

Proyecto final - Aplicación de consola “libreta de direcciones”

PRINCIPIOS DE CONSTRUCCIÓN DE SOFTWARE(85142)

UNIVERSIDAD VERACRUZANA

**FACULTAD DE CONTADURÍA Y
ADMINISTRACIÓN**

Jesús Antonio Morales Cruz-S21017265

TRABAJO: Proyecto final - Aplicación de consola “libreta de direcciones”

PROGRAMA: Licenciatura en Ingeniería de software

EE: PRINCIPIOS DE CONSTRUCCIÓN DE
SOFTWARE(85142)

DOCENTE: VERGARA CAMACHO JOSE ANTONIO

LIS401



UNIVERSIDAD VERACRUZANA
Facultad de Contaduría y Administración

Sección A: Descripción:	3
A.1) Estado del sistema:	3
A.2) Lista de clases:	5
Sección B: URL de JavaDoc:	6
Sección C: UML/Diseño:	6
Sección D: Capturas de pantalla del sistema:	7
D.1) Evidencia 1:	7
Leer entradas del archivo de datos seguidas de Mostrar listado. El	7
D.2) Evidencia 2:	8
D.3) Evidencia 3:	8
D.4) Evidencia 4:	9

Sección A: Descripción:

El sistema consiste en una aplicación de libreta de direcciones que permite gestionar y organizar información de contactos. El sistema se compone de varias clases que trabajan juntas para lograr su funcionalidad. El sistema fue hecho en replit ya que cuento con un sistema operativo diferente.

La clase principal del sistema es AddressBookApplication, que se encarga de iniciar la aplicación. En su método main(), crea una instancia de la clase AddressBook y otra instancia de la clase Menu. A continuación, llama al método displayMenu() de Menu para mostrar el menú principal al usuario y capturar sus opciones.

La clase AddressBook representa la libreta de direcciones y contiene una lista de objetos de la clase AddressEntry, que representan las entradas de dirección individuales. Esta clase proporciona métodos para agregar, eliminar y buscar entradas de dirección. Además, ofrece la funcionalidad de ordenar las entradas alfabéticamente por apellido.

La clase AddressEntry representa una entrada de dirección en la libreta de direcciones. Contiene campos para almacenar información como el nombre, dirección, correo electrónico y número de teléfono. También tiene métodos para obtener y establecer los valores de estos campos. Cada entrada de dirección está asociada a un objeto de la clase Address, que representa la dirección física y contiene campos para la calle, ciudad, estado y código postal.

A.1) Estado del sistema:

El sistema actualmente se encuentra en funcionamiento y todas las partes principales están operativas. Pero hay pequeños detalles en el funcionamiento, uno de ellos es al importar direcciones desde un archivo, ya que en el archivo "Direcciones.txt" que crea

el sistema por defecto cuando no se importa un archivo tiene esta nomenclatura:

```
1 Nombre: Diego Jair Dominguez de leon
2 Calle: Av. Univeersidad
3 Ciudad: Coatzacoalcos
4 Estado: Veracruz
5 Código Postal: 96566
6 Correo Electrónico: soydiego@gmail.com
7 Teléfono: 922377233
8
9 Nombre: Jesus Navarrez Espinoza Rojas
10 Calle: Abasolo
11 Ciudad: Los Mochis
12 Estado: Sinaloa
13 Código Postal: 827323
14 Correo Electrónico: jelty@gmail.com
15 Teléfono: 28973
16
17 Nombre: Hugo Sánchez
18 Calle: Miguel Hidalgo
19 Ciudad: Tamaulipas
20 Estado: Mexico
21 Código Postal: 8573
22 Correo Electrónico: soyhugo@gmail.com
23 Teléfono: 83382
24
```

