

F. R. Tiquetes

Juan Daniel Gómez Ríos, Daniel Ricardo Rodríguez, Daniel Amaris Rojas (Grupo 5/ Equipo 8)

I. INTRODUCCIÓN

El presente documento presenta una segunda visión/prototipo y explicación de funcionalidades ,esquemas e información de un software que facilita la reserva de tiquetes aéreos con a un usuario.

II. DESCRIPCIÓN DEL PROBLEMA A RESOLVER

Con este Proyecto se espera poder aplicar las estructuras de datos a un software que le permita a un Usuario hacer reserva de tiquetes aéreos con distintas aerolíneas que tengan implementado en su base de datos destinos nacionales e internacionales , y así mismo se asignen a una fecha de ida y de regreso .

III. USUARIOS DEL PRODUCTO DE SOFTWARE

El Usuario se encargara de escoger las respectivas fechas de su vuelo , así mismo como su origen y el destino a que piensa viajar , el Usuario tendra una identificación única y tendrá acceso para cancelar la reserva.

IV. REQUERIMIENTOS FUNCIONALES DEL SOFTWARE

• Creación de Vuelos de parte del administrador:

• Descripción:

El administrador puede crear nuevos vuelos que incluyen origen , destino , numero de vuelo , fechas de ida y de regreso

• Acciones iniciadoras y comportamiento esperado:

Se pedirá por orden los datos del vuelo , y luego terminado los campos de información se procederá a agregar el vuelo a una lista enlazada que es un atributo de una clase vuelo que se almacenará en un archivo txt

• Requerimientos funcionales:

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para completarla es tan sencillo como escribir o asignar los datos a pedir y darle a enter o en caso de interfaz darle aceptar,ok ,siguiente ;en caso que el adminisitrador escribe datos invalidos el programa dará a

conocer el error , para invalidar la acción de agregar vuelos

• Actualización de Vuelos de parte del administrador:

• Descripción:

El administrador puede actualizar vuelos que incluyen origen , destino , numero de vuelo , fechas de ida y de regreso y la capacidad del vuelo , también puede actualizar los pasajeros abordo ,algunas funcionalidades como borrar pasajeros o agregar , hacemos uso de árboles binarios AVL

• Acciones iniciadoras y comportamiento esperado:

Se pedirá el numero del vuelo , y luego se actualizara el vuelo de una lista enlazada que es un atributo de una clase vuelo que está almacenada en un archivo txt serializado

• Requerimientos funcionales:

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para completarla es tan sencillo como escribir o asignar los datos a pedir y darle a enter o en caso de interfaz darle aceptar,ok ,siguiente ;en caso que el adminisitrador escribe datos invalidos el programa dará a conocer el error , para invalidar la acción de agregar vuelos

• Eliminación de Vuelos de parte del administrador:

• Descripción

El administrador puede eliminar datos de los vuelos que incluyen origen , destino , numero de vuelo , fechas de ida y de regreso y la capacidad del vuelo, además usuarios con reserva en ese vuelo

• Acciones iniciadoras y comportamiento esperado:

Se pedirá por orden los datos del vuelo , y luego terminado los campos de información se procederá a quitar el vuelo a una lista enlazada que es un atributo de una clase vuelo que se almacenará en un archivo txt serializado

- *Requerimientos funcionales:*

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para completarla es tan sencillo como escribir o asignar los datos a pedir y darle a enter o en caso de interfaz darle aceptar,ok ,siguiente ;en caso que el adminisitrador escribe datos invalidos el programa dará a conocer el error , para invalidar la acción de agregar vuelos

Consulta de Vuelos de parte del administrador:

- *Descripción*

El administrador puede consultar datos de todos los vuelos almacenados que dentro de ellos incluyen origen , destino , numero de vuelo , fechas de ida y de regreso y la capacidad del vuelo, además usuarios con reserva en ese vuelo , en el caso de consultar datos de un Usuario el administrador implementa un árbol avl para su búsqueda.

- *Acciones iniciadoras y comportamiento esperado:*

Los vuelos se consultan en una base de datos que los almacena y el administrador solo tiene que dar a una opción “mostrar vuelos” y cada vez que entre a un vuelo ver los datos dentro de él, así como los datos de los usuarios

- *Requerimientos funcionales:*

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para consultar los datos es tan sencillo como escribir o asignar los datos a pedir y darle a enter o en caso de interfaz darle aceptar,ok ,siguiente a los vuelos indicados ;en caso que el adminisitrador escribe datos invalidos el programa dará a conocer el error , para invalidar la acción de consultar vuelos

Búsqueda parcial de Vuelos de parte del administrador:

- *Descripción*

El administrador puede buscar datos de un vuelo en específico que incluyen origen , destino , numero de vuelo , fechas de ida y de regreso y la capacidad del vuelo, además usuarios con reserve en ese vuelo

- *Acciones iniciadoras y comportamiento esperado:*

El administrador podrá accede a un vuelo en específico digitando el número de vuelo en una opción que se llame búsqueda de vuelo, también puede buscar vuelos apartir de destinos y respectivas fechas.

