UNIVERSIDAD DEL BIO-BIO
FACULTAD DE CIENCIAS EMPRESARIALES
ESCUELA DE INGENIERÍA CIVIL INFORMÁTICA

PROYECTO SEMESTRAL 2024-2
PRIMER AVANCE
Programación Orientada a Objetos

Profesores: Ma. Antonieta Soto Ch., Miguel Romero V. Fecha: 30-08-2024

A. ASPECTOS GENERALES

Se necesita construir una aplicación para una empresa que vende pasajes de bus de diversas empresas y variados recorridos. Por ahora solo consideraremos pasajes para viajes en bus de una única empresa de trasporte de pasajeros que realiza un único recorrido similar al de Chillán Concepción.

En esta primera versión de la aplicación solo consideraremos las funcionalidades relacionadas con la creación de clientes, buses y viajes, la venta de pasajes, así como un listado de las ventas realizadas a la fecha, otro listado con todos los viajes existentes indicando diversos datos entre ellos número de asientos ocupados y disponibles, y un listado de los pasajeros de un cierto viaje realizado por un cierto bus.

La metodología de trabajo será incremental, es decir, se harán entregas parciales o avances que implementarán parte de las funcionalidades, en particular se desarrollarán cuatro avances, tal que el avance final corresponderá a la aplicación completa.

B. DISEÑO DE LA APLICACIÓN

B.1. DIAGRAMA DE CLASES

Para el primer avance, se debe implementar el diagrama de clases que se presenta en la Figura 1 y Figura 2. Las clases presentes se describen más adelante en este documento.

Es importante destacar que la aplicación se debe construir basándose estrictamente en el diagrama de clases antes mencionado. Es decir, al finalizar el primer avance, se deberá contar con la implementación de todas las clases, relaciones, atributos y métodos presentes en dicho diagrama de clases, asegurándose de utilizar todos y cada uno de los métodos. No se pueden añadir constantes, enumeraciones, atributos, relaciones, constructores o métodos a las clases, ni tampoco agregar clases. Solo se permite agregar métodos privados, si ello lleva a una mejora del código o se logra mayor reúso, simplicidad y/o legibilidad.

Una excepción la constituyen las clases que no presentan expresamente un constructor en el diagrama de clases, en tales casos debe considerar la existencia de un constructor sin parámetros, el cual podrá implementar en su código o simplemente omitirlo usando el constructor por defecto que Java contempla en esos casos. Si el constructor omitido se encuentra en clases relacionadas por herencia, deben agregar constructores consistentes con la ubicación de la o las clases en el árbol de herencia correspondiente.

En el Anexo A, al final de este documento, se describen las tareas y consideraciones relevantes que se deben tener presentes al implementar los constructores y métodos del diagrama de clases. No se incluyen constructores y métodos excesivamente simples.

