**ANÁLISIS DEL PROBLEMA – TAREA INTEGRADORA III**

**Juan Jose Diaz Parra - A00381098**

**Requerimientos del problema:**

* **R1:** Registrar vehículos (nuevos o usados) para la venta.
* **R2:** Calcular el precio total de venta de un vehículo, pero este mecanismo variará de acuerdo al tipo del vehículo. Si el vehículo cuenta con la documentación vencida (es decir que dentro de los documentos asociados al vehículo no haya SOAT y certificado de revisión técnico-mecánica del año en curso) se le debe cobrar $500,000 adicionales. Los autos eléctricos cuestan un 20% adicional de su precio base, los híbridos un 15%, y a los de gasolina no se les varía nada. Si el automóvil es usado tiene un descuento del 10%. Las motocicletas, por la alta demanda que tienen en el país, tienen un costo adicional de 4%, si son usadas tienen un descuento del 2%. A los cálculos ya mencionados, el concesionario puede descontarle un porcentaje adicional.
* **R3**: Generar informes con todos los datos de los vehículos (incluido el precio total de venta) de acuerdo a los siguientes criterios:
  + Tipo de vehículo
  + Tipo de combustible
  + Autos nuevos/usados
* **R4:** Dado el id de un vehículo, se debe mostrar el estado de sus documentos y generar un listado con los números de SOAT, revisión técnico-mecánica y tarjeta de propiedad (si la tiene)
* **R5**: Generar un mapa del parqueadero que permita identificar los espacios vacíos y ocupados.
* **R6**: Generar informes de la ocupación del parqueadero de acuerdo con los siguientes criterios:
  + Listado de vehículos (y su información) dado un rango de años.
  + Datos del vehículo más antiguo y más nuevo.
  + Porcentaje de ocupación del parqueadero.

**Análisis de requerimientos:**

**Requerimiento 1:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R1:** El usuario debe poder registrar vehículos (nuevos o usados) para la venta. | | |
| **Resumen** | El usuario debe poder registrar un vehículo con sus respectivos atributos y sus documentos. Asimismo, al final del proceso se debe comprobar si se puede añadir el vehículo creado al parqueadero. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Tipo de vehículo a registrar | int |  |
| Precio base | double |  |
| Marca | String |  |
| Modelo | Int |  |
| Cilindraje | double |  |
| kilometraje | double |  |
| Tipo del vehículo | int |  |
| Placa del vehículo | String | Solo se lee si el vehículo es usado |
| Numero de puertas | int | Solo se lee si se está registrando un carro |
| Polarizado de ventanas | int | Solo se lee si se está registrando un carro |
| Tipo de carro | Int | Solo si se está registrando un carro |
| Tipo de motocicleta | int | Solo se lee si se está registrando una motocicleta |
| Duración de la batería | double | Solo se lee si esta registrando un carro eléctrico o un carro hibrido |
| Tipo de cargador | int | Solo se lee si se está registrando un carro eléctrico o un carro hibrido |
| Capacidad del tanque | double | Solo se lee si es un carro a gasolina, hibrido o motocicleta |
| Tipo de gasolina | double | Solo se lee si es un carro a gasolina |
| Precio del SOAT | double | Solo si un carro es usado |
| Año del SOAT | double | Solo si el carro es usado |
| Cobertura del SOAT | double | Solo si el carro es usado |
| Precio de la técnica mecánica | double | Solo si un carro es usado |
| Año de la técnica mecánica | double | Solo si el carro es usado |
| Nivel de gases de la técnica mecánica | double | Solo si el carro es usado |
| Precio de la tarjeta de propiedad | double | Solo si un carro es usado |
| Año de la tarjeta de propiedad | double | Solo si el carro es usado |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el método addVehicle() en la clase controladora, pasando como parámetros los datos que se ingresaron. Dentro de este se ejecuta el metodo constructor de el vehículo que se ingresó. Además, se crean los documentos de acuerdo con las opciones del usuario y se añaden al vehículo. Finalmente se ejecuta el metodo addVehicleToParkingLot() Para ver si se puede añadir el vehículo al parqueadero | | |
| **Resultado o postcondición** | Se agrega un nuevo vehículo al sistema con sus respectivos documentos y si es el caso se agrega a el parqueadero | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Estado del proceso | String |  |

**Requerimiento 2:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R2:** El sistema debe poder calcular el precio total de venta de un vehículo | | |
| **Resumen** | Calcular el precio total de venta de un vehículo, pero este mecanismo variará de acuerdo al tipo del vehículo. Si el vehículo cuenta con la documentación vencida (es decir que dentro de los documentos asociados al vehículo no haya SOAT y certificado de revisión técnico-mecánica del año en curso) se le debe cobrar $500,000 adicionales. Los autos eléctricos cuestan un 20% adicional de su precio base, los híbridos un 15%, y a los de gasolina no se les varía nada. Si el automóvil es usado tiene un descuento del 10%. Las motocicletas, por la alta demanda que tienen en el país, tienen un costo adicional de 4%, si son usadas tienen un descuento del 2%. A los cálculos ya mencionados, el concesionario puede descontarle un porcentaje adicional. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Id del vehículo | int |  |
| **Actividades generales necesarias para obtener los resultados** | Como cuando se crea el vehículo se calcula su precio de venta, en este requerimiento se ejecuta el metodo getSalePriceOfVehicle() que busca en arraylist de vehículos y obtiene el precio de venta del vehículo y lo imprime en pantalla | | |
| **Resultado o postcondición** | Se muestra en pantalla el precio de venta del vehículo con el id ingresado | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Precio total del vehículo | String | Se muestra si el vehículo existe |

