



UNIVERSIDAD VERACRUZANA

**FACULTAD DE CONTADURÍA Y
ADMINISTRACIÓN**

CATEDRÁTICO: Dr. José Antonio Vergara Camacho

PROGRAMA: Licenciatura Ingeniería de Software

EE: Principios de construcción de software

**Actividad 02: Proyecto final - Aplicación de consola
“libreta de direcciones”**

GRUPO: Lis-401

Descripción

El programa es una aplicación de libreta de direcciones que permite a los usuarios gestionar y almacenar información de contactos. Utiliza la programación orientada a objetos y el patrón de diseño Singleton

En un patrón Singleton, la clase en cuestión tiene un constructor privado para evitar la creación directa de instancias desde otras partes del programa. En su lugar, la clase proporciona un método estático para obtener la única instancia disponible.

La implementación del patrón Singleton puede variar dependiendo del lenguaje de programación utilizado. En Java, se utiliza un campo estático privado dentro de la clase para almacenar la única instancia, y el método estático de acceso (a menudo llamado `getInstance()`) se encarga de crear la instancia si aún no existe y devolverla en cualquier caso.

El código se distribuyó en dos paquetes: "Directorio.AddressBook" contiene las clases relacionadas con la manipulación de los datos del directorio de direcciones, mientras que el paquete "Directorio.Address" contiene las clases relacionadas con la interfaz de usuario y las interacciones con el usuario. Estas clases trabajan juntas para permitir al usuario agregar, eliminar, buscar, mostrar y guardar entradas en el directorio de direcciones.

A continuación se presenta un listado general de las clases de la libreta de direcciones.

Clases del paquete "Directorio.AddressBook":

Clase AddressBook:

Descripción: Representa el directorio de direcciones y contiene los métodos para manipular los datos en el directorio.

Atributos:

entries: Lista de objetos AddressEntry que representan las entradas de la libreta de direcciones.

Métodos principales:

- **addEntry**(AddressEntry entry): Agrega una nueva entrada al directorio.
- **removeEntry**(AddressEntry entry): Elimina una entrada del directorio.
- **searchEntryByLastName**(String lastName): Busca entradas en el directorio por apellido.
- **getAllEntriesInAlphabeticalOrder**(): Obtiene todas las entradas en el directorio en orden alfabético.

Clase AddressEntry:

Descripción: Representa una entrada individual en el directorio de direcciones.

Atributos:

Información personal: nombre, apellido, dirección, ciudad, estado, código postal.

Información de contacto: email, número de teléfono.

Métodos principales:

Métodos de acceso y modificación para los atributos de la entrada.

Clases del paquete "Directorio.Address":

Clase AddressBookApplication:

Descripción: Clase principal que inicia la aplicación del directorio de direcciones.

Atributos:

menu: Referencia a la clase Menu para mostrar el menú y gestionar las interacciones con el usuario.

addressBook: Referencia a la clase AddressBook para acceder al directorio de direcciones.

Métodos principales:

- **run():** Inicia la aplicación mostrando el menú.

Clase Menu:

Descripción: Permite al usuario manipular los datos del directorio de direcciones mediante un menú interactivo.

Atributos:

sc: Objeto de la clase Scanner para leer la entrada del usuario.

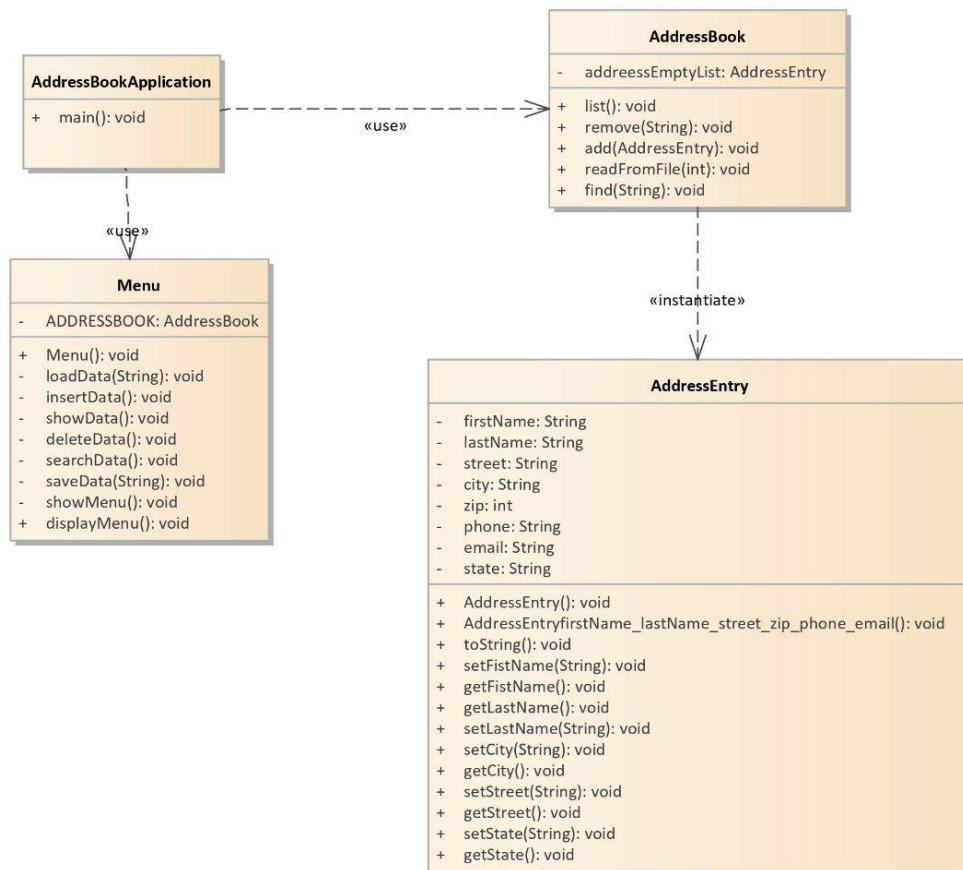
ADDRESSBOOK: Referencia a la clase AddressBook para acceder al directorio de direcciones.