UNIVERSIDAD DEL BIO-BIO
FACULTAD DE CIENCIAS EMPRESARIALES
ESCUELA DE INGENIERÍA CIVIL INFORMÁTICA

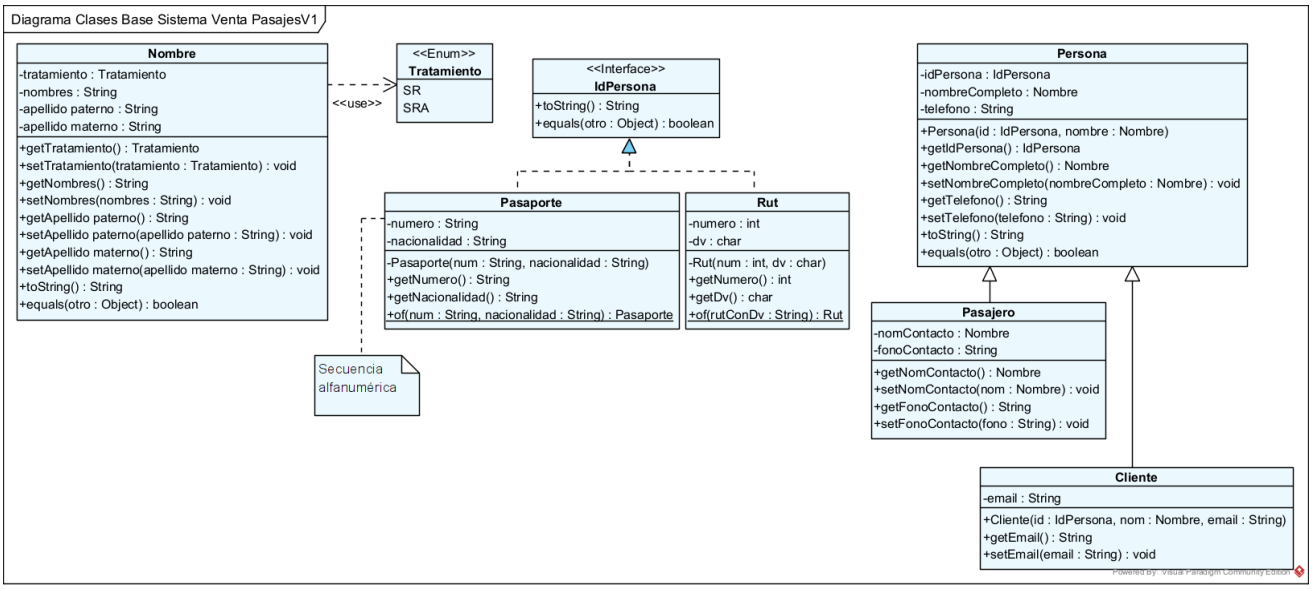

Figura 1. Diagrama de clases del primer avance, parte I.

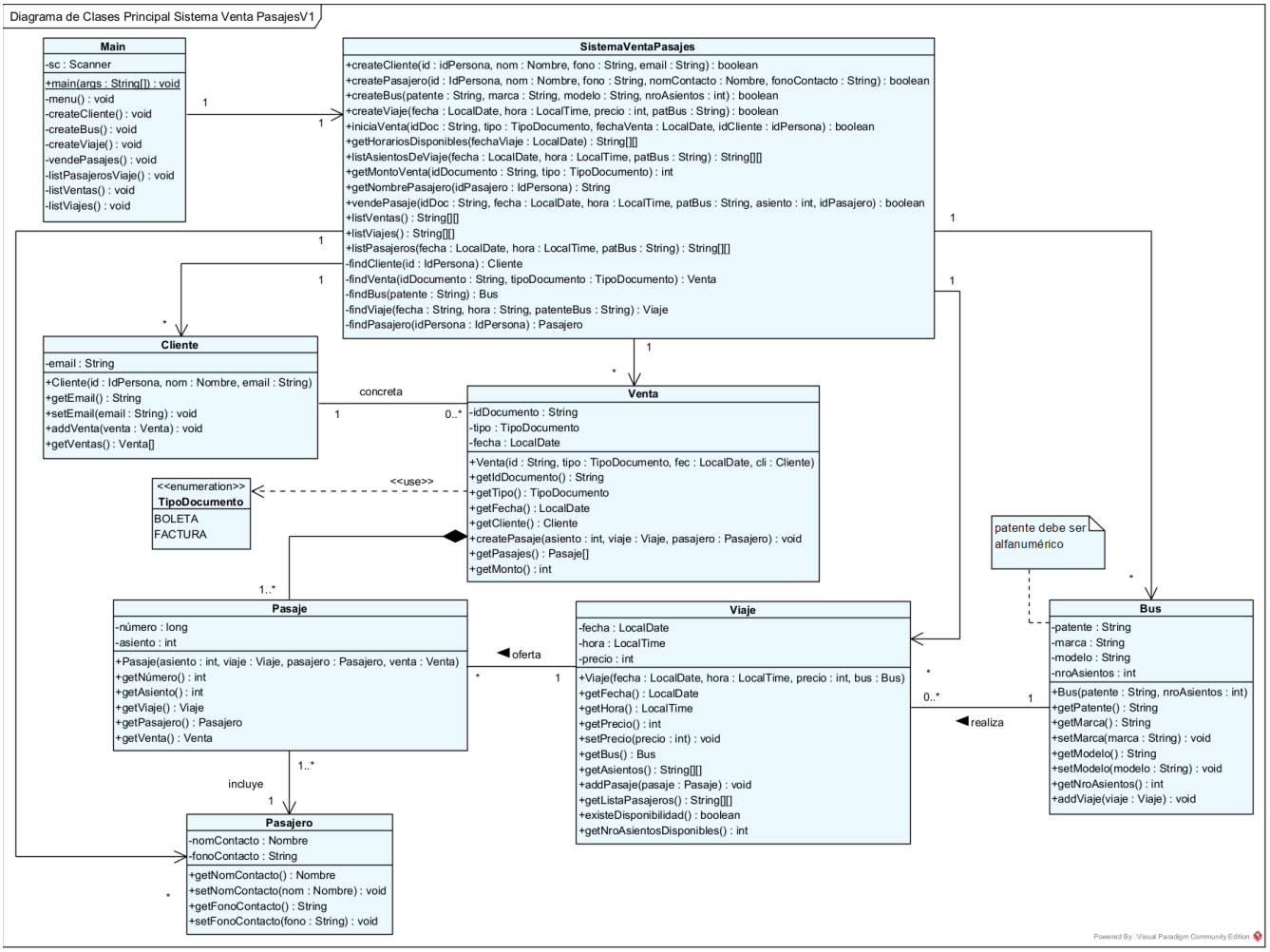

Figura 2. Diagrama de clases del primer avance, parte II.

