

INSTITUTO TECNOLÓGICO DE OAXACA

CLUB DE PROGRAMACIÓN

NOMBRE: MIGUEL ANGEL GOMEZ ROMERO

CARRERA: ING. SISTEMAS COMPUTACIONALES

GRUPO: 2SC

SEMESTRE: SEGUNDO

HORARIO: 15:00-17:00

FECHA DE ENTREGA: 20 DE MAYO DEL 2020.

**REPORTE DE PROYECTO FINAL DE CLUB DE PROGRAMACION**

Planteamiento del Problema:

Una empresa quiere registrar los diferentes tipos de transporte que ingresan a su estacionamiento y para ello se quiere crear mediante el uso de la programación orientada a objetos aplicando el tema de herencia. Los datos que la empresa quiere obtener es Marca, Modelo, Placas y Capacidad. (**Nota:** en la empresa se tiene conocido que generalmente ingresan mayormente automóviles y autobuses que transportan al personal).

Defina las clases con sus relaciones de herencia de la manera más apropiada para modelar este sistema e impleméntelas a través de un proyecto de formas como la siguiente:

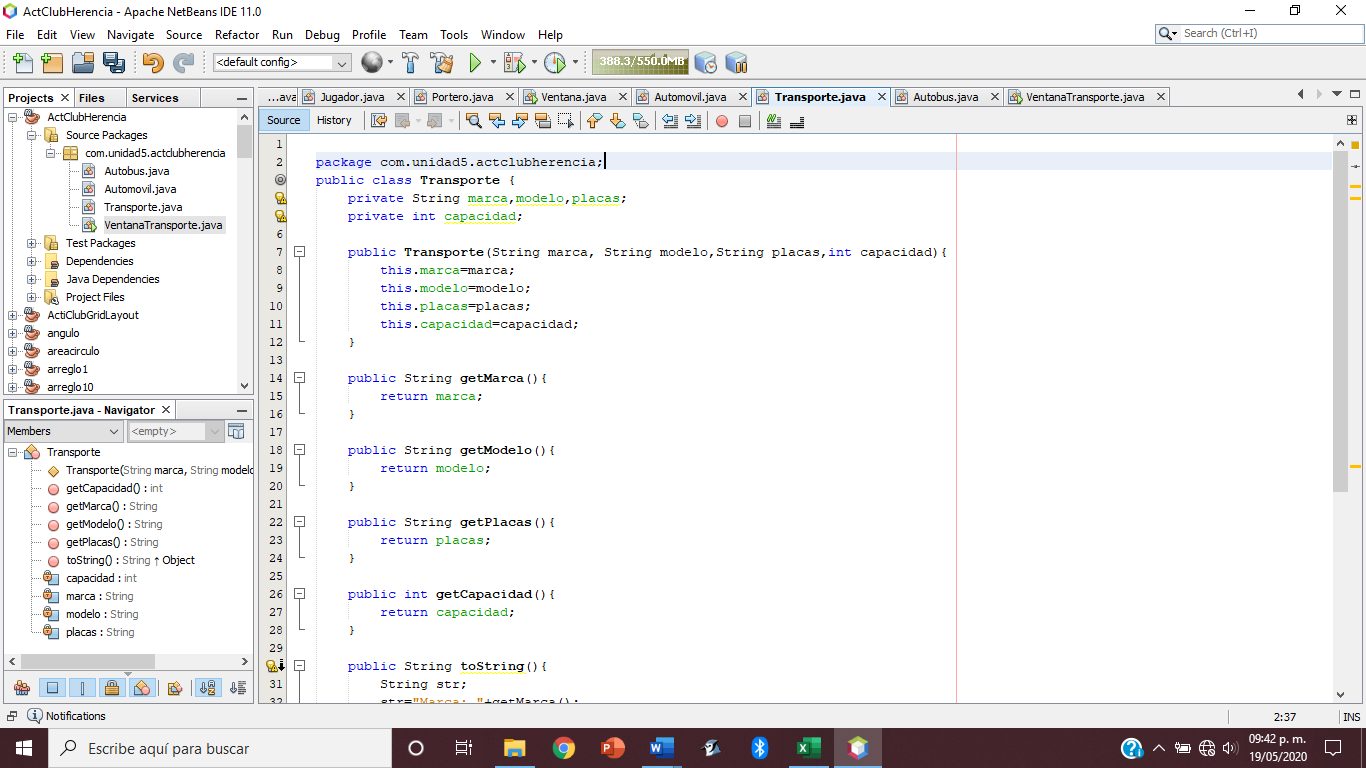
**Especificaciones:**

* Cuando el usuario seleccione automóvil o autobús se deben de pedir los datos correspondientes para cada uno.
* Instanciar cada una de las clases, capture sus datos para mostrarlos posteriormente.
* Implementar la sobrescrita del método ToString() para mostrar los datos de los diferentes tipos de transporte.
* Se requiere guardar los datos de cada transporte que se vayan agregando en un archivo de texto.

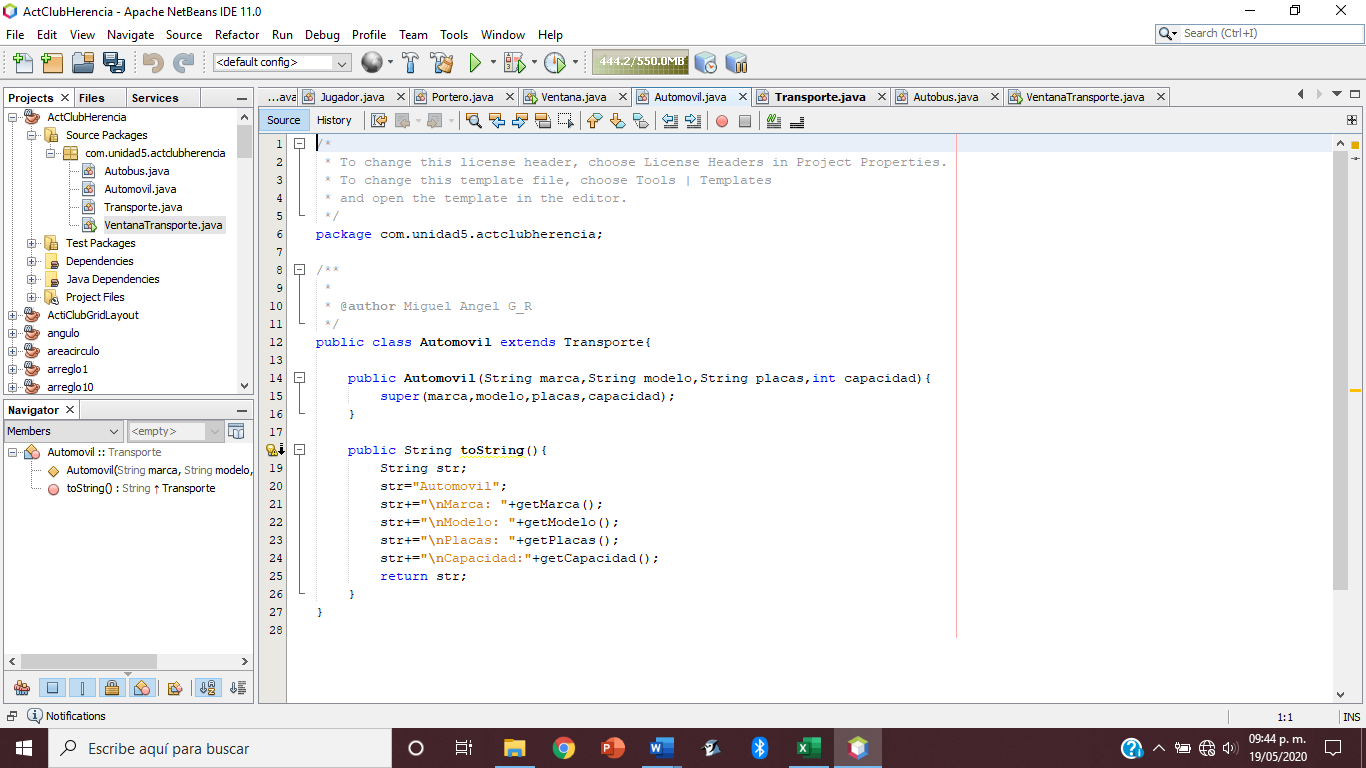
DISEÑO DE LA SOLUCIÓN

Lo primero fue crear mi clase padre o super clase la cual se llame “Transporte”, de la cual salen las subclases o clases hijas “Automóvil” y “Autobús”, las cuales heredaron tanto los atributos como los métodos que contenía la clase padre.