- *Requerimientos funcionales:*

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para completarla es tan sencillo como escribir o asignar el número de vuelo en una opción para administrador que diga”buscar vuelo” y darle a enter o en caso de interfaz darle aceptar,ok ,siguiente ;en caso que el adminisitrador escribe datos invalidos el programa dará a conocer el error , para invalidar la acción de busqueda de vuelo

Ordenamiento de Vuelos de parte del administrador:

- *Descripción*

El administrador puede ordenar los datos de un vuelo ,asi como el vuelo en si

- *Acciones iniciadoras y comportamiento esperado:*

Con la funcionalidad de consulta de vuelos o de busqueda de vuelos el administrador podrá acceder a los vuelos y a los datos de ellos para ordenar ciertos atributos como pueden ser pasajeros , destino de ida o vuelta de un vuelo y hasta su fecha.

- *Requerimientos funcionales:*

Solo el administrador tiene acceso a esta funcionalidad, por lo tanto contará con un código secreto o confidencial para acceder a ella ,para completarla es tan sencillo como elegir el vuelo con las funcionalidades de busqueda de vuelo y modificar por indice o por entrada el orden de algunos datos en especificos mencionados anteriormente

Asignación de Vuelos de parte del usuario:

- *Descripción*

Básicamente es la forma como el Usuario hace una reserva de un vuelo con una aerolínea

- *Acciones iniciadoras y comportamiento esperado:*

se pedirá al Usuario llenar campos de origen , destino y fechas de ida y vuelta asi como el número de pasajeros de la reserva, luego de completer la reserva el Usuario quedará almacenado en una matriz que hace parte de la estructura del vuelo, además todo Usuario se almacenará en un árbol avl , para facilitar su búsqueda e inserción.

- *Requerimientos funcionales:*

El Usuario tiene que completar la fácil tarea de llenar unos campos del vuelo que le interesa y luego dar unos datos específicos como lo son su nombre , su apellido , su dirección de correo electrónico y su id , y luego de dar a un botón de aceptar se completará la reserva , algunos campos serán obligatorios por lo que la reserve no se completará si el Usuario no proporciona los datos pedidos para el programa desde inicio a fin de la reserva, cuando la reserva se completa al Usuario se le muestra los datos de su vuelo.

Cancelación de Vuelos de parte del usuario:

- *Descripción*

El Usuario puede cancelar la reserva de vuelo realizada anteriormente con algunos datos

