

Requerimientos Funcionales

1. Requerimientos Funcionales

Nombre	RF#1 Cargar la información de los vehículos.
Resumen	<p>Permite cargar la información de los vehículos disponibles en el dataset. Los atributos a cargar son los siguientes:</p> <ul style="list-style-type: none">• Buying: Precio de compra.• Maintenance: Precio de mantenimiento.• Doors: Número de puertas.• Persons: Número de personas máximo en el vehículo.• Lug_Boot: Tamaño de la cajuela.• Safety: Nivel de seguridad del vehículo.
Entradas	Un archivo con los datos de los vehículos.
Salidas	El archivo cargado de manera correcta.

Nombre	RF#2 Mostrar la información de los vehículos en una tabla.
Resumen	<p>Permite mostrar la información de los vehículos disponibles en el dataset en una tabla. Los atributos a mostrar son los siguientes:</p> <ul style="list-style-type: none">• Buying: Precio de compra.• Maintenance: Precio de mantenimiento.• Doors: Número de puertas.• Persons: Número de personas máximo en el vehículo.• Lug_Boot: Tamaño de la cajuela.• Safety: Nivel de seguridad del vehículo.
Entradas	
Salidas	Una tabla donde se visualizan los datos de cada vehículo cargado.

<i>Nombre</i>	RF#3 Filtrar los atributos de los vehículos en la tabla de acuerdo a categorías.
<i>Resumen</i>	Permite clasificar la información de los vehículos, de acuerdo a la elección del usuario y de acuerdo a los atributos de los vehículos.
<i>Entradas</i>	El atributo del vehículo, la categoría para filtrar.
<i>Salidas</i>	Una tabla con los valores filtrados de acuerdo a la categoría elegida por el usuario.

<i>Nombre</i>	RF#4 Crear gráficos para la información de los atributos de los vehículos de la tabla.
<i>Resumen</i>	<p>Permite mostrar 5 gráficos de acuerdo a los atributos de los vehículos. Los gráficos muestran la siguiente información.</p> <ul style="list-style-type: none"> • Cantidad de vehículos por precio de venta. • Cantidad de vehículos por precios de mantenimiento. • Cantidad de vehículos por número de puertas. • Cantidad de vehículos por número de personas. • Cantidad de vehículos por tamaño de la cajuela.
<i>Entradas</i>	
<i>Salidas</i>	Una gráfico por cada criterio establecido.

<i>Nombre</i>	RF#5 Determinar el resultado para la opción de compra del vehículo.
<i>Resumen</i>	<p>Permite decidir de acuerdo a los atributos de los vehículos, que tan buena o mala opción de compra es un vehículo. Los atributos para realizar dicha clasificación son los siguientes:</p> <ul style="list-style-type: none"> • Buying: Precio de compra.

	<ul style="list-style-type: none"> • Maintenance: Precio de mantenimiento. • Doors: Número de puertas. • Persons: Número de personas máximo en el vehículo. • Lug_Boot: Tamaño de la cajuela. • Safety: Nivel de seguridad del vehículo.
<i>Entradas</i>	Los atributos del vehículo a clasificar.
<i>Salidas</i>	BAD para mala opción de compra, GOOD para buena opción de compra.

<i>Nombre</i>	RF#6 Crear árbol de decisión para la información de los vehículos.
<i>Resumen</i>	Permite crear un árbol de decisión basado en un conjunto de entrenamiento que contiene diferentes valores para los atributos de los vehículos. Para crear el archivo de entrenamiento, se aplicó el método estadístico de selección de muestra. Una vez la muestra, seleccionada de manera aleatoria se extrae del archivo, los datos restantes se utilizan para crear el ID3.
<i>Entradas</i>	El archivo creado después de extraer la muestra representativa.
<i>Salidas</i>	El árbol de decisión creado a partir de dicho conjunto.

<i>Nombre</i>	RF#7 Probar el funcionamiento del árbol de decisión.
<i>Resumen</i>	Permite probar el árbol de decisión a partir de un conjunto de prueba. Este conjunto no está clasificado, por lo que el programa debe mostrar la salida correspondiente para cada ejemplo del conjunto. Los datos de este conjunto son extraídos por la muestra representativa y desclasificados.
<i>Entradas</i>	El archivo con los datos de prueba.
<i>Salidas</i>	BAD para mala opción de compra, GOOD para buena opción de compra. Este resultado por cada uno de los ejemplos enviados.