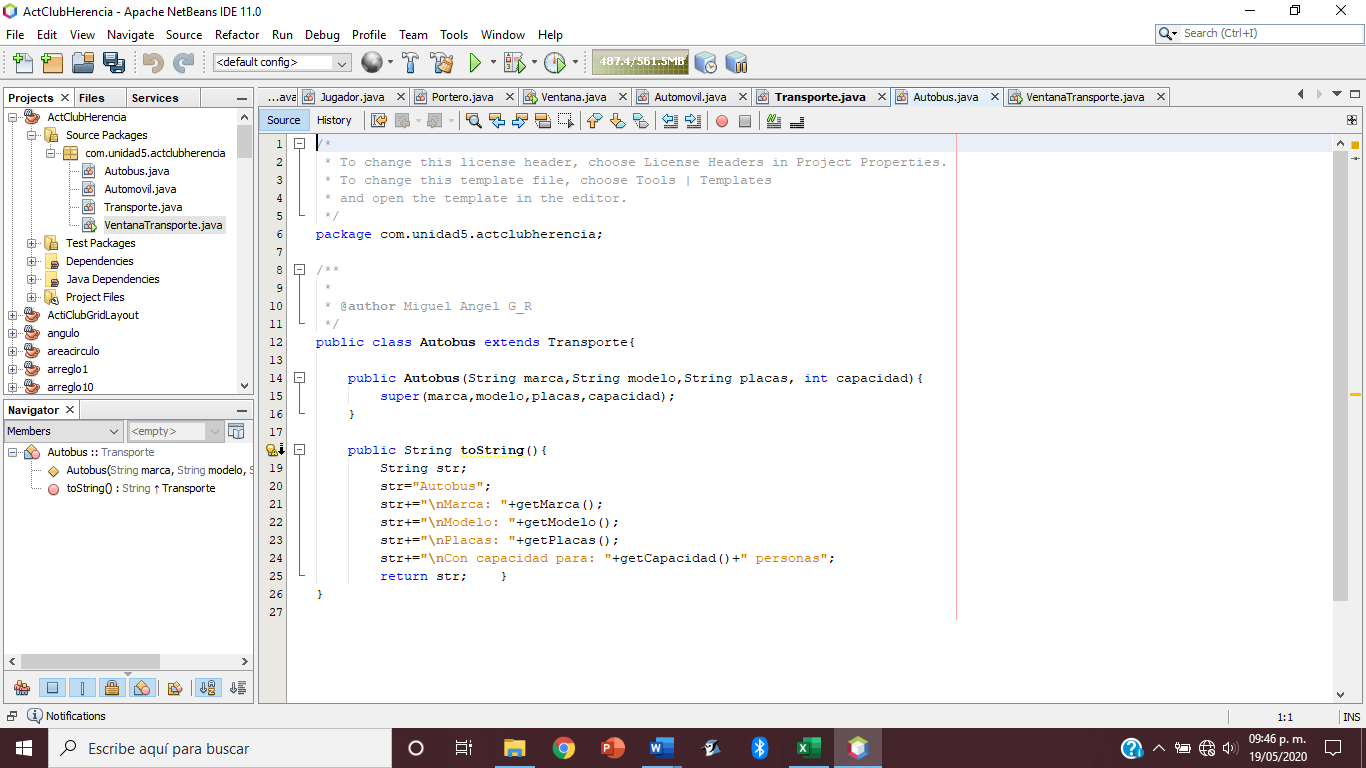
Lo siguiente fue crear una interfaz la cual se adaptaría a mis clases, y así poder crear nuevos objetos de dichas clases, y también se crearía un archivo de texto lo cual permitiría que se guarden los datos de los objetos creados.



Aquí se puede ver la clase Transporte con sus atributos y métodos correspondientes.



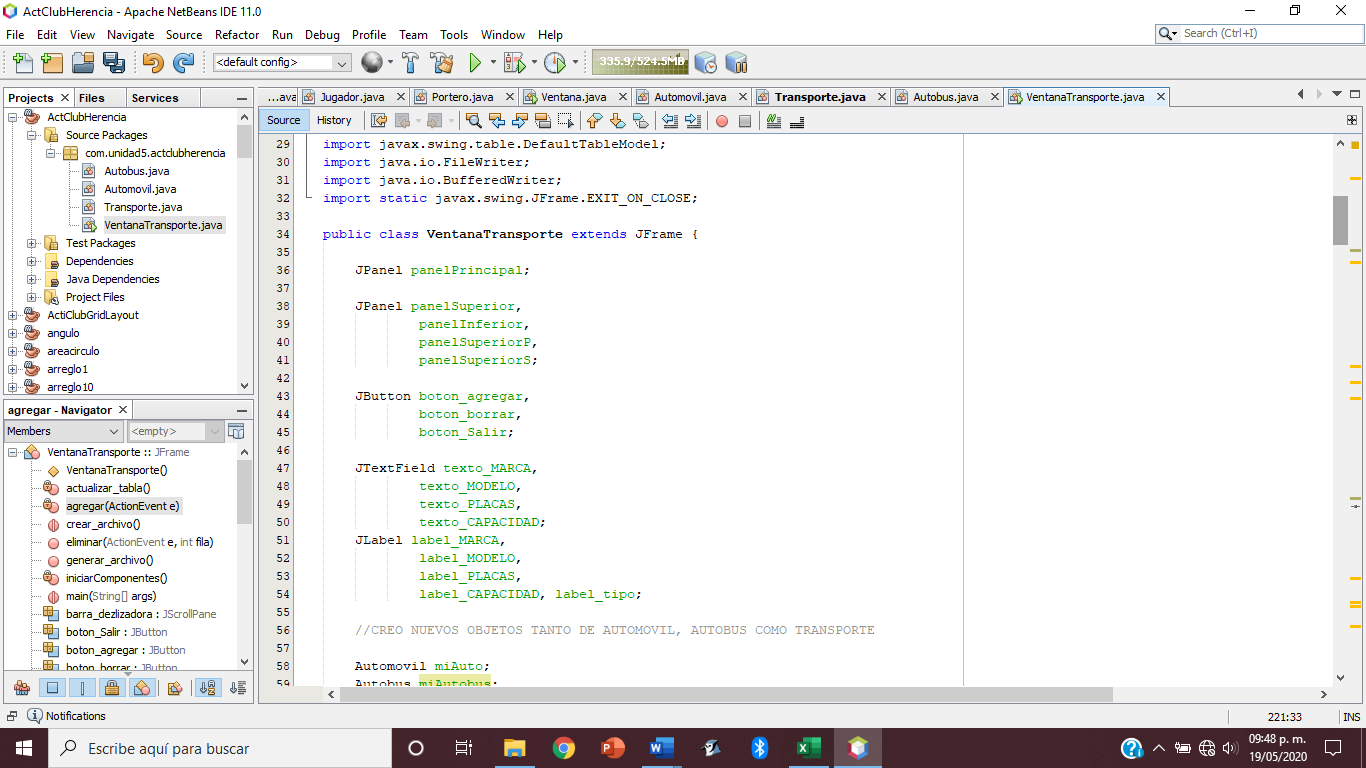
La primera subclase es **Automóvil,** en la cual con la palabra reservada **super** estamos haciendo el llamado de herencia de la clase Transporte.



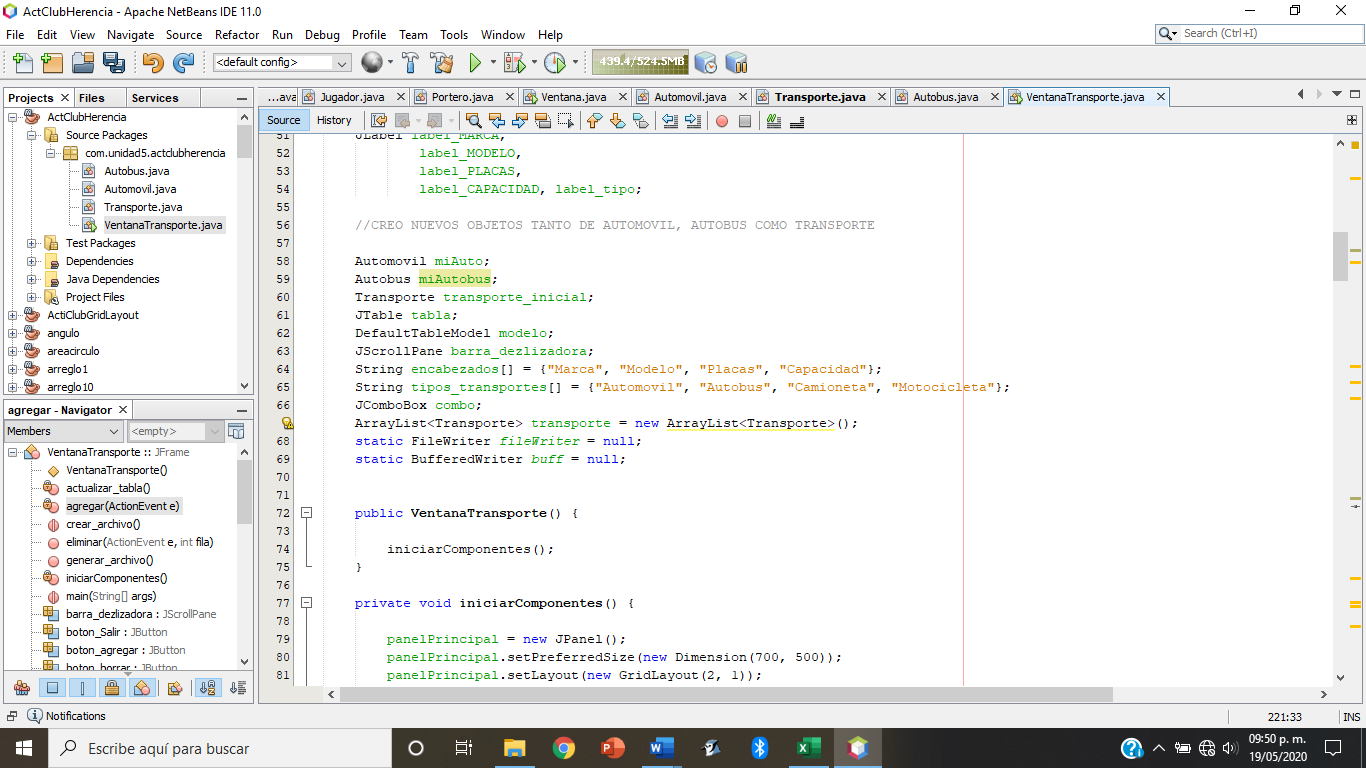
De igual manera en la segunda subclase **Autobús**, utilizamos la palabra reservada **super** para la herencia de la clase Transporte.