**Requerimiento 3:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R3:** El sistema debe poder generar informes con todos los datos de los vehículos | | |
| **Resumen** | El sistema debe generar informes con todos los datos de los vehículos (incluido el precio total de venta) de acuerdo con los siguientes criterios: tipo de vehículo, tipo de combustible, autos nuevos/usados | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Tipo de criterio a buscar | int |  |
| Tipo de vehículo | int | Se lee solo si el usuario elige el criterio del tipo de vehículo |
| Tipo de combustible | int | Se lee solo si el usuario elige el criterio del tipo de combustible |
| Autos nuevos/Usados | int | Se lee solo si el usuario elige el criterio de autos nuevos/usados |
| **Actividades generales necesarias para obtener los resultados** | Dependiendo de la opción que elija el usuario se va a ejecutar el metodo showInformationByTypeOfVehicle(), showInformationTypeOfFuel(), showInformationUsedOrNew() que van a buscar en el arraylist de vehículos dependiendo del criterio de búsqueda seleccionado | | |
| **Resultado o postcondición** | Se muestra en pantalla la información de los vehículos según el criterio de búsqueda ingresado por el usuario | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Información de los vehículos | String | Se muestra si existe algún vehículo con esas caracteristicas |

**Requerimiento 4:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R4:** El sistema debe poder Generar informes con todos los datos de los vehículos de acuerdo a ciertos criterios | | |
| **Resumen** | El sistema debe generar informes con todos los datos de los vehículos (incluido el precio total de venta) de acuerdo a los siguientes criterios: tipo de vehículo, tipo de combustible, autos nuevos/usados | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Tipo de criterio a buscar | int |  |
| Tipo de vehículo | int | Se lee solo si el usuario elige el criterio del tipo de vehículo |
| Tipo de combustible | int | Se lee solo si el usuario elige el criterio del tipo de combustible |
| Autos nuevos/Usados | int | Se lee solo si el usuario elige el criterio de autos nuevos/usados |
| **Actividades generales necesarias para obtener los resultados** | Dependiendo de la opción que elija el usuario se va a ejecutar el metodo showInformationByTypeOfVehicle(), showInformationTypeOfFuel(), showInformationUsedOrNew() que van a buscar en el arraylist de vehículos dependiendo del criterio de búsqueda seleccionado | | |
| **Resultado o postcondición** | Se muestra en pantalla la información de los vehículos según el criterio de búsqueda ingresado por el usuario | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Información de los vehículos | String | Se muestra si existe algún vehículo con esas características |

**Requerimiento 5:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R5:** Generar un mapa del parqueadero que permita identificar los espacios vacíos y ocupados. | | |
| **Resumen** | El sistema debe generar un mapa del parqueadero donde se guardan los vehículos con modelo inferiores o iguales a 2014 y que sean usados. En este dibujo se debe mostrar si un espacio esta ocupado o esta vacío. Además, a manera de bonus se debe mostrar que tipo de vehículo esta en una casilla ocupada | | |
| **Actividades generales necesarias para obtener los resultados** | Se va a ejecutar el metodo showParkingLot() que hace un recorrido general en la matriz de vehículos que hay en el parqueadero y dependiendo si la celda esta llena o vacía y dependiendo del tipo de vehículo que hay en la casilla va a generar un String con el dibujo del parqueadero | | |
| **Resultado o postcondición** | Se muestra en pantalla el dibujo del parqueadero del concesionario | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Dibujo del parqueadero | String |  |

**Requerimiento 6:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R6:** El sistema debe poder generar informes de la ocupación del parqueadero de acuerdo con ciertos criterios | | |
| **Resumen** | El sistema debe poder generar informes de la ocupación del parqueadero de acuerdo con los siguientes criterios: listado de vehículos (y su información) dado un rango de años, Datos del vehículo más antiguo y más nuevo, porcentaje de ocupación del parqueadero. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Tipo de criterio a buscar | int |  |
| Año del vehículo | int | Se lee solo si el usuario elige el criterio del rango de años |
| Mas antiguo/ más nuevo | int | Se lee solo si el usuario elige el criterio del del vehículo más antiguo y nuevo |
| **Actividades generales necesarias para obtener los resultados** | Dependiendo de la opción que elija el usuario se va a ejecutar el metodo showVehiclesInParkingLotByYear(),showDataNewestVehicle(), showDataOldestVehicle() o showOcupationPercentageParkingLot() que van a hacer un recorrido parcial o global (dependiendo del caso) y se va a mostrar en pantalla lo que devuelven estos metodos | | |
| **Resultado o postcondición** | Se muestra en pantalla ya sea la información de los vehículos con ciertas características en el parqueadero o el porcentaje de ocupación del parqueadero | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| Información de los vehículos | String | Se muestra si el usuario eligió el criterio de búsqueda del año del vehículo o del vehículo más nuevo o más antiguo. |
|  | Porcentaje de ocupación del parqueadero | int | Se muestra solo si el usuario eligió el criterio del porcentaje de ocupación del parqueadero |