B.2. INTERFAZ DE USUARIO DEL PRIMER AVANCE

Durante las versiones iniciales de la aplicación se implementará una interfaz de usuario ASCII. El primer avance contempla el despliegue de un menú principal como el que se muestra en la Figura 3.

Se debe validar que la opción ingresada por el usuario esté en el rango válido, en caso contrario se debe desplegar un mensaje indicando el error.

En cuanto a los restantes datos que se deban leer, supondremos en este primer avance que dichos datos siempre serán correctos, por lo que no necesitará validarlos. Sin embargo, debe contemplar las situaciones que expresamente se indican en cada funcionalidad descrita en esta sección.

```
============================
...::: Menú principal :::...

  1) Crear cliente
  2) Crear bus
  3) Crear viaje
  4) Vender pasajes
  5) Lista de pasajeros
  6) Lista de ventas
  7) Lista de viajes
  8) Salir
----------------------------
..:: Ingrese número de opción:
```

Figura 3. Ventana del menú principal.

Descripción de las funcionalidades del menú

Crear cliente

Al crear un cliente, se debe solicitar el ingreso de los datos necesarios, estos son: tipo de identificador, identificador que corresponda (según elección previa), tratamiento, primer nombre, apellido paterno, apellido materno, teléfono y un email. El identificador debe leerse en el formato que se espera recibir, para lo cual debe dar las indicaciones al usuario, de modo que este lo ingrese correctamente. Respecto del tratamiento, se debe permitir al usuario seleccionar el adecuado (Sr o Sra). La Figura 4 muestra la creación de un nuevo cliente. Si un cliente no puede crearse porque ya existe otro con el mismo identificador se debe desplegar un mensaje apropiado. En caso de poder crear el cliente se debe informar al usuario la creación exitosa del mismo.

```
  ...:::: Crear un nuevo Cliente ::::....

 Rut[1] o Pasaporte[2] : 1
                R.U.T : 11.111.111-1
    Sr.[1] o Sra. [2] : 1
              Nombres : Juan Jose
     Apellido Paterno : Perez
     Apellido Materno : Rios
       Telefono movil : 948753235
                Email : jjperez@gmail.com

...:::: Cliente guardado exitosamente ::::....
```

Figura 4. Pantalla creación exitosa de un cliente.

Crear bus

Para crear un bus, se deben solicitar los datos correspondientes, esto es, patente, marca, modelo y número de asientos, tal como se muestra en la Figura 5. Si un bus no puede crearse porque ya existe otro con la misma patente se debe desplegar un mensaje apropiado. De igual modo, se debe informar al usuario la creación exitosa del bus.

```
...:::: Creación de un nuevo BUS ::::....

             Patente : AZVB25
               Marca : Mercedes Benz
              Modelo : Centauro
  Número de asientos : 40

...:::: Bus guardado exitosamente ::::....
```

Figura 5. Pantalla creación exitosa de un bus.

Crear viaje

Esta opción permitirá crear un viaje, para ello se debe leer fecha (formato dd/mm/aaaa), hora (formato hh:mm), precio y patente del bus (alfanumérico), tal como se muestra en la Figura 6. Si no existe un bus con la patente ingresada o si ya existe un viaje para la fecha, hora y bus ingresados, se deberá informar al usuario mediante un mensaje apropiado. Al igual que en los casos previos, debe informar al usuario la creación exitosa del viaje.

```
...::: Creación de un nuevo Viaje :::....

    Fecha[dd/mm/yyyy] : 02/09/2024
          Hora[hh:mm] : 10:20
               Precio : 4100
          Patente Bus : AZVB25

...::: Viaje guardado exitosamente :::....
```

Figura 6. Pantalla creación exitosa de un viaje.

Vender pasajes

La Figura 7 muestra la pantalla de inicio de la venta en la que se solicita el identificador de documento de la nueva venta, el tipo de documento (boleta o factura), la fecha de la venta, el tipo de identificador del cliente (rut o pasaporte) y el identificador del cliente que desea comprar pasajes, el cual deberá leerse considerando el tipo de identificador escogido por el usuario. Si no es posible realizar la venta porque ya existe otra venta con el identificador de documento ingresado o no existen viajes o no existe disponibilidad de asientos en la fecha indicada, o bien, no existe un cliente con el identificador ingresado, se debe desplegar un mensaje al usuario señalando que la venta no se puede concretar, dando término a la misma.

```
        ...::: Venta de pasajes :::....


:::: Datos de la Venta
             ID Documento : 10012
Tipo documento: [1] Boleta [2] Factura : 1
Fecha de venta[dd/mm/yyyy] : 30/08/2024

:::: Datos del cliente

    Rut[1] o Pasaporte[2] : 1
                    R.U.T : 11.111.111-1
           Nombre Cliente : Sr. José Esteban Cid Aedo
```

Figura 7. Pantalla venta de pasajes, parte inicial.