Y para importar un archivo y querer mostrar las direcciones, se deben poner de esta manera:

```
1 Nombre: Jesus Navarrez, Apellido: Espinoza Rojas, Calle: Abasolo,
  Ciudad: Los Mochis, Estado: Sinaloa, Código Postal: 827323, Correo
  Electrónico: jelty@gmail.com, Teléfono: 28973
2
3 Nombre: Diego Jair, Apellido: Dominguez de leon, Calle: Av.
  Univeersidad, Ciudad: Coatzacoalcos, Estado: Veracruz, Código Postal:
  96566, Correo Electrónico: soydiego@gmail.com, Teléfono: 922377233
```

Es un detalle en cuanto a la lectura del archivo, aunque solo es estético para las direcciones, ya que el sistema lee el archivo importado, ya que si hace una nomenclatura en el documento como en el archivo por defecto que crea el sistema o de nombre "Datos.txt" el sistema lanzará una excepción del archivo:

```
java -classpath .\target\dependency\ Main
Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: a
Ingrese el nombre del archivo de texto: Prueba.txt
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException:
Index 1 out of bounds for length 1
    at AddressEntry.fromString(AddressEntry.java:46)
    at Menu.loadEntriesFromFile(Menu.java:105)
    at Menu.displayMenu(Menu.java:54)
    at Main.main(Main.java:4)
exit status 1
$
```

Igual que si se selecciona importar desde un archivo, los datos del archivo importado pasarán al archivo por defecto de "Datos.txt" cuando se importe e intente insertar una nueva entrada, no encontré solución a eso, pero me parece algo funcional, por si el

usuario necesita una copia con nuevos datos o simplemente no quiere modificar los datos de su archivo

A.2) Lista de clases:

AddressEntry (Entrada de dirección):

Propósito: Representa una entrada de dirección en la libreta de direcciones. Contiene campos para almacenar información como el nombre, dirección, correo electrónico y número de teléfono.

Estructura de datos: Utiliza variables de instancia para almacenar los datos de la entrada de dirección.

Eficiencia: Las operaciones principales de esta clase, como obtener y establecer los valores de los campos, son de complejidad $O(1)$, lo que las hace eficientes.

AddressBook (Libreta de direcciones):

Propósito: Representa una libreta de direcciones que contiene una lista de entradas de dirección.

Estructura de datos: Utiliza una lista para almacenar las entradas de dirección.

Eficiencia: La adición y eliminación de entradas de dirección son operaciones de complejidad $O(1)$ amortizado en promedio. La búsqueda de entradas de dirección por apellido tiene una complejidad de $O(n)$, donde n es el número de entradas en la libreta.

Address (Dirección):

Propósito: Representa una dirección con los campos de calle, ciudad, estado y código postal.

Estructura de datos: Utiliza variables de instancia para almacenar los datos de la dirección.

Eficiencia: Las operaciones principales de obtener y establecer los valores de los campos de dirección son de complejidad $O(1)$, lo que las hace eficientes.

Menu (Menú):

Propósito: Gestiona la interfaz de usuario y presenta un menú de opciones para que el usuario interactúe con la libreta de direcciones.

Estructura de datos: Utiliza métodos y variables locales para gestionar la presentación del menú y capturar las opciones del usuario.

Eficiencia: La eficiencia de las operaciones de menú depende de la implementación específica de cada opción. En general, las operaciones de visualización y captura de opciones son de complejidad $O(1)$.

AddressBookApplication (Aplicación de libreta de direcciones):

Propósito: Contiene la clase principal de la aplicación. Crea una instancia de AddressBook, una instancia de Menú y llama al método displayMenu() de Menu para mostrar el menú y capturar las entradas del usuario.

Estructura de datos: Utiliza objetos de las clases AddressBook y Menu.

