

**EVALUACIÓN NRO 2**

<b>ASIGNATURA</b>	<b>PGY2121 – DESARROLLO DE SOFTWARE DE ESCRITORIO</b>
<b>PROFESOR</b>	
<b>SECCIÓN</b>	
<b>NOMBRE</b>	

<b>PUNTAJE MÁXIMO</b>	<b>48 puntos</b>	<b>NOTA</b>
<b>PUNTAJE OBTENIDO</b>		
<b>FECHA</b>		
<b>DURACIÓN</b>	<b>120 minutos</b>	

**INDICACIONES GENERALES**

La nota 4,0 se obtiene logrando un 60% del puntaje total.

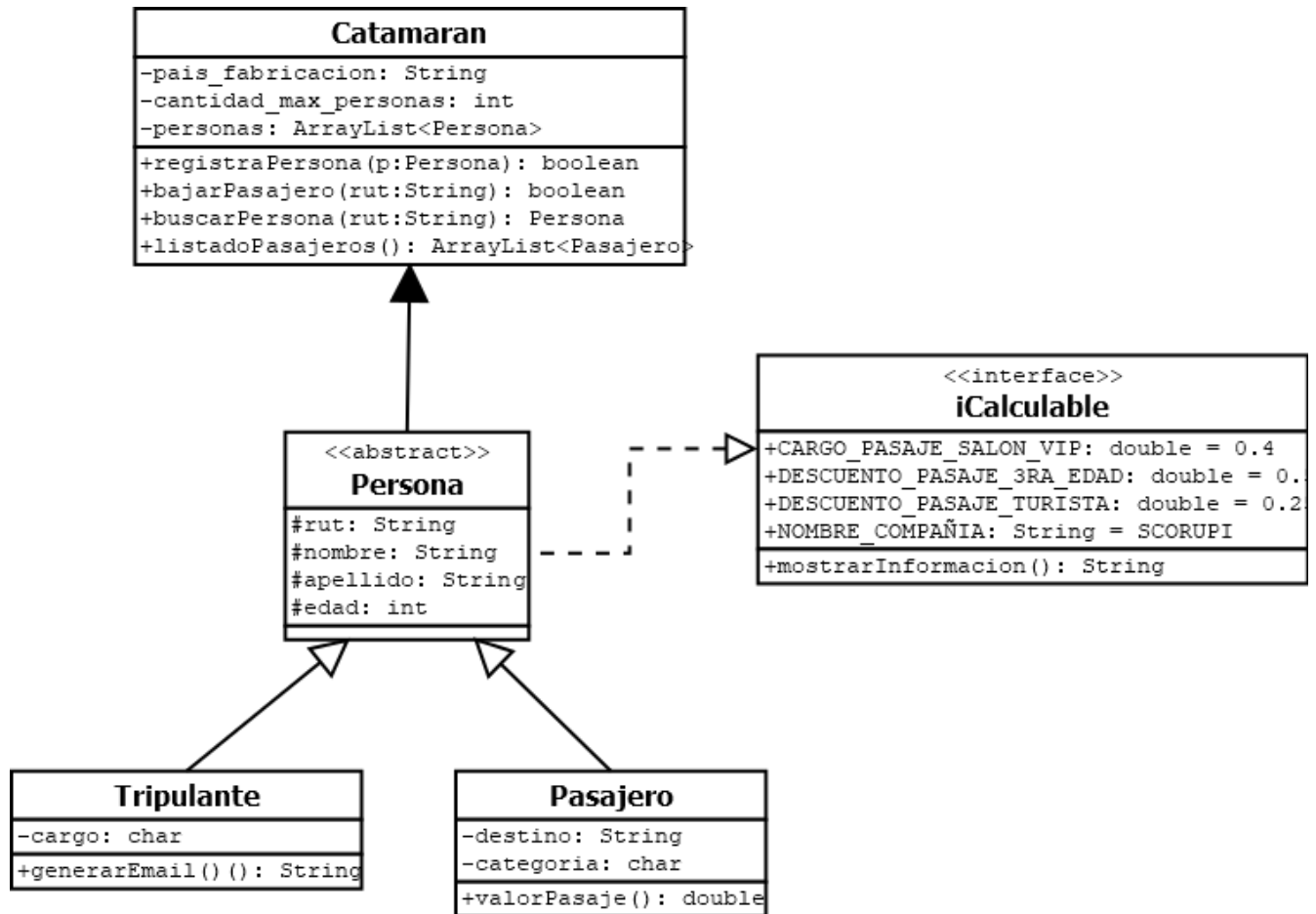
**ENUNCIADO**

Una empresa naviera que transporta a turistas por los lagos del extremo sur necesita crear una aplicación que permita registrar a todos los pasajeros y tripulantes que ingresan a un catamarán antes del comenzar un viaje. Para ello necesitan de todo su apoyo en este proceso ya que servirá para agilizar el proceso de trazabilidad dada la contingencia que aún vive nuestro país por la pandemia causada por el COVID-19.

Los datos que se necesitan registrar de cada persona que sea registrada son: Rut, nombre, apellido y edad. En el caso específico de los tripulantes se pide el cargo el cual puede ser Capitán (C) o Asistente (A), mientras que en el caso de los pasajeros el destino y categoría la cual puede Salón VIP (S) o Normal (N).

La aplicación debe tener la posibilidad de validar que toda persona registrada no pueda ser registrada nuevamente y a su vez la posibilidad de entregar un listado con solo los pasajeros.

A continuación, se presenta el diagrama de clases en el cual puede ver una aproximación de las clases que deberán ser programadas:



Consideraciones generales:

- Cada clase debe tener sus respectivos constructores, sobrecargando los constructores correspondientes.
- Generar accesadores y mutadores para todas las clases.
- La clase Tripulante NO debe permitir la posibilidad de que otra clase pueda heredar de ella.

En el caso de la *Interface* se deberán declarar los valores y firma de los métodos que se indican en el diagrama considerando que tendrán que ser sobrescrito según cada clase.

**mostrarInformacion:** este método debe entregar la siguiente información:

**Tripulante:** Capitán Andrés Bello - a.bello@scorupi.cl

**Pasajero:** 12345678-9 | Esteban Dido 35 años | Pasajero Salón VIP | Destino Laguna San Rafael | Valor Pasaje \$ 420.000

**generarEmail:** este método retorna el email de cada chofer. Para ello considerar el siguiente modelo:

primera letra del nombre + "." + apellido + "@" + nombre\_compañía + ".cl"

**valorPasaje:** este método retorna el valor final del pasaje. Este valor se calculará a partir de un valor base que está dado por el destino de cada pasajero.

- Laguna San Rafael: 500.000
- Lago Todos los Santos: 250.000
- Lago Riñihue: 350.000
- Lago General Carrera: 450.000
- Rio Futaleufú: 400.000
- Cabo de Hornos: 600.000

Al valor según destino se le debe agregar el cargo si la categoría del pasajero es salón VIP o aplicar el descuento asociado a si es mayor de edad o si es turista.

En la clase Catamaran se deben codificar los siguientes métodos:

**registrarPersona:** este método deberá retornar TRUE si la persona no está en el catamaran y puede ser registrada o retornará FALSE si ya está registrada. La búsqueda debe ser según el rut. Debe considerar que no se puede sobrepasar la cantidad de personas permitidas por el catamarán.

**bajarPasajero:** este método retornará TRUE o FALSE según si se bajó o no el PASAJERO indicado. Importante acotar que solo se pueden bajar personas que sean pasajeros. **Los tripulantes no se ven afectados por este método.**

**buscarPersona:** este método debe retornar una persona que debe ser buscada por su rut. Puede ser un pasajero o tripulante el resultado buscado.

**listarPasajeros:** este método debe retornar una colección con todos los pasajeros que hay registrados.

Finalmente, para probar el funcionamiento de su código se requiere que programe una clase que contenga el método MAIN en el cual deberá realizar las siguientes operaciones

1. Crear una Catamaran.
2. Crear al menos 5 Personas entre tripulantes y pasajeros.
3. Registrar a todas las personas.
4. Ejecutar los métodos que permiten buscar, bajar y listar.

Utilice la nomenclatura del lenguaje para la definición de packages, clases, visibilidad de atributos y encapsule su código.

El nombre del proyecto debe ser según la siguiente definición:

Nombre + Apellido + Sección.

Para enviar el proyecto siga las instrucciones que les entregará su profesor.

Éxitos muchachos!!!!