre, int nit, String correo, String genero) {...22 lines }
```

GraficaGenero: Esta función se encarga de realizar la gráfica de pastel para el género

```
public JFreeChart graficaGenero() {...37 lines }
```

Clase Lógica Producto:

Top3Ventas: Esta función se encarga de realizar la parte lógica para mostrar el top 3 ventas

```
public JFreeChart top3Ventas(String Titulo) {...49 lines }
```

Descargar Datos: Esta función se encarga de descargar los datos de producto

```
public void descargarDatos(String nombre, Object guardar) {...9 lines }
```

Exportar listado Pdf: esta funcion se encarga de hacer el exportado a pdf

```
public void exportarListadoPdf() {...45 lines }
```

CargaMasiva: Esta función se encarga de hacer la carga masiva

```
public void cargaMasiva() {...37 lines }
```

Único: Esta función se encarga de que los código no se repitan

```
public boolean unico(int codigo) {...12 lines }
```

Crear: Esta función se encarga la parte lógica del botón “crear”

```
public boolean crear(int codigo, String nombre, int nit, String correo, String genero) {...16 lines }
```

Actualizar: Esta función se encarga la parte lógica del botón “Actualizar”

```
public boolean actualizar(int codigo, String nombre, int nit, String correo, String genero) {...22 lines }
```

Clase Lógica Sucursales:

Descargar Datos: Esta función se encarga de descargar los datos de Sucursales

```
public void descargarDatos(String nombre, Object guardar) {...9 lines }
```

Exportar listado Pdf: esta funcion se encarga de hacer el exportado a pdf

```
public void exportarListadoPdf() {...45 lines }
```

CargaMasiva: Esta función se encarga de hacer la carga masiva

```
public void cargaMasiva() {...37 lines }
```

Único: Esta función se encarga de que los código no se repitan

```
public boolean unico(int codigo) {...12 lines }
```

Crear: Esta función se encarga la parte lógica del botón “crear”

```
public boolean crear(int codigo, String nombre, int nit, String correo, String genero) {...16 lines }
```

Actualizar: Esta función se encarga la parte lógica del botón “Actualizar”

```
public boolean actualizar(int codigo, String nombre, int nit, String correo, String genero) {...22 lines }
```

Clase: LogicaVendedores

Clase Lógica Producto:

Top3Ventas: Esta función se encarga de realizar la parte lógica para mostrar el top 3 ventas


```
public JFreeChart top3Ventas(String Titulo) {...49 lines }
```

Descargar Datos: Esta función se encarga de descargar los datos de Vendedores

```
public void descargarDatos(String nombre, Object guardar) {...9 lines }
```

Exportar listado Pdf: esta funcion se encarga de hacer el exportado a pdf

```
public void exportarListadoPdf() {...45 lines }
```

CargaMasiva: Esta función se encarga de hacer la carga masiva

```
public void cargaMasiva() {...37 lines }
```

Único: Esta función se encarga de que los código no se repitan

```
public boolean unico(int codigo) {...12 lines }
```

Crear: Esta función se encarga la parte lógica del botón “crear”

```
public boolean crear(int codigo, String nombre, int nit, String correo, String genero) {...16 lines }
```

Actualizar: Esta función se encarga la parte lógica del botón “Actualizar”

```
public boolean actualizar(int codigo, String nombre, int nit, String correo, String genero) {...22 lines }
```

Clase VistaAdmin:

GeneralSucursal: Esta función se encarga de toda la parte visual para “Sucursales”

```
public void GenerarSucursal() {...128 lines }
```

GeneralProducto: Esta función se encarga de toda la parte visual para “producto”

```
public void GenerarProductos() {...167 lines }
```

GeneraCliente: Esta función se encarga de toda la parte visual para “Cliente”

```
public void GenerarClientes() {...178 lines }
```

GenerarVendedor: Esta función se encarga de toda la parte visual para “Vendedor”

```
public void GenerarVendedor() {...156 lines }
```

Clase: VistaAdminActualizarCliente

Crear: Esta función se encarga de toda la parte lógica para el botón actualizar en “cliente”

```
public void Crear() {...88 lines }
```

Clase: VistaAdminActualizarCliente

Crear: Esta función se encarga de toda la parte lógica para el botón crear en “cliente”

```
public void Crear() {...88 lines }
```

Clase: VistaAdminActualizarProducto

Crear: Esta función se encarga de toda la parte lógica para el botón actualizar en “producto”

```
public void Crear() {...69 lines }
```

Clase: VistaAdmincrearProducto

Crear: Esta función se encarga de toda la parte lógica para el botón crear en “producto”

```
public void Crear() {...93 lines }
```

Clase: VistaAdminActualizarSucursal

Crear: Esta función se encarga de toda la parte lógica para el botón actualizar en “Sucursal”

```
public void Crear() {...79 lines }
```

Clase: VistaAdminCrearSucursal

Crear: Esta función se encarga de toda la parte lógica para el botón crear en “Sucursal”

```
public void Crear() {...82 lines }
```

Clase: VistaAdminActualizarVendedor

Crear: Esta función se encarga de toda la parte lógica para el botón actualizar en “Vendedor”

```
public void Crear() {...101 lines }
```

Clase: VistaAdminCrearSucursal

Crear: Esta función se encarga de toda la parte lógica para el botón crear en “Sucursal”

```
public void Crear() {...108 lines }
```

Clase: VistaVendedor

GenerarNuevaVenta: Esta función se encarga de toda la parte visual de nueva venta en el modulo de vendedores

```
public void GenerarNuevaVenta() {...238 lines }
```

GenerarVentas: Esta función se encarga de toda la parte visual de Generar Ventas en el modulo de vendedores

```
public void GenerarVentas() {...130 lines }
```