INTERFAZ VENTANATRANSPORTE

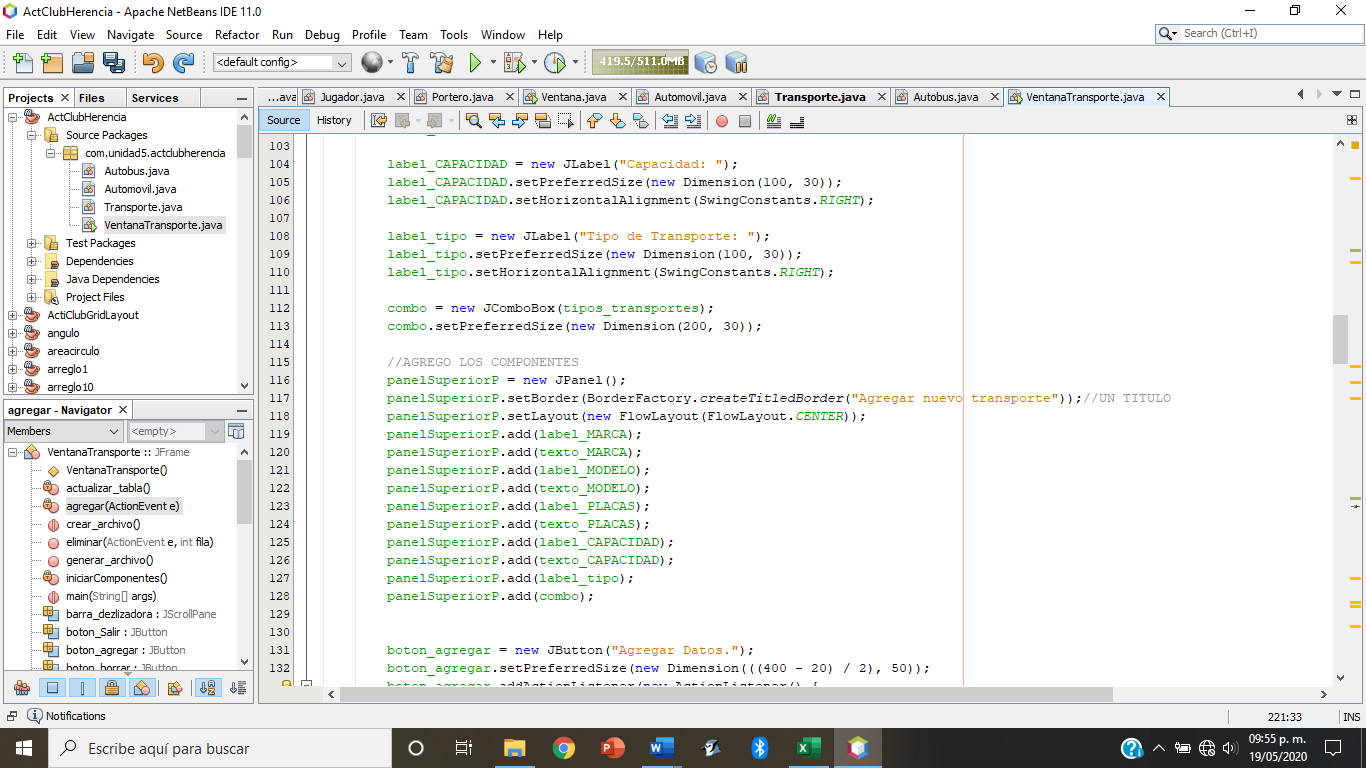
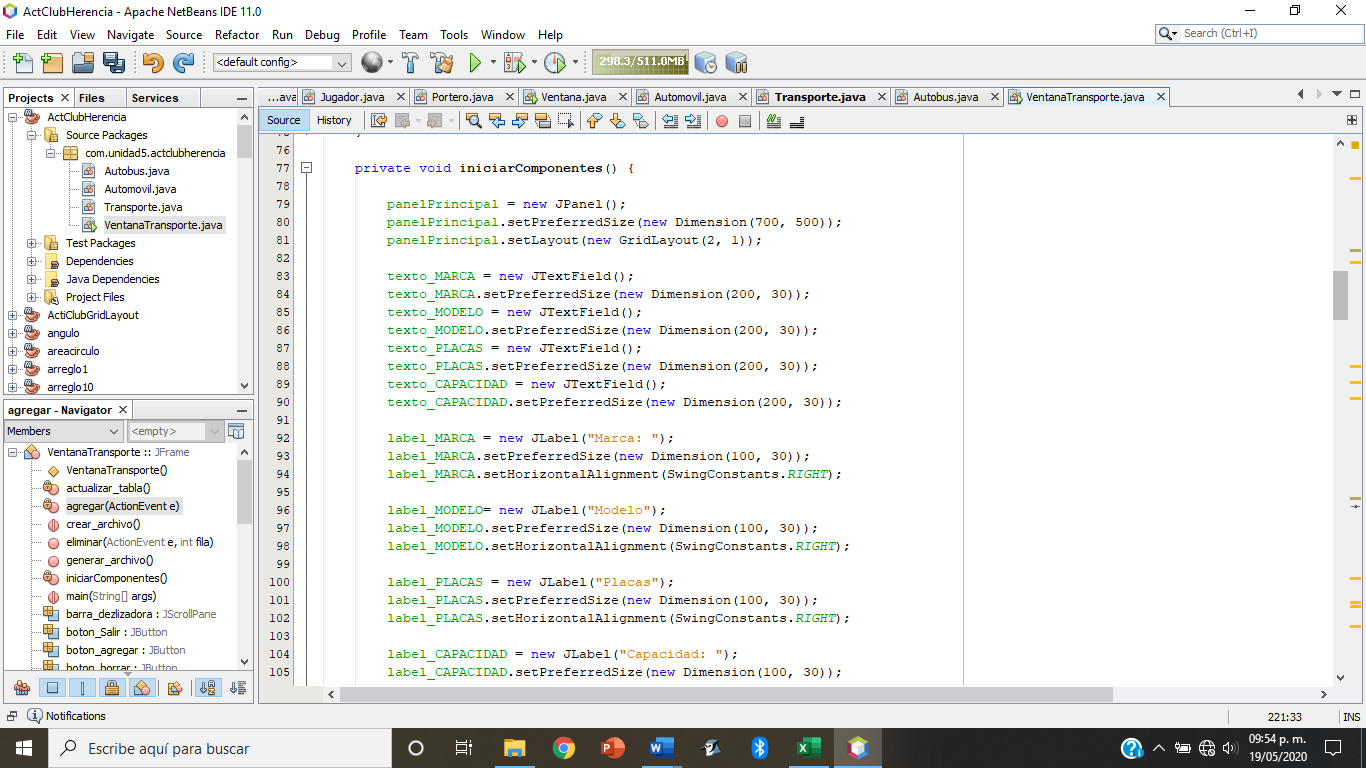
Cree todos los componentes necesarios para mi interfaz:



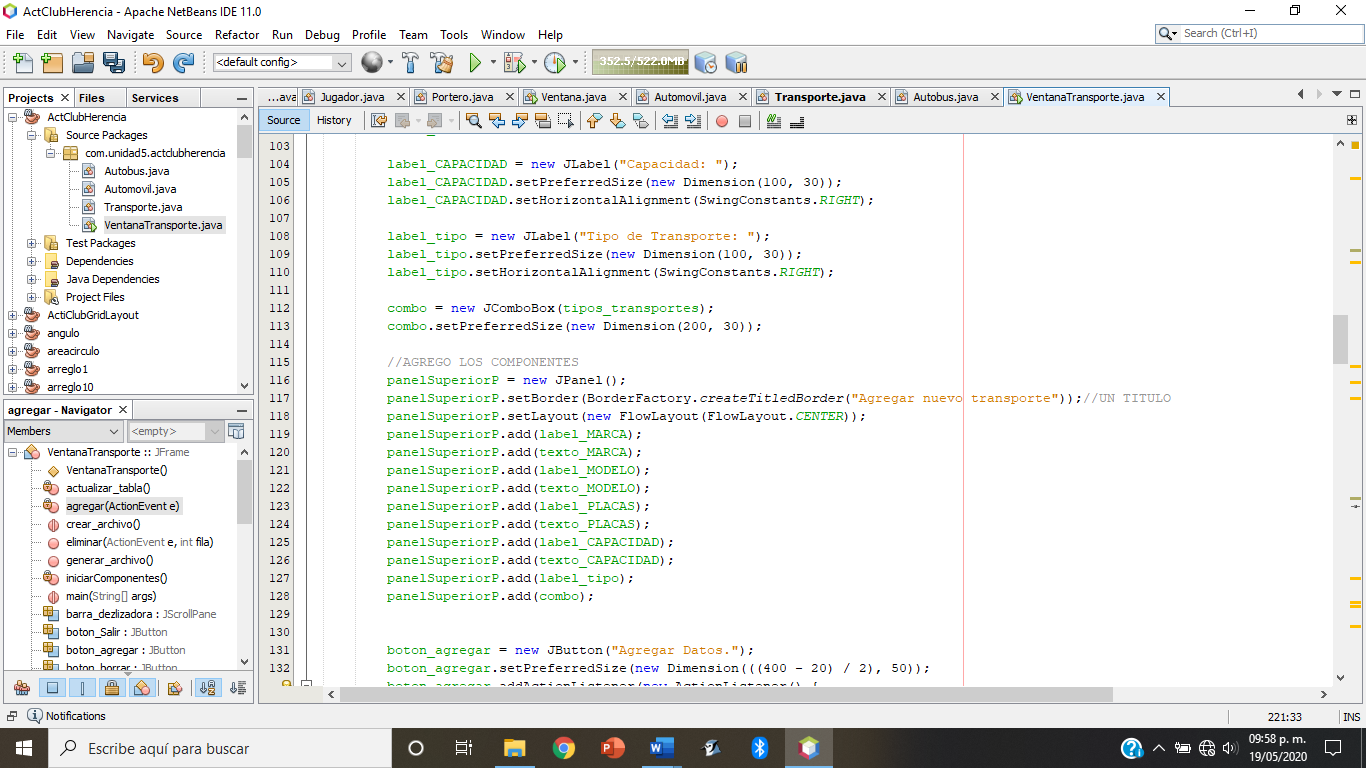
Ahora cree mis objetos de cada clase dependiendo del tipo de transporte y la clase padre, además de un arreglo el cual agregara cada objeto que se cree.



