

28 DE SEPTIEMBRE DE 2022

PROYECTO PROGRAMADO 2
SISTEMA DE ALQUILER DE BICICLETAS

JEAN HUNT
LENGUAJES DE PROGRAMACIÓN
Ingeniería en computación




Tabla de contenido

Manual de usuario.....	1
Compilación y ejecución.....	2
Tutorial	3
1. <i>Menú principal</i>	4
2. <i>Menú operativo</i>	4
3. <i>Menú general</i>	9
Documentación general	13
1. Descripción del problema.....	14
2. Diseño del programa:	14
2.1. <i>Decisiones de diseño</i>	14
2.2. <i>Algoritmos usados</i>	15
3. Librerías utilizadas.....	17
4. Análisis de resultados	17

Ilustración 1: Compilación y ejecución	2
Ilustración 2: Ejecución sin compilación.....	3
Ilustración 3: Carga de datos	3
Ilustración 4: Menú principal.....	4
Ilustración 5: Menú operativo.....	4
Ilustración 6: Mostrar parqueos	5
Ilustración 7: Mostrar bicicletas	5
Ilustración 8: Mostrar usuarios	6
Ilustración 9: Menú estadísticas.....	7
Ilustración 10: Top 5 usuarios	7
Ilustración 11: Top 5 parqueos	8
Ilustración 12: Top 3 bicicletas.....	8
Ilustración 13: Resumen	8
Ilustración 14: Menú general	9
Ilustración 15: Consulta de bicicletas	10
Ilustración 16: Alquiler 1	10
Ilustración 17: Alquiler 2.....	11
Ilustración 18: Resumen alquiler	11
Ilustración 19: Facturar alquiler.....	12
Ilustración 20: Objetivos alcanzados	18

Manual de usuario

Compilación y ejecución

Este sistema de alquiler de bicicletas funciona a través de la consola de comandos y archivos de texto, no utiliza ninguna librería externa al lenguaje por lo que lo único que se necesita para ejecutar el programa es tener el compilador del lenguaje de programación Haskell.

La guía para la instalación y ejecución del compilador se encuentra en el siguiente enlace: <https://www.haskell.org/downloads/>

Una vez instalado el compilador, para ejecutar el programa seguiremos los siguientes pasos:

1. Abrir una terminal de comandos, ya sea integrada o de terceros.
2. Dirigirse a la ruta en donde se encuentra el archivo proyecto.
3. Compilar el programa haciendo uso del siguiente comando: `ghc "nombreArchivo.hs"`, en este caso: `ghc main.hs`

Nota: Puede compilar el programa desde cualquier directorio, pero tendrá que ingresar la ruta completa del archivo, por ejemplo: `ghc C://escritorio/programa/main.hs`

4. La compilación creará 3 archivos: `main.hi`, `main.o` y `main.exe`
5. Para ejecutar el programa ingrese “`main.exe`” en la consola

```
C:\Users\jeanp\Desktop\TEC\2022\2[Semestre]\Lenguajes\Proyectos\PY2\P2-Lenguajes-S2-2022>ghc main.hs
[1 of 2] Compiling Estructuras      ( Estructuras.hs, Estructuras.o )
[2 of 2] Compiling Main              ( main.hs, main.o )
Linking main.exe ...

C:\Users\jeanp\Desktop\TEC\2022\2[Semestre]\Lenguajes\Proyectos\PY2\P2-Lenguajes-S2-2022>main.exe

##### Alquiler de bicicletas #####
```

Ilustración 1: Compilación y ejecución

6. El programa también se puede ejecutar sin ser compilado, para ello utilice el comando `runghc main.hs`

```
C:\Users\jeanp\Desktop\TEC\2022\2\Semestre\Lenguajes\Proyectos\PY2\P2-Lenguajes-S2-2022>runghc main.hs
##### Alquiler de bicicletas #####
```

Ilustración 2: Ejecución sin compilación

Todos los datos que sean solicitados deberán ser ingresados mediante la consola de comandos, por ejemplo, selección mediante índices numéricos o entrada por teclado de algunos valores.

Tutorial

El sistema se divide en dos submenús derivados de un menú principal los cuales son menú operativo y menú general. Una vez iniciado el programa se solicitarán 3 archivos que corresponden al almacén de datos de parqueos, bicicletas y usuarios. El programa incluye estos archivos, sin embargo, se puede utilizar algún otro directorio siempre y cuando cumpla con el formato necesario.

```
##### Alquiler de bicicletas #####
Indique la ruta de parqueos:
parqueos.txt
Indique la ruta de bicicletas:
bicicletas.txt
Indique la ruta de usuarios:
usuarios