Si es posible iniciar la venta, se debe proceder a solicitar al usuario el número de pasajes que desea comprar el cliente y la fecha del viaje, a partir de esos datos se deben desplegar los datos de los viajes disponibles para la fecha ingresada anteriormente. Se solicita al usuario que ingrese el viaje que ha escogido como se muestra en la Figura 8.

```
:::: Pasajes a vender
      Cantidad de pasajes : 2
Fecha de viaje[dd/mm/yyyy] : 31/08/2024

:::: Listado de horarios disponibles
     *----------*----------*----------*----------*
     | BUS      |   SALIDA |    VALOR | ASIENTOS |
     |----------+----------+----------+----------|
   1 | AB.CD-12 |    06:15 |   $3.000 |       45 |
     |----------+----------+----------+----------|
   2 | EF.GH-34 |    07:00 |   $3.500 |       52 |
     |----------+----------+----------+----------|
   3 | IJ.KL-56 |    07:45 |   $4.000 |       40 |
     |----------+----------+----------+----------|
   4 | MN.OP-78 |    08:30 |   $5.000 |       38 |
     |----------+----------+----------+----------|
   5 | QR.ST-90 |    09:15 |   $5.500 |       44 |
     |----------+----------+----------+----------|
   6 | AB.CD-12 |    10:00 |   $3.000 |       45 |
     |----------+----------+----------+----------|
   7 | EF.GH-34 |    10:45 |   $3.500 |       52 |
     |----------+----------+----------+----------|
   8 | IJ.KL-56 |    11:30 |   $4.000 |       40 |
     |----------+----------+----------+----------|
   9 | MN.OP-78 |    12:15 |   $5.000 |       38 |
     |----------+----------+----------+----------|
  10 | QR.ST-90 |    13:00 |   $5.500 |       44 |
     |----------+----------+----------+----------|
  11 | AB.CD-12 |    13:45 |   $3.000 |       45 |
     |----------+----------+----------+----------|
  12 | EF.GH-34 |    14:30 |   $3.500 |       52 |
     |----------+----------+----------+----------|
  13 | IJ.KL-56 |    15:15 |   $4.000 |       40 |
     |----------+----------+----------+----------|
  14 | MN.OP-78 |    16:00 |   $5.000 |       38 |
     |----------+----------+----------+----------|
  15 | QR.ST-90 |    16:45 |   $5.500 |       44 |
     |----------+----------+----------+----------|
  16 | AB.CD-12 |    17:30 |   $3.000 |       45 |
     |----------+----------+----------+----------|
  17 | EF.GH-34 |    18:15 |   $3.500 |       52 |
     |----------+----------+----------+----------|
  18 | IJ.KL-56 |    19:00 |   $4.000 |       40 |
     |----------+----------+----------+----------|
  19 | MN.OP-78 |    19:45 |   $5.000 |       38 |
     |----------+----------+----------+----------|
  20 | QR.ST-90 |    20:30 |   $5.500 |       44 |
     *----------*----------*----------*----------*
Seleccione viaje en [1..20] :
```

Figura 8. Pantalla venta de los pasajes solicitados, primera parte.

En la Figura 9 se presenta la pantalla en la que se despliegan los asientos libres (aparecen números de asiento) y ocupados (aparecen *) del viaje seleccionado. Se debe solicitar al usuario que seleccione el o los asientos que desea (que debe corresponder al número de pasajes que ingresó previamente). A continuación, por cada asiento seleccionado, se debe solicitar el identificador del pasajero, habiendo consultado previamente el tipo de identificador correspondiente. Si no existe un pasajero con el id indicado, se deben solicitar los restantes datos de este para proceder a crearlo (nombre, teléfono, nombre del contacto, teléfono del contacto). Al finalizar la venta de los pasajes solicitados (ya registrados en el sistema) se despliega el monto a pagar por el cliente.

```
:::: Asientos disponibles para el viaje seleccionado
*---*---*---*---*---*
| 1 | 2 |   | 4 | 3 |
|---+---+---+---+---|
| 5 | 6 |   | 8 | 7 |
|---+---+---+---+---|
| 9 | 10|   | 12| 11|
|---+---+---+---+---|
| 13| 14|   | 16| 15|
|---+---+---+---+---|
| 17| 18|   | 20| 19|
|---+---+---+---+---|
| 21| 22|   | 24| 23|
|---+---+---+---+---|
| 25| 26|   | 28| 27|
|---+---+---+---+---|
| 29| 30|   | 32| 31|
|---+---+---+---+---|
| 33| 34|   | 36| 35|
|---+---+---+---+---|
| 37| 38|   | 40| 39|
|---+---+---+---+---|
| 41| 42|   | 44| 43|
|---+---+---+---+---|
| 45| * |   | * | * |
*---*---*---*---*---*
Seleccione sus asientos [separe por ,] : 12,11
```

```
:::: Datos pasajeros 1
    Rut[1] o Pasaporte[2] : 1
                    R.U.T : 33.333.333-3

:::: Pasaje agregado exitosamente

:::: Datos pasajeros 2
    Rut[1] o Pasaporte[2] : 1
                    R.U.T : 44.444.444-4

:::: Pasaje agregado exitosamente

:::: Monto total de la venta: $6.000

   ...:::: Venta generada exitosamente ::::....
```