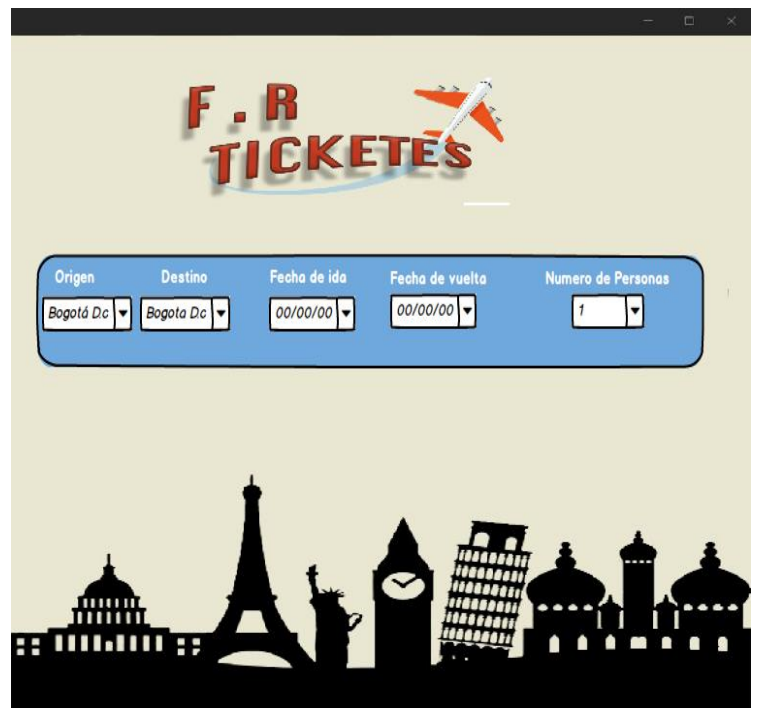
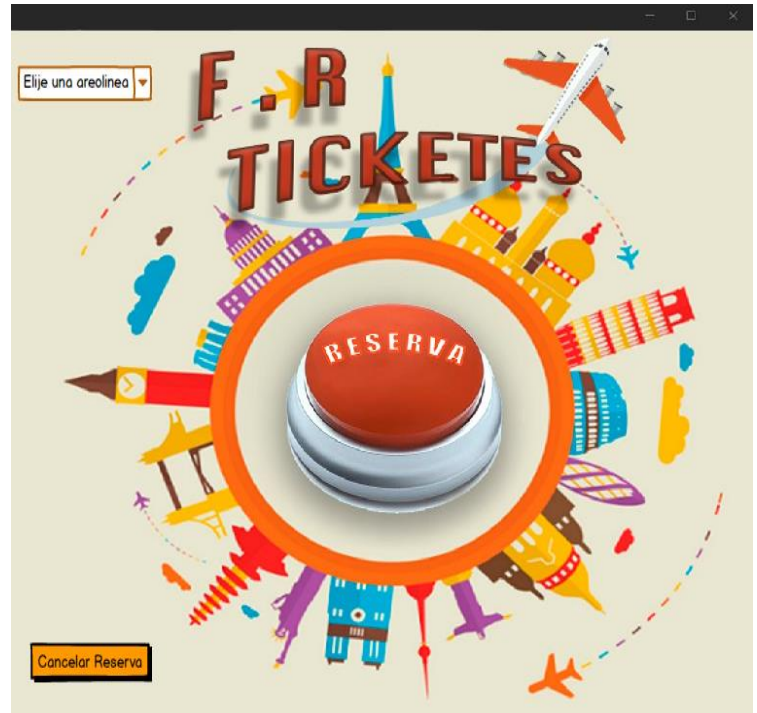
- *Acciones iniciadoras y comportamiento esperado:*

El Usuario tendrá que darle a la opción “cancelar reserva” Y proporcionar la máxima cantidad de datos de él y el número de vuelo y el asiento asignado , no todos los campos serán obligatorios pero abrá más precision de búsqueda , luego de la cancelación el Usuario se eliminara de la matriz que representa el avión de su vuelo , además se eliminará del árbol avl.

- *Requerimientos funcionales:*

El Usuario Tendrá que tener presente el número de vuelo que reservo , así como su o los asientos asignado , además opcionalmente puede poner los datos de destino y origen de la reserva como su respectiva fecha , luego de llenar esos campos y darle aceptar el Usuario habrá cancelado la reserva, si el Usuario no proporciona los campos obligatorios que pide la interfaz , no se completará la cancelación pero apartir de un mensaje de error se le informará al Usuario de la falla.

V.DESCRIPCIÓN DE LA INTERFAZ DE USUARIO PRELIMINAR



VI. ENTORNOS DE DESARROLLO Y OPERACIÓN

Se desarrollará en el lenguaje de programación Java , y los equipos a desarrollar el software tendrán procesadores de intel i7 de segunda generación hasta i7 septima generación y memoria RAM entre 4gb y 16gb , con un Sistema operativo de 64 bits

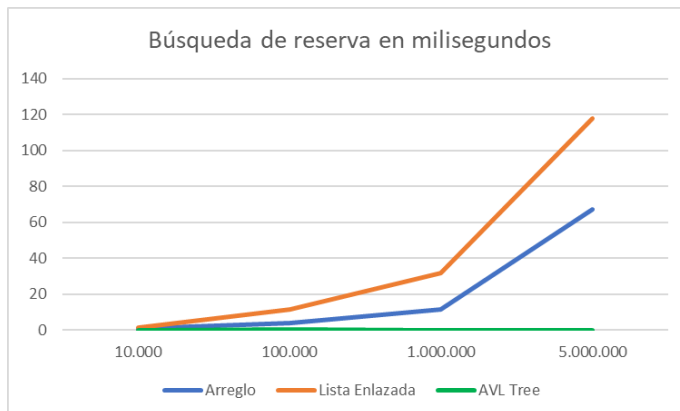
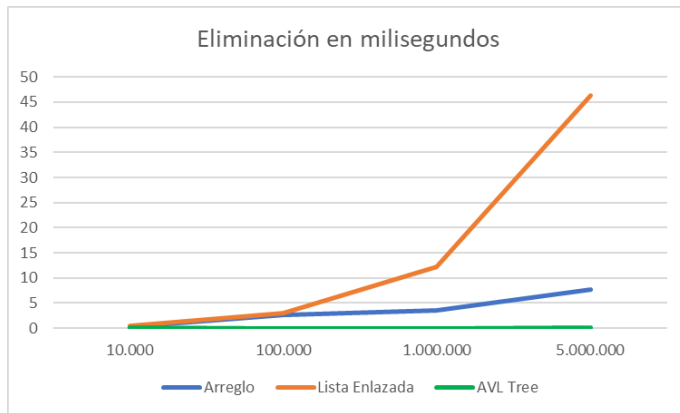
VII. PROTOTIPO DE SOFTWARE INICIAL