Métodos principales:

Métodos para cargar, insertar, mostrar, eliminar, buscar y guardar datos en el directorio.

- **displayMenu():** Muestra el menú y maneja las interacciones con el usuario.

UML - Modelo de Clases



Capturas de pantalla

Evidencia 1:

Leer entradas del archivo de datos seguidas de Mostrar listado. 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

```
===== MENU =====  
a) Cargar entradas desde un archivo  
b) Agregar una entrada  
c) Eliminar una entrada  
d) Buscar una entrada  
e) Mostrar todas las entradas  
f) Guardar datos en un CSV  
g) Salir
```

```
=====
Ingrese su opción: e
Name: Mario Condado
Street: Lopez Ortiz
City: NY
State: Kansas
Postal Code: 97854
Email: edoardo
Phone Number: edoado@gmail.com
=====
Name: Jose Francisco
Street: Santes
City: Primo
State: NY
Postal Code: 96730
Email: Edoarod
Phone Number: jose@gmail.com
=====
Presione Enter para continuar...
```

Evidencia 2

Inmediatamente después de la evidencia 1, agregue un nuevo objeto AddressEntry seguido de un mostrar el resultado de un nuevo listado en consola

```
===== MENU =====
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar una entrada
e) Mostrar todas las entradas
f) Guardar datos en un CSV
g) Salir
=====
Ingrese su opción: b
Agregar una entrada:
Nombre: Cacarlin
Apellido: Montevideo
Dirección: Avenida Siempre Viva
Ciudad: NY
Estado: NY
Código Postal: 87564
Número de teléfono: 988445631
Email: ArribaLasChivas@gmail.com
La entrada se ha agregado correctamente.
Presione Enter para continuar...
```

```
=====
Ingrese su opción: e
Name: Mario Condado
Street: Lopez Ortiz
City: NY
State: Kansas
Postal Code: 97854
Email: edoardo
Phone Number: edoado@gmail.com
=====
Name: Jose Francisco
Street: Santes
City: Primo
State: NY
Postal Code: 96730
Email: Edoarod
Phone Number: jose@gmail.com
=====
Name: Cacarlin Montevideo
Street: Avenida Siempre Viva
City: NY
State: NY
Postal Code: 87564
Email: 988445631
Phone Number: ArribaLasChivas@gmail.com
=====
```

Evidencia 3:

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

```
===== MENU =====
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar una entrada
e) Mostrar todas las entradas
f) Guardar datos en un CSV
g) Salir
=====
Ingrese su opción: c
Eliminar
Ingrese el apellido de la entrada a eliminar: Francisco
Se encontraron las siguientes entradas:
1. Name: Jose Francisco
Street: Santes
City: Primo
State: NY
Postal Code: 96730
Email: Edoarod
Phone Number: jose@gmail.com
Ingrese el número de la entrada a eliminar: 1
La entrada se ha eliminado correctamente.
Presione Enter para continuar...
```

```
===== MENU =====
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar una entrada
e) Mostrar todas las entradas
f) Guardar datos en un CSV
g) Salir
=====
Ingrese su opción: e
Name: Mario Condado
Street: Lopez Ortiz
City: NY
State: Kansas
Postal Code: 97854
Email: edoardo
Phone Number: edoado@gmail.com
=====
Name: Cacarlín Montevideo
Street: Avenida Siempre Viva
City: NY
State: NY
Postal Code: 87564
Email: 988445631
Phone Number: ArribaLasChivas@gmail.com
=====
```


Evidencia 4:

Inmediatamente siguiendo la evidencia 4 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.

```
===== MENU =====
a) Cargar entradas desde un archivo
b) Agregar una entrada
c) Eliminar una entrada
d) Buscar una entrada
e) Mostrar todas las entradas
f) Guardar datos en un CSV
g) Salir
=====
Ingrese su opción: d
Buscar
Ingrese el apellido a buscar: Montevideo
Se encontraron las siguientes entradas:
Name: Cacarlin Montevideo
Street: Avenida Siempre Viva
City: NY
State: NY
Postal Code: 87564
Email: 988445631
Phone Number: ArribaLasChivas@gmail.com
Presione Enter para continuar...
|
```