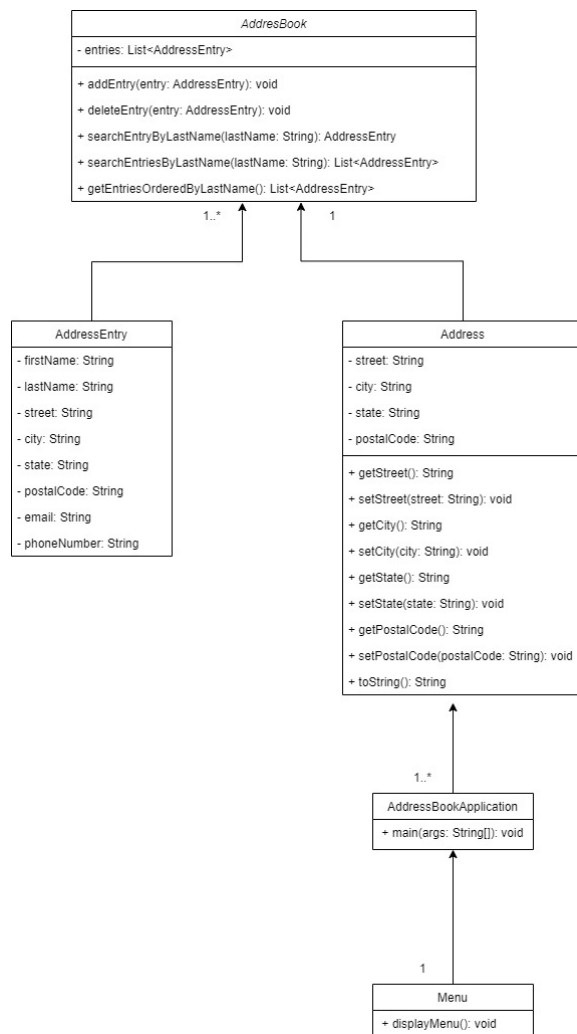
Eficiencia: La eficiencia de esta clase está determinada por las operaciones realizadas en AddressBook y Menu. Las operaciones principales de esta clase son de complejidad.

Sección B: URL de JavaDoc:

<file:///D:/Downloads/JavaDoc/JavaDoc/package-tree.html>

Si hay un error, el javadoc esta en el repositorio y en el .ZIP

Sección C: UML/Diseño:

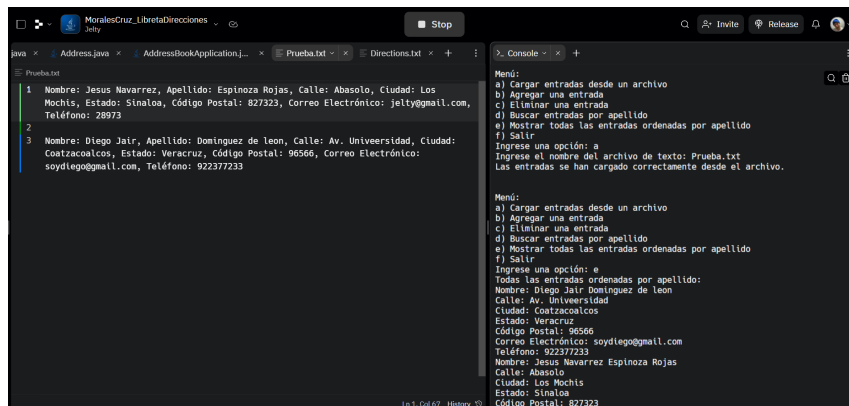


Sección D: Capturas de pantalla del sistema:

Capturas de pantalla que muestran su aplicación funcionando para los siguientes casos:

D.1) Evidencia 1:

Leer entradas del archivo de datos seguidas de Mostrar listado. El El archivo de datos debe contener un mínimo de 2 direcciones. Incluir captura de pantalla de salida y también mostrar el contenido correspondiente del archivo de datos



```
java x Address.java x AddressBookApplication.j... x Prueba.txt x Directions.txt x +
Prueba.txt
1 Nombre: Jesus Navarrez, Apellido: Espinoza Rojas, Calle: Abasolo, Ciudad: Los
Mochis, Estado: Sinaloa, Código Postal: 827323, Correo Electrónico: jeltty@gmail.com,
Teléfono: 28973
2
3 Nombre: Diego Jair, Apellido: Dominguez de leon, Calle: Av. Universidad, Ciudad:
Coatzacoalcos, Estado: Veracruz, Código Postal: 96566, Correo Electrónico:
soydlego@gmail.com, Teléfono: 922377233

Ménú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: a
Ingrese el nombre del archivo de texto: Prueba.txt
Las entradas se han cargado correctamente desde el archivo.

Ménú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: e
Todas las entradas ordenadas por apellido:
Nombre: Diego Jair Dominguez de leon
Calle: Av. Universidad
Ciudad: Coatzacoalcos
Estado: Veracruz
Código Postal: 96566
Correo Electrónico: soydlego@gmail.com
Teléfono: 922377233
Nombre: Jesus Navarrez Espinoza Rojas
Calle: Abasolo
Ciudad: Los Mochis
Estado: Sinaloa
Código Postal: 827323
```