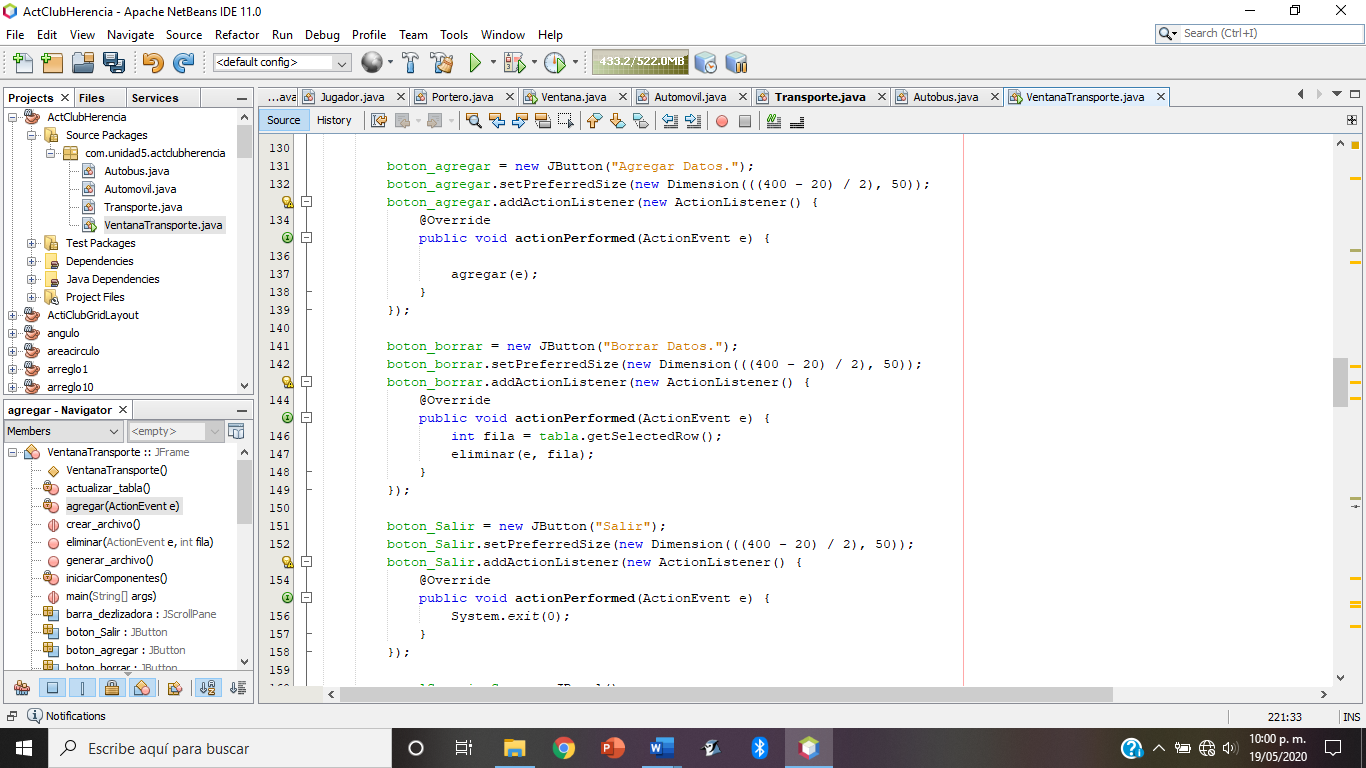
Inicialice cada componente como sus dimensiones, en los label el nombre correspondiente con el cual se visualizarían



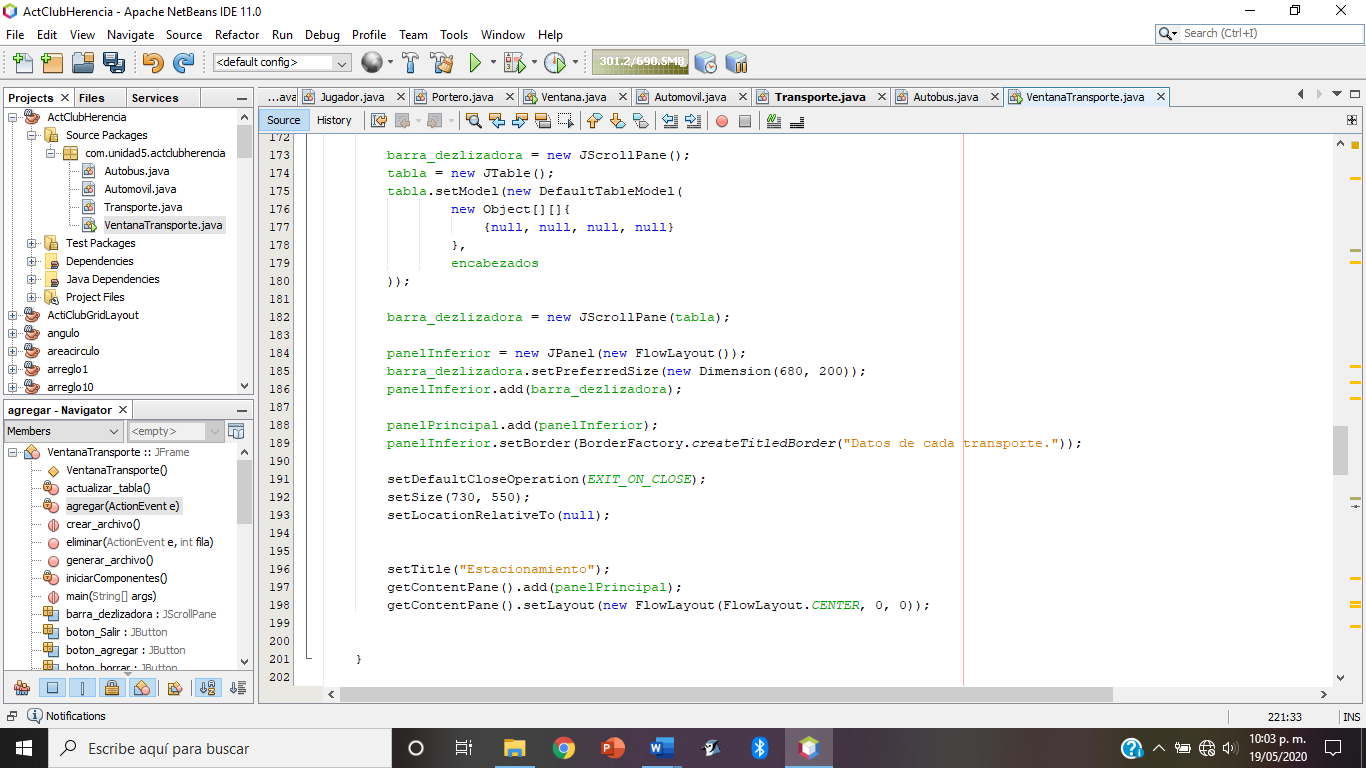
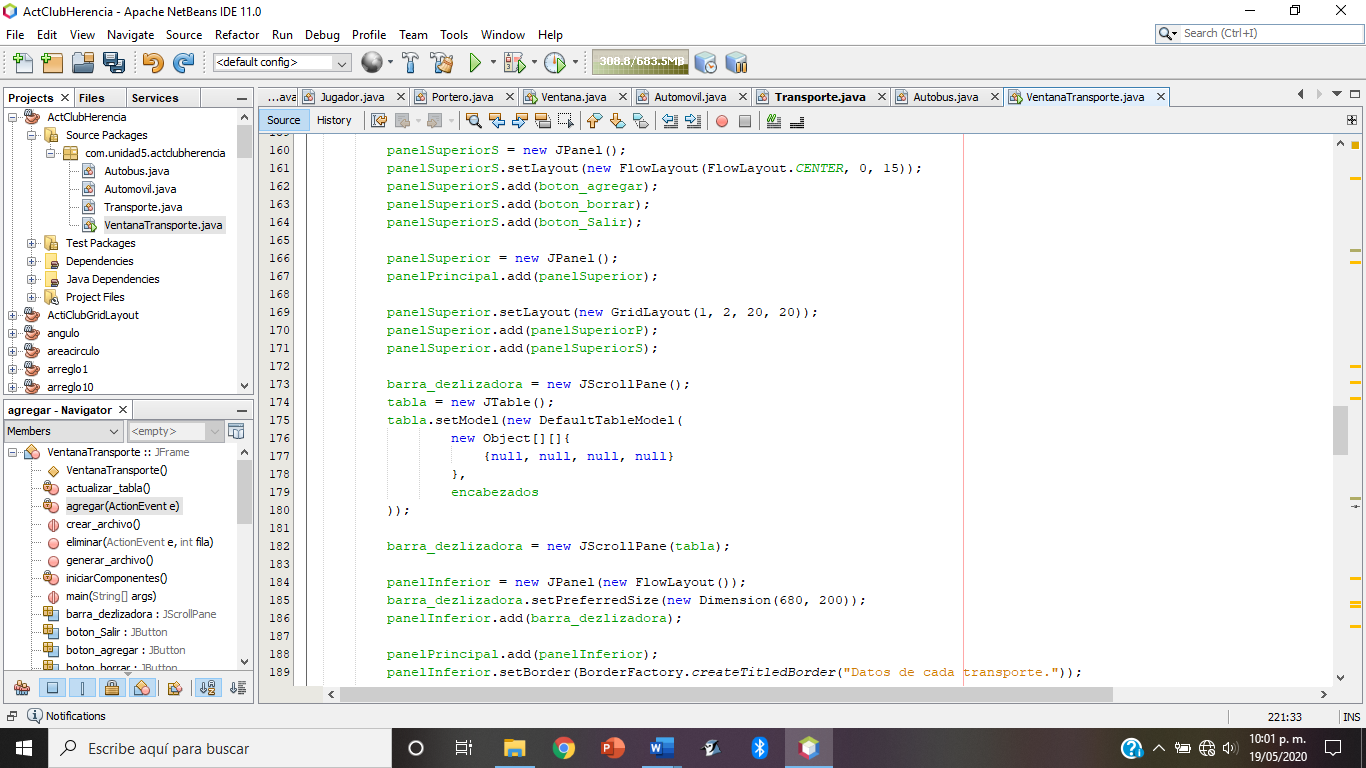
Agregamos los componentes en el JPanel()