Figura 9. Pantalla venta de los pasajes solicitados, segunda parte.

La Figura 10 muestra la pantalla que corresponde a la parte final de la venta de pasajes, en la que se despliegan los pasajes (boletos) que se han vendido al cliente.

```
:::: Imprimiendo los pasajes

------------------ PASAJE --------------------
NUMERO DE PASAJE : 1724990705229
FECHA DE VIAJE   : 31/08/2024
HORA DE VIAJE    : 10:00
PATENTE BUS      : AB.CD-12
ASIENTO          : 12
RUT/PASAPORTE    : 33.333.333-3
NOMBRE PASAJERO  : Sra. Ángela Eugenia Muñoz Naranjo
---------------------------------------------



------------------ PASAJE --------------------
NUMERO DE PASAJE : 1724990724745
FECHA DE VIAJE   : 31/08/2024
HORA DE VIAJE    : 10:00
PATENTE BUS      : AB.CD-12
ASIENTO          : 11
RUT/PASAPORTE    : 44.444.444-4
NOMBRE PASAJERO  : Sr. Carlos Patricio Figueroa Orellana
---------------------------------------------
```

Figura 10. Pantalla con despliegue de pasajes vendidos.

Listar pasajeros de un viaje

La Figura 11 muestra una pantalla con el despliegue de los pasajeros de un viaje. Para obtener el listado se debe ingresar fecha y hora del viaje, así como la patente del bus. A partir de estos datos, se genera el listado, el cual se despliega con los datos y formato que se presentan en la Figura 11. En caso de no existir el bus o un viaje con los datos indicados se debe informar al usuario con un mensaje apropiado.

```
...:::: Listado de pasajeros de un viaje ::::....

Fecha del viaje[dd/mm/yyyy] : 31/08/2024
    Hora del viaje[hh:mm] : 10:00
            Patente bus : AB.CD-12
*---------*----------------*-------------------------------*-------------------------------*-------------------*
| ASIENTO |        RUT/PASS | PASAJERO                      | CONTACTO                      | TELEFONO CONTACTO |
|---------+----------------+-------------------------------+-------------------------------+-------------------|
|      12 |    33.333.333-3 | Sra. Ángela Eugenia Muñoz Naran| Sra. María Paz Daza Barrera    | (+569) 1672 3421  |
|---------+----------------+-------------------------------+-------------------------------+-------------------|
|      11 |    44.444.444-4 | Sr. Carlos Patricio Figueroa Or| Sr. José Esteban Cid Aedo      | (+569) 1122 3344  |
*---------*----------------*-------------------------------*-------------------------------*-------------------*
```

Figura 11. Pantalla con listado de pasajeros de un viaje.

Listar ventas

La Figura 12 muestra una pantalla con el despliegue de las ventas realizadas. Observe los datos que se incluyen y el formato de la salida. Si no existen ventas registradas en el sistema se debe indicar al usuario desplegando un mensaje apropiado.

```
        ...:::: Listado de ventas ::::....

*------------*----------*------------*----------------*--------------------------------*--------------*--------------*
| ID DOCUMENT| TIPO DOCU|      FECHA |   RUT/PASAPORTE | CLIENTE                        | CANT BOLETOS |  TOTAL VENTA |
|------------+----------+------------+-----------------+--------------------------------+--------------+--------------|
|      10012 | BOLETA   | 30/08/2024 |    11.111.111-1 | Sr. José Esteban Cid Aedo      |            2 |       $6.000 |
*------------*----------*------------*----------------*--------------------------------*--------------*--------------*
```

Figura 12. Pantalla con listado de ventas realizadas.