D.2) Evidencia 2:

inmediatamente después de D.1, agregue un nuevo objeto AddressEntry seguido de un mostrar el resultado de un nuevo listado en consola

```
Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: b
Ingrese el nombre: Hugo
Ingrese el apellido: Sánchez
Ingrese la calle: Miguel Hidalgo
Ingrese la ciudad: Tamaulipas
Ingrese el código postal: 8573
Ingrese el correo electrónico: soyhugo@gmail.com
Ingrese el número de teléfono: 83382
Ingrese el estado: Mexico
La entrada se está guardando...
Las entradas se han guardado correctamente en el archivo.

Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: e
Todas las entradas ordenadas por apellido:
Nombre: Diego Jair Dominguez de leon
Calle: Av. Univeersidad
Ciudad: Coatzacoalcos
Estado: Veracruz
Código Postal: 96566
Correo Electrónico: soydiego@gmail.com
Teléfono: 922377233
Nombre: Jesus Navarrez Espinoza Rojas
Calle: Abasolo
Ciudad: Los Mochis
Estado: Sinaloa
Código Postal: 827323
Correo Electrónico: jelty@gmail.com
Teléfono: 28973
Nombre: Hugo Sánchez
Calle: Miguel Hidalgo
Ciudad: Tamaulipas
Estado: Mexico
Código Postal: 8573
Correo Electrónico: soyhugo@gmail.com
Teléfono: 83382
```

```
1) Nombre: Jesus Navarrez Espinoza Rojas
Calle: Abasolo
Ciudad: Los Mochis
Estado: Sinaloa
Código Postal: 827323
Correo Electrónico: jelty@gmail.com
Teléfono: 28973
Ingrese el número de la entrada que desea eliminar: 1
La entrada ha sido eliminada correctamente.

Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: e
Todas las entradas ordenadas por apellido:
Nombre: Diego Jair Dominguez de leon
Calle: Av. Univeersidad
Ciudad: Coatzacoalcos
Estado: Veracruz
Código Postal: 96566
Correo Electrónico: soydiego@gmail.com
Teléfono: 922377233
```

D.3) Evidencia 3:

inmediatamente después de D.2 hacer una eliminación de una entrada seguida de un mostrar listado

```
Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: c
Ingrese el inicio del apellido del usuario: Espin
Coincidencias encontradas:
1) Nombre: Jesus Navarrez Espinoza Rojas
Calle: Abasolo
Ciudad: Los Mochis
Estado: Sinaloa
Código Postal: 827323
Correo Electrónico: jelty@gmail.com
Teléfono: 28973
Ingrese el número de la entrada que desea eliminar: 1
La entrada ha sido eliminada correctamente.
```


D.4) Evidencia 4:

inmediatamente siguiendo D.3 hago una búsqueda usando una entrada que debería recuperar al menos una entrada. Ahora haga una búsqueda usando una entrada que no debería recuperar ninguna entrada. Coloque capturas de pantalla de cada hallazgo aquí.

```
Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: d
Ingrese el inicio del apellido del usuario: Espin
No se encontraron coincidencias para el apellido proporcionado.

Menú:
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar entradas por apellido
e) Mostrar todas las entradas ordenadas por apellido
f) Salir
Ingrese una opción: █
```