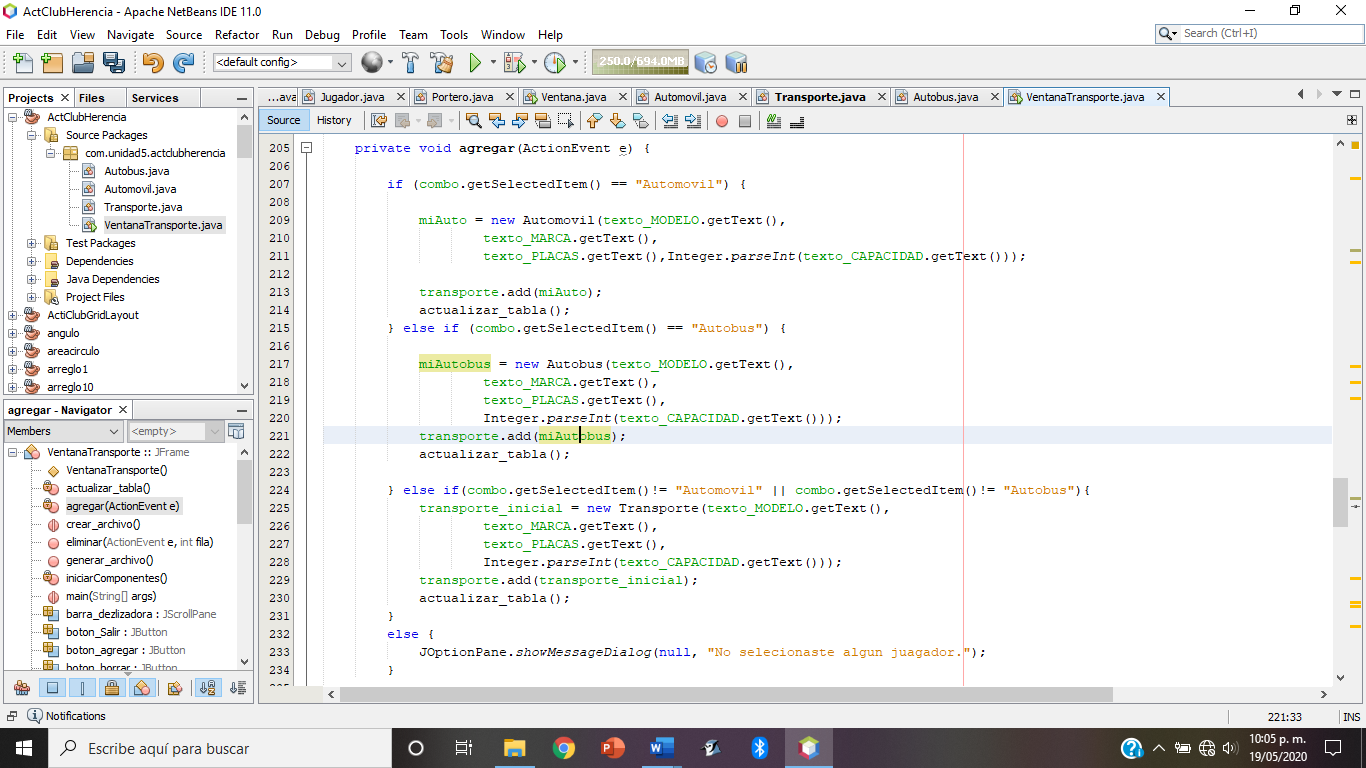
Agregamos las funciones de los botones Agregar Datos, Borrar Datos y Salir, al momento de hacer clic.



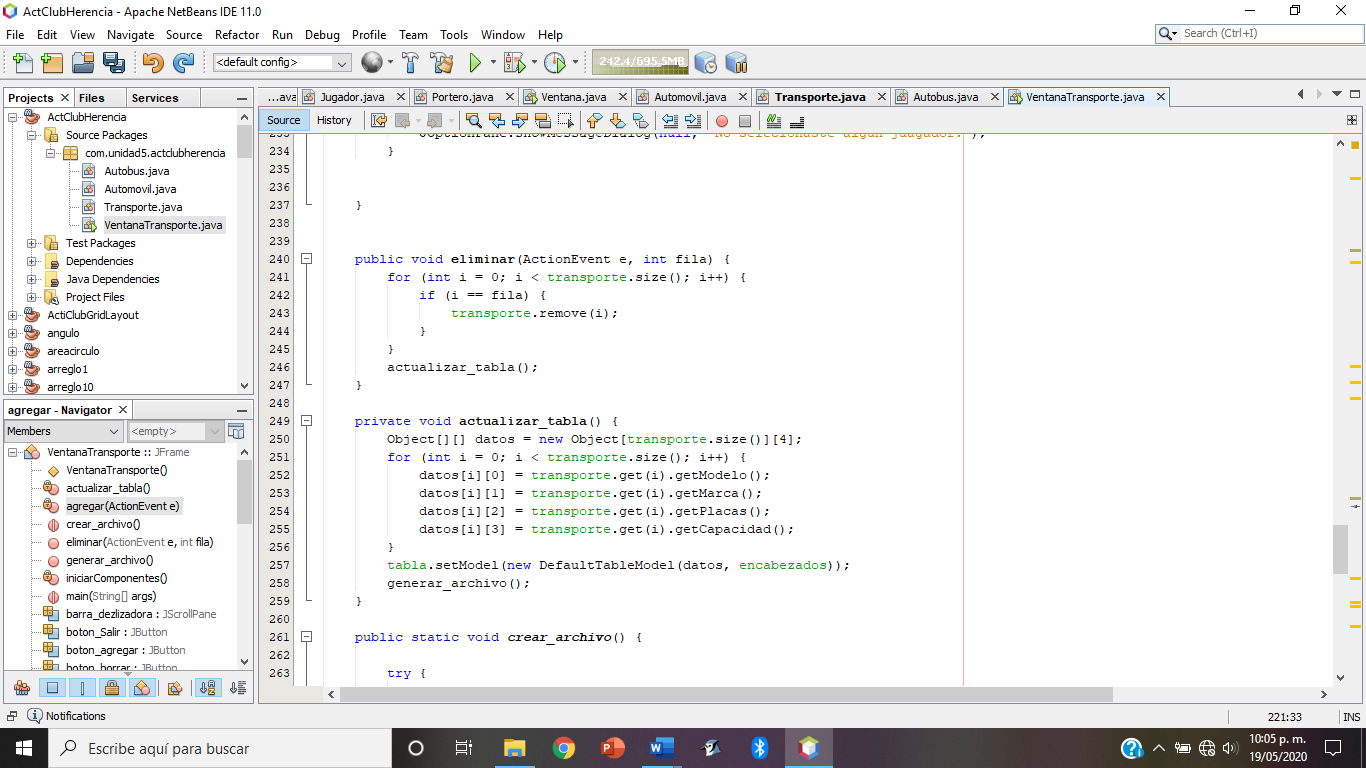
Agregamos los últimos componentes de la interfaz.



En el método agregar, será para cuando se seleccione que tipo de transporte crear, este los agrege al arreglo y los guarde mediante el objeto correspondiente.

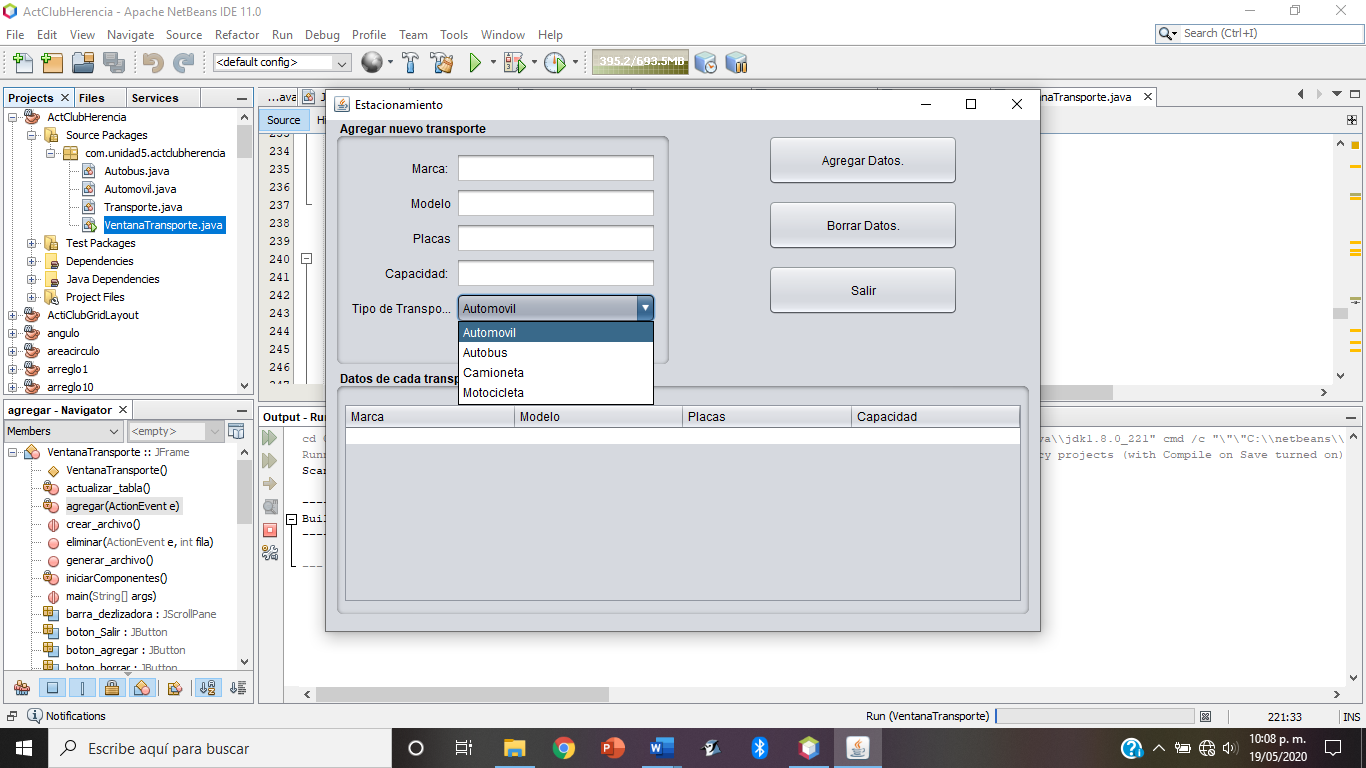


Y por último los métodos de Eliminar y Actualizar tabla, esto para cuando queramos eliminar un objeto del arreglo o querer visualizar al momento de crear uno nuevo.



NIVEL GRÁFICO

Se propuso diseñar una Interfaz Gráfica de Usuario como la siguiente:



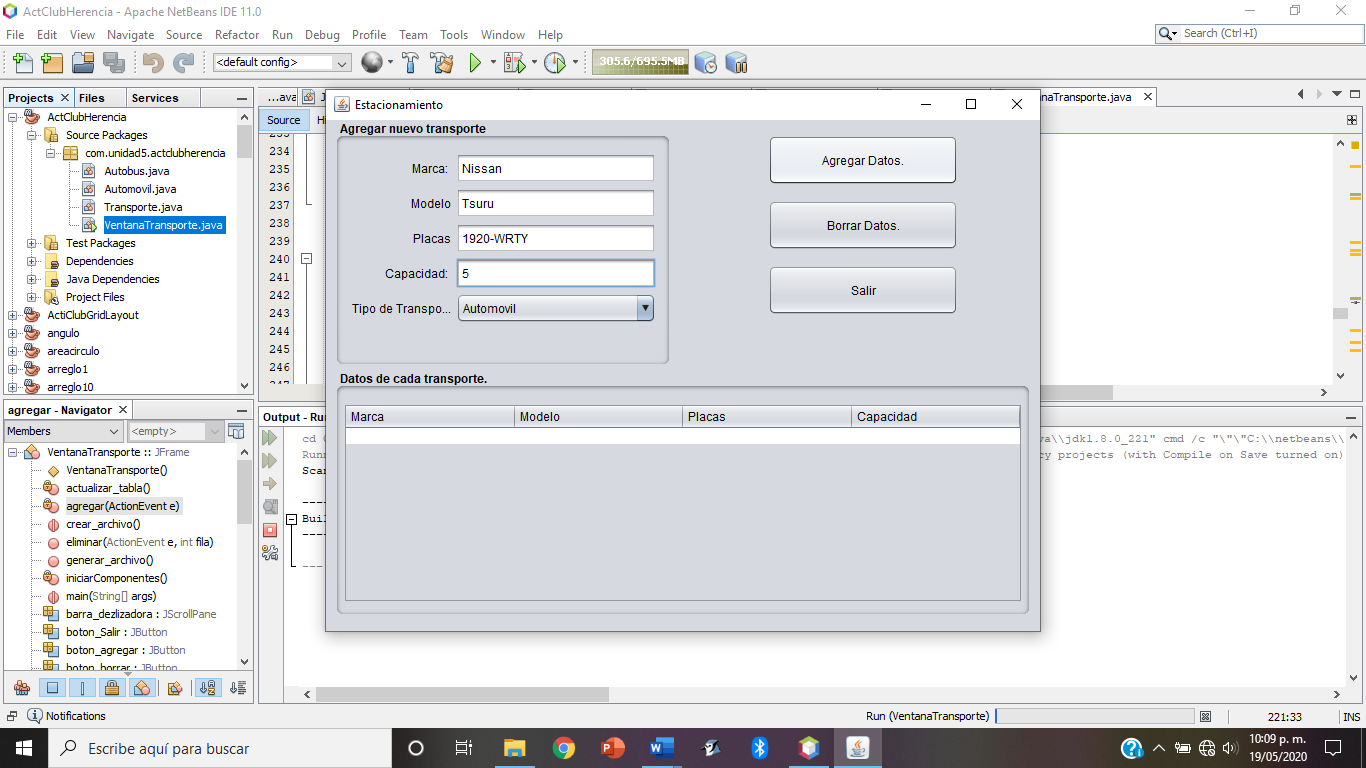
**Herramientas a usar:**

* Netbeans IDE 8.2
* Lenguaje Java
* Clase FileWritter para manejo de Archivos
* Clase BufferedWritter para escribir sobre archivos
* Componentes gráficos de la Liberia swing de java
* Diseño de Clases
* Uso de Herencia
* Administradores de Distribuciones en GUI

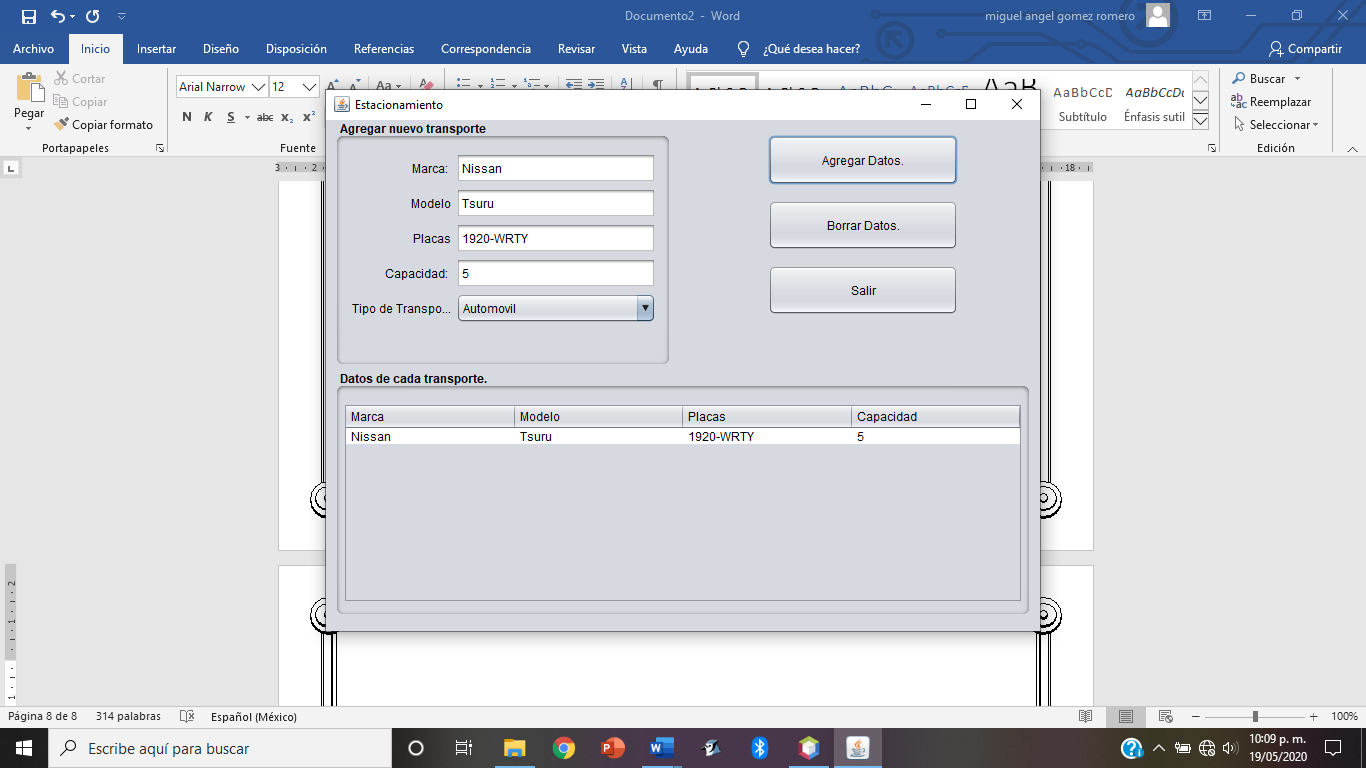
|  |  |
| --- | --- |
| JLabel | texto\_MARCA, texto\_MODELO, texto\_PLACAS, texto\_CAPACIDAD. |
| JTextField | **label\_MARCA, label\_MODELO, label\_PLACAS, label\_CAPACIDAD, label\_tipo;** |
| JPanel | **panelPrincipal, panelSuperior, panelInferior, panelSuperiorP, panelSuperiorS.** |
| JButton | **boton\_agregar, boton\_borrar, boton\_Salir.** |
| JTable | **tabla** |
| DefaultTableModel | **modelo** |
| JScrollPane | **barra\_dezlizadora** |
| JComboBox | **combo** |
| ArrayList<Transporte> | **transporte** |