<https://github.com/ddamarisr/Data-Structures-Formatted>

VIII.PRUEBAS DEL PROTOTIPO

Nombre funcionalidad	Tipo(s) de estructura de datos	Cantidad de datos probados	Tiempos de ejecución (mS)
Eliminación de una reserva	10.000	Arreglo	0,3341
		Lista Enlazada	0,5328
		AVL	0,0442
		BST	1,2519
	100.000	Arreglo	2,6669
		Lista Enlazada	3,0314
		AVL	0,0264
		BST	Stack Overflow error
	1.000.000	Arreglo	3,5988
		Lista Enlazada	12,2863
		AVL	0,0264
		BST	Stack Overflow error
Añadido de una reserva	10.000	Arreglo	7,614
		Lista Enlazada	46,4165
		AVL	0,0357
		BST	Stack Overflow error
	100.000	Arreglo	0,0216
		Lista Enlazada	0,0005
		AVL	0,0205
		BST	0,4675
	1.000.000	Arreglo	0,1538
		Lista Enlazada	0,0067
		AVL	0,0187
		BST	Stack Overflow error
Búsqueda de una reserva	10.000	Arreglo	1,6399
		Lista Enlazada	0,0005
		AVL	0,01
		BST	Stack Overflow error
	100.000	Arreglo	8,8844
		Lista Enlazada	0,0001
		AVL	0,0458
		BST	Stack Overflow error
	1.000.000	Arreglo	1,0153
		Lista Enlazada	1,3664
		AVL	0,0204
		BST	1,2675
	10.000	Arreglo	3,8879
		Lista Enlazada	11,6141
		AVL	0,1671
		BST	Stack Overflow error
	100.000	Arreglo	11,6246
		Lista Enlazada	31,8325
		AVL	0,0075
		BST	Stack Overflow error
	1.000.000	Arreglo	67,1532
		Lista Enlazada	117,9878
		AVL	0,0214
		BST	Stack Overflow error

Funcionalidad	Estructura de Datos	Análisis de complejidad Big(O)
Eliminación por valor	Arreglo	$O(n)$
	Lista Enlazada	$O(n)$
	AVL	$O(\log_2(n))$
	BST	$O(h) / O(n)$ si es degenerado
Insertar	Arreglo	$O(n)$ al final+ incremento tamaño
	Lista Enlazada	$O(1)$ al final
	AVL	$O(\log_2(n))$
	BST	$O(h) / O(n)$ si es degenerado
Búsqueda por valor	Arreglo	$O(n)$
	Lista Enlazada	$O(n)$
	AVL	$O(\log_2(n))$
	BST	$O(h) / O(n)$ si es degenerado



- Durante las pruebas de rendimiento tuvimos muchas dificultades para realizar las pruebas sobre los árboles BST pues se tardaban demasiado tiempo o directamente no funcionaban (Stack Overflow Error). Descubrimos que esto se explica porque al ingresar los datos de prueba en orden estábamos ocasionando que el BST se degenerara a tal punto que era imposible ingresarle 100.000 datos o más dentro de un tiempo razonable.
- Las pilas son muy útiles a la hora de implementar y manejar las ventanas de la interfaz de usuario.
- La serialización de java es de gran ayuda a la hora de generar persistencia de objetos. En general es mejor alternativa que escribir la información a texto plano.

VIII. DIFICULTADES Y LECCIONES APRENDIDAS

- La implementación de un árbol AVL es un reto significativo pues se trata de una estructura de datos mucho más complicada de implementar que las vistas anteriormente. No obstante, como muestran las pruebas de rendimiento, implementar estos árboles en datos sobre los que haya que realizar búsquedas vale la pena.