Listar viajes

La Figura 13 muestra una pantalla con el despliegue de los viajes registrados en el sistema. Observe los datos que se incluyen y el formato de la salida. Si no existen viajes registrados, se debe indicar al usuario desplegando un mensaje apropiado.

```
        ...:::: Listado de viajes ::::....

*--------------*--------------*--------------*--------------*--------------*
|        FECHA |         HORA |       PRECIO |   DIPONIBLES | PATENTE      |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        06:15 |       $3.000 |           45 | AB.CD-12     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        07:00 |       $3.500 |           52 | EF.GH-34     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        07:45 |       $4.000 |           40 | IJ.KL-56     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        08:30 |       $5.000 |           38 | MN.OP-78     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        09:15 |       $5.500 |           44 | QR.ST-90     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        10:00 |       $3.000 |           43 | AB.CD-12     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        10:45 |       $3.500 |           52 | EF.GH-34     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        11:30 |       $4.000 |           40 | IJ.KL-56     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        12:15 |       $5.000 |           38 | MN.OP-78     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        13:00 |       $5.500 |           44 | QR.ST-90     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        13:45 |       $3.000 |           45 | AB.CD-12     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        14:30 |       $3.500 |           52 | EF.GH-34     |
|--------------+--------------+--------------+--------------+--------------|
|   31/08/2024 |        15:15 |       $4.000 |           40 | IJ.KL-56     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        16:45 |       $5.500 |           44 | QR.ST-90     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        17:30 |       $3.000 |           45 | AB.CD-12     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        18:15 |       $3.500 |           52 | EF.GH-34     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        19:00 |       $4.000 |           40 | IJ.KL-56     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        19:45 |       $5.000 |           38 | MN.OP-78     |
|--------------+--------------+--------------+--------------+--------------|
|   29/09/2024 |        20:30 |       $5.500 |           44 | QR.ST-90     |
*--------------*--------------*--------------*--------------*--------------*
```

Figura 13. Pantalla con listado de viajes registrados.

C. CARACTERÍSTICAS DEL TRABAJO A PRESENTAR

En este primer avance:

- Los estudiantes conformarán equipos de 4 estudiantes designados por el/la profesor/a. De manera excepcional un equipo podrá estar conformado por más o menos estudiantes, según lo determine el/la profesora a cargo. Cada equipo debe darse un nombre. Uno de los integrantes del equipo registrará el nombre del equipo y de sus integrantes en el foro dispuesto para ello, en la sección del proyecto semestral en Moodle.
- **No** se contempla documentación escrita tipo informe.
- Se debe llevar un tablero para este primer avance en Trello donde se almacenará este documento y algún otro recurso de apoyo. Lo más importante, sin embargo, es que se especificar las tareas por desarrollar, en desarrollo, en revisión de pares o finalizadas por cada integrante del equipo. Durante todo el período de desarrollo se deberá mostrar en dicho tablero el grado de avance de esta versión del proyecto y el trabajo de cada integrante con las evidencias que correspondan.
- Se debe entregar el código de esta versión del proyecto a través de GitHub, no existirá otro medio. Todos los integrantes deben evidenciar el trabajo realizado a través de sus contribuciones que deben quedar registradas en su rama del proyecto dentro de Github y en la versión final integrada en la rama main. Dicho trabajo debe ser significativo en importancia y magnitud, así como proporcional con el de los demás integrantes del equipo.
- Los profesores pueden citar a uno o más equipos para que presenten o expliquen su trabajo, en cuyo caso todos los integrantes deberán acudir a tal citación.

D. PLAZOS

El código final del primer avance del proyecto deberá estar disponible en la rama main de GitHub antes de las 23:55 hrs. del día 23/09/2024. Solo se revisará el código que cumpla lo anterior.

E. OTROS

El código de cada clase deberá incluir el nombre completo de los autores al inicio y respetar las demás convenciones de los programas escritos en Java (sangrías, nombres de variables significativos, etc.).

¡IMPORTANTE!

- Las copias, parciales o totales, facilitadas por cualquier medio, incluidas herramientas automatizadas, serán evaluadas con nota 1,0. Esta nota será compartida por todas las versiones de proyecto involucradas en la copia.

- Si la versión de un proyecto no se encuentra en GitHub o se entrega fuera de plazo será evaluado con nota 1,0.
- La nota de cada integrante del equipo dependerá del grado de contribución o aporte real que haga al trabajo total que demande la versión correspondiente del proyecto.
- Quienes no participen en la sesión de presentación o explicación de la versión del proyecto, al que un profesor/a haya citado, podrán ser evaluados con nota 1,0. Se eximen de lo anterior quienes hayan justificado su inasistencia formalmente a través del Departamento de Desarrollo Estudiantil o de la Jefatura de Carrera. En todo caso, el/la profesor/a podrá citar a tales estudiantes a una sesión recuperativa de la actividad.

ANEXO A. DOCUMENTACIÓN DE LAS CLASES DEL MODELO

La Tabla 1 presenta una descripción de los constructores y métodos más relevantes de las clases presentes en el diagrama de clases del proyecto.