Pruebas e Implementación

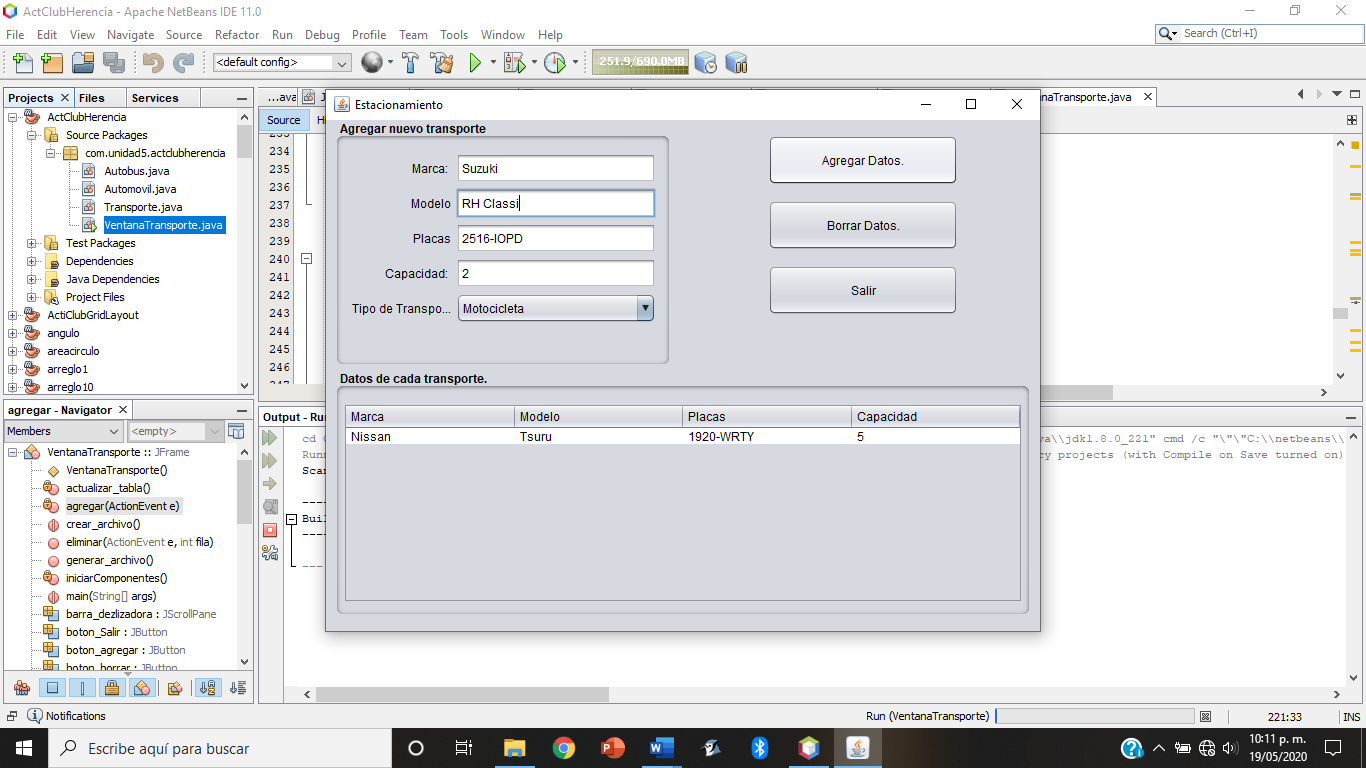
CREAMOS UN NUEVO AUTOMOVIL



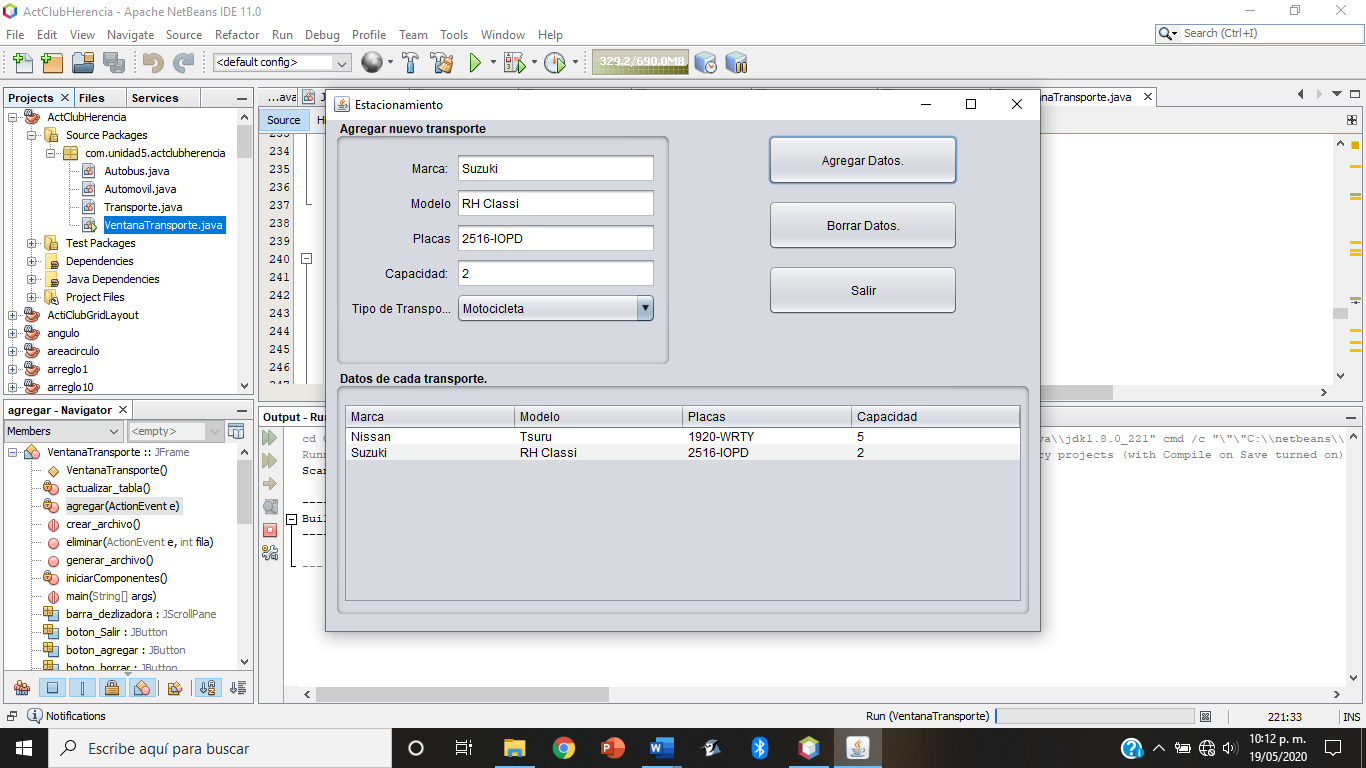
RELLENAMOS LOS CAMPOS CORRESPONDIENTES Y CLIC EN EL BOTON AGREGAR DATOS



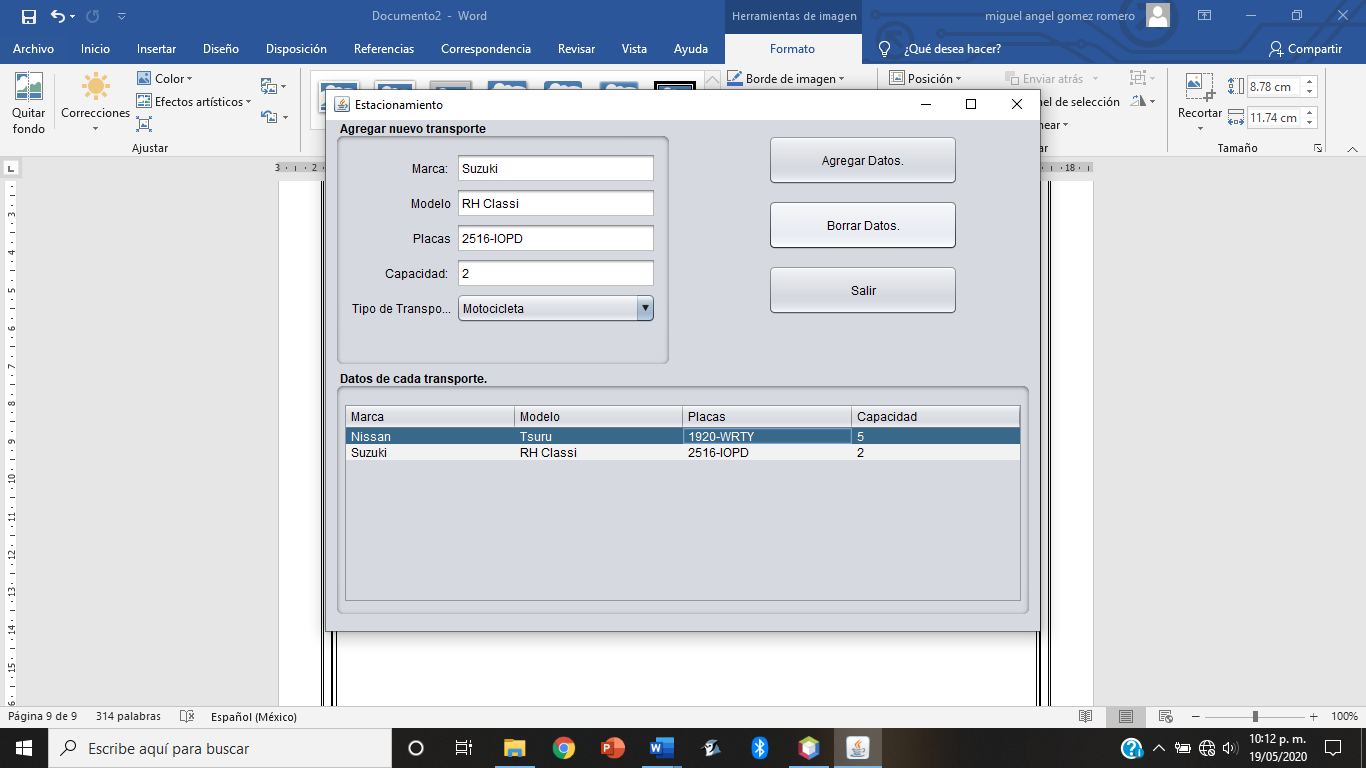
Y NOS APARECERA EN EL TABLA DE ABAJO LOS DATOS DEL AUTOMOVIL



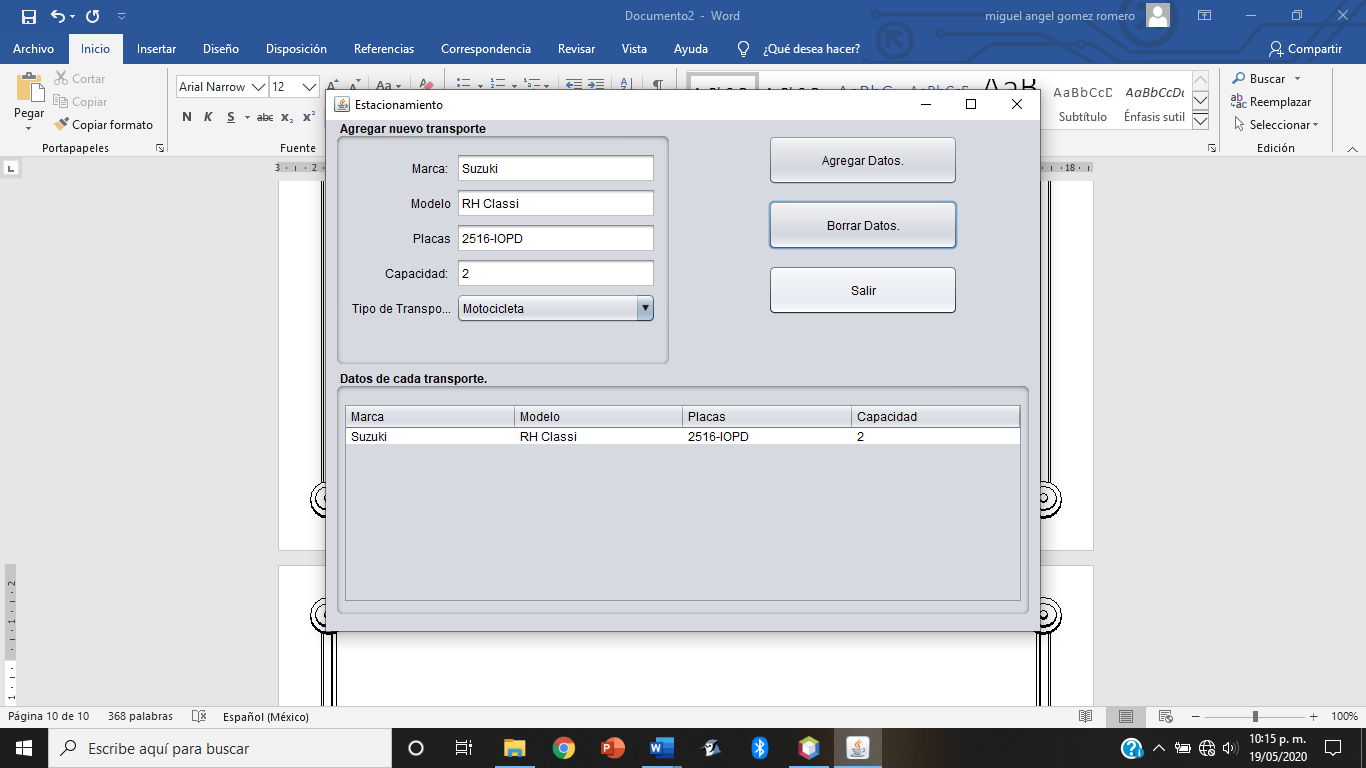
AHORA AGREGAREMOS OTRO DE TIPO MOTOCICLETA



Y AHORA TENDREMOS DOS TRANSPORTES CREADOS.



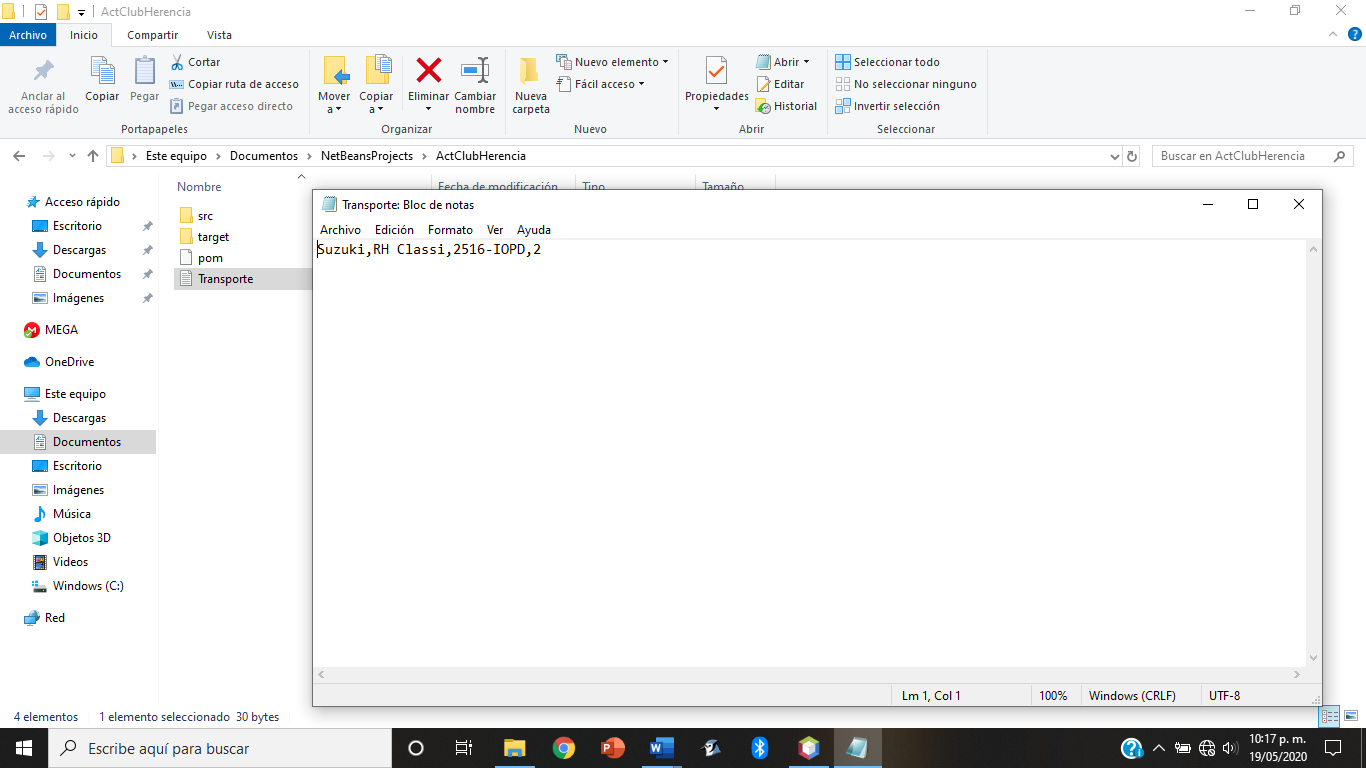
PARA ELIMINAR UN OBJETO, LO SELECCIONAMOS Y LE DAMOS CLIC EN EL BOTON BORRAR DATOS



Y PODEMOS CREAR TANTOS TIPOS DE TRANSPORTE ASI COMO ELIMINARLOS.

**PARA SALIR DEL PROGRAMA SOLO BASTA CON DARLE CLIC EN EL BOTON SALIR.**

ARCHIVO DE TEXTO



**COMO ELIMINAMOS UNO, SOLO NOS APRECE EL TIPO DE TRANSPORTE MOTOCICLETA CON LOS DATOS QUE AGREGAMOS.**