<i>Nombre</i>	RF#8 Mostrar gráficamente el árbol de decisión manual.
<i>Resumen</i>	Permite visualizar con un diagrama de árbol, el resultado obtenido para el árbol de decisión. De esta manera es posible observar la información de forma más organizada.
<i>Entradas</i>	
<i>Salidas</i>	El gráfico correspondiente para el árbol de decisión.

<i>Nombre</i>	RF#9 Crear árbol de decisión para la información de los vehículos usando una librería externa de C#.
<i>Resumen</i>	Permite crear un árbol de decisión basado en un conjunto de entrenamiento que contiene diferentes valores para los atributos de los vehículos. Para este caso se utilizó una librería externa de machine learning llamada Accord, y en base a ella se obtuvo un árbol de decisión.
<i>Entradas</i>	El archivo con el conjunto de entrenamiento.
<i>Salidas</i>	El árbol de decisión creado a partir de dicho conjunto.

<i>Nombre</i>	RF#10 Probar el funcionamiento del árbol de decisión usando una librería externa de C#.
<i>Resumen</i>	Permite probar el árbol de decisión usando una librería externa de C# a partir de un conjunto de pruebas. Este conjunto no está clasificado, por lo que el programa debe mostrar la salida correspondiente para cada ejemplo del conjunto..
<i>Entradas</i>	El archivo con los datos de prueba.
<i>Salidas</i>	BAD para mala opción de compra, GOOD para buena opción de compra. Este resultado por cada uno de los ejemplos enviados.

<i>Nombre</i>	RF#11 Determinar el resultado para la opción de compra del vehículo usando una librería externa de C#.
---------------	--

<i>Resumen</i>	<p>Permite decidir de acuerdo a los atributos de los vehículos, que tan buena o mala opción de compra es un vehículo. Los atributos para realizar dicha clasificación son los siguientes:</p> <ul style="list-style-type: none"> • Buying: Precio de compra. • Maintenance: Precio de mantenimiento. • Doors: Número de puertas. • Persons: Número de personas máximo en el vehículo. • Lug_Boot: Tamaño de la cajuela. • Safety: Nivel de seguridad del vehículo.
<i>Entradas</i>	Los atributos del vehículo a clasificar.
<i>Salidas</i>	BAD para mala opción de compra, GOOD para buena opción de compra.

<i>Nombre</i>	RF#12 Mostrar gráficamente el árbol de decisión de librería externa .
<i>Resumen</i>	Permite visualizar con un diagrama de árbol, el resultado obtenido para el árbol de decisión. De esta manera es posible observar la información de forma más organizada.
<i>Entradas</i>	
<i>Salidas</i>	El gráfico correspondiente para el árbol de decisión.

<i>Nombre</i>	RF#13 Crear gráficos para mostrar la información y los resultados obtenidos por el experimento realizado .
<i>Resumen</i>	<p>Permite mostrar 3 gráficos de acuerdo a los atributos de los vehículos. Los gráficos muestran la siguiente información:</p> <ul style="list-style-type: none"> • Resultados del experimento para la muestra: Sample 1. • Resultados del experimento para la muestra: Sample 2. • Resultados del experimento para la muestra: Training.
<i>Entradas</i>	
<i>Salidas</i>	El gráfico correspondiente para cada muestra.

2. Requerimientos No Funcionales

<i>Nombre</i>	RNF #1 Interfaz gráfica.
<i>Resumen</i>	El programa permite mostrar una interfaz gráfica y visualizar la solución del archivo cargado.

<i>Nombre</i>	RNF #2 Lenguaje de C#
<i>Resumen</i>	El programa debe ser implementado en el lenguaje de programación C#.

<i>Nombre</i>	RNF #3 Carga de datos por medio de dataset por CSV
<i>Resumen</i>	El programa debe obtener la información para clasificar a través de un archivo CSV.

<i>Nombre</i>	RNF #4 Datos del experimento guardados en archivo CSV
<i>Resumen</i>	El programa debe guardar los resultados del experimento realizado en un archivo CSV.