Clase/ Interface	Constructor/ Método	Descripción
Persona	**toString**	Retorna un String con el id, nombre completo y teléfono separados por una coma.
	equals	Retorna true si el id es igual.
Nombre	**toString**	Retorna un String con el tratamiento, nombre, apellido paterno y apellido materno separados por un espacio.
	equals	Retorna true si el nombre, apellido paterno y apellido materno son iguales
Pasaporte	**Constructor**	Se espera que los datos pasados como parámetro sean válidos.
	toString	Retorna un String con el formato XXXXXXXX XXXXXXXXXXXX, correspondiente al número y a la nacionalidad.
	equals	Retorna true si el número y la nacionalidad son iguales.
	of	Retorna un objeto Pasaporte si los datos son correctos, o null en caso contrario.
Rut	**Constructor**	Se espera que los datos pasados como parámetro sean válidos.
	toString	Retorna un String con el formato 99.999.999-X o 9.999.999-X.
	equals	Retorna true si el número y el dígito verificador son iguales.
	of	Retorna un objeto Rut si el String contiene el número y el dígito verificador, con puntos y guion, y estos son correctos, de lo contrario retorna null.
Venta	**Constructor**	Crea una nueva venta asignado a los atributos los parámetros recibidos. Asocia el cliente a la venta, preocupándose de que el objeto Cliente agregue esta venta en su colección.
	createPasaje	Crea un objeto Pasaje, a partir de los datos recibidos como parámetro, agregándolo a su colección.
Pasaje	**Constructor**	Crea un nuevo Pasaje considerando los datos recibidos como parámetro, los cuales supone son correctos. El número de pasaje se obtiene usando un método de alguna clase de la API de Java que permita generar números aleatorios no duplicados. Finalmente, el constructor se preocupa que el objeto Viaje agregue este pasaje a su colección.
Viaje	**Constructor**	Crea un nuevo Viaje con los datos que recibe como parámetro, los cuales supone correctos, asegurándose que el objeto Bus correspondiente agregue a su colección este viaje.
	getAsientos	Retorna un arreglo bidimensional donde, por cada asiento existente en el bus para este viaje, indica su número (valor mayor o igual a 1) y si este se encuentra libre u ocupado.

	getListaPasajeros	Retorna un arreglo bidimensional con los datos de los pasajeros que tienen un pasaje asignado. Por cada pasajero (fila) indica: id, nombre, nombre del contacto y número de teléfono del contacto, cada uno de estos datos en una columna separada.
	existeDisponibilidad	Retorna true si para este viaje existe, al menos, un asiento disponible.
	getNroAsientosDisponibles	Retorna el número de asientos que aún tiene disponibles (no se han vendido).
SistemaVentaPasajes	**createCliente**	Crea un objeto Cliente con los datos que se reciben como parámetro y se almacena en la colección correspondiente, siempre que no exista otro cliente con el mismo idPersona. Retorna true si la acción se puede realizar, false en caso contrario.
	createPasajero	Crea un objeto Pasajero con los datos que se reciben como parámetro y se almacena en la colección correspondiente, siempre que no exista otro pasajero con el mismo idPersona. Retorna true si la acción se puede realizar, false en caso contrario.
	createBus	Crea un objeto Bus con los datos que se reciben como parámetro y se almacena en la colección correspondiente, siempre que no exista otro bus con la misma patente. Retorna true si la acción se puede realizar, false en caso contrario.
	createViaje	Crea un objeto Viaje a partir de los datos que se reciben como parámetro y se almacena en la colección correspondiente, siempre que no exista otro viaje con la misma fecha y hora de salida para el bus que se indica. Retorna true si la acción se puede realizar, false en caso contrario.
	getHorariosDisponibles	Retorna un arreglo bidimensional con datos relevantes de los viajes que se realizarán/realizaron en la fecha que se pasa como parámetro. Los datos que se incluyen por cada viaje son patente del bus que lo realiza, hora, precio de un pasaje y el número de asientos disponibles de. Si no existen viajes en la fecha que se indica, el método retorna un arreglo de tamaño cero.
	listAsientosDeViaje	Retorna un arreglo unidimensional donde, por cada asiento del viaje con fecha y hora que se indican como parámetro, que posee el bus cuya patente se recibe como tercer parámetro, indica si se encuentra ocupado o libre. El método retorna un arreglo de tamaño cero si no existe un viaje con los datos que se indican.
	iniciaVenta	Crea una nueva venta con los datos que se reciben como parámetro. Nótese que el cliente cuyo id se recibe como parámetro ya debiera existir en el sistema. El método retorna true si se puede realizar la acción y false si no es posible llevarlo a cabo porque ya existe una venta con el idDocumento dado o si no existe un cliente con el id dado.

	getMontoVenta	Retorna el monto de la venta cuyo idDocumento y tipo se pasan como parámetro. Si no existe una venta con los datos que se indican, retorna cero.
	getNombrePasajero	Retorna el nombre del pasajero cuyo idPasajero se pasa como parámetro, null en caso de que no exista un pasajero con el id dado.
	vendePasaje	Solicita a la venta, cuyo idDocumento y tipo se recibe como parámetros, crear un nuevo pasaje asociado al viaje cuya fecha y hora de salida se indican asociado al bus con la patente dada, ligando al nuevo pasaje, el pasajero correspondiente. Respecto de este último, el método recupera el pasajero cuyo idPersona se recibe como parámetro. El método retorna true si es posible crear el nuevo pasaje asociándolo a la venta. Si no es posible crear el nuevo pasaje porque no existe una venta con el idDocumento y tipo dados o no existe un viaje con los datos dados o no existe un bus cuya patente se indica o no existe un pasajero con el id dado, el método retorna false.
	listVentas	Retorna un arreglo bidimensional con los datos que se observan en el listado de ventas realizadas de la Figura 12
	listViajes	Retorna un arreglo bidimensional con los datos que se observan en el listado de viajes registrados de la Figura 13.
	listPasajeros	Retorna un arreglo bidimensional con los datos que se observan en el listado de pasajeros del viaje cuyos datos se indican incluyendo la patente del bus que realiza dicho viaje.
	findXXXXX	Busca, dentro de la colección que corresponda según la clase que se indica en el nombre del método (XXXXX), el objeto que coincida con el o los datos que se pasan como parámetro, los cuales constituyen el criterio de búsqueda. El método retorna el objeto que coincida con el criterio de búsqueda. Si ningún objeto coincide con el criterio de búsqueda, el método retorna null.
Main	**menu**	Despliega el menú principal, lee la opción escogida por el usuario y, dependiendo de la opción, invoca alguno de los métodos de la propia clase Main. Si la opción es errónea despliega un mensaje indicando el problema. Finaliza la ejecución del programa cuando el usuario escoge Salir, desplegando un mensaje apropiado. Ver Figura 3.
	createCliente	Despliega un título apropiado para que el usuario observe que ha accedido a la creación de un cliente. Luego, solicita y lee los datos que permiten crear un cliente, desplegando un mensaje que informe al usuario el resultado obtenido. Ver Figura 4.
	createBus	Despliega un título apropiado para que el usuario observe que ha accedido a la creación de un bus. Luego, solicita y lee los datos que permiten crear un bus, desplegando un mensaje que informe al usuario el resultado obtenido. Ver Figura 5.

	createViaje	Despliega un título apropiado para que el usuario observe que ha accedido a la creación de un viaje. Luego, solicita y lee los datos que permiten crear un viaje, desplegando un mensaje que informe al usuario el resultado obtenido. Ver Figura 6.
	vendePasajes	Despliega un título apropiado para que el usuario observe que ha accedido a la venta de pasajes a un cliente. Luego, solicita y lee los datos que permiten iniciar una venta. Si los datos iniciales no permiten concretar la venta, se despliega un mensaje que informe al usuario la situación, finalizando la acción. Ver la Figura 7 que muestra el inicio exitoso de una venta. En caso de poder proceder con la venta de pasajes, se solicita y lee los datos necesarios para la venta de cada pasaje que requiera el cliente, esto es el número de pasajes requeridos y la fecha del viaje, procediendo a desplegar datos de todos los viajes que tienen una disponibilidad suficiente de asientos, ver Figura 8, solicitando al usuario seleccione un viaje. A continuación, se despliegan todos los asientos del viaje seleccionado diferenciando los libres de los ocupados y se solicita al usuario ingrese los asientos deseados por el cliente. Por cada asiento elegido, se solicita el tipo de identificador y el identificador del pasajero que utilizará el asiento, se verifica que el pasajero exista, si no es así se solicitan todos los datos necesarios y se crea el pasajero, procediendo a incorporar un nuevo pasaje a la venta. Al concluir el ingreso de los pasajeros, se presenta al usuario el monto total que deberá pagar el cliente, ver la Figura 9. La venta finaliza desplegando al usuario cada pasaje (boleto) que ha comprado, ver Figura 10.
	listPasajerosViaje	Despliega un listado con los datos de los pasajeros de un viaje para la fecha, hora y patente de bus que los realiza, todos los cuales deben leerse. Los datos del listado y su formato se observan en la Figura 11.
	listVentas	Despliega un listado con los datos de las ventas realizadas. Los datos del listado y su formato se observan en la Figura 12.
	listViajes	Despliega un listado con los datos de los viajes registrados en el sistema. Los datos del listado y su formato se observan en la Figura 13.

Tabla 1. Descripción de constructores y métodos de las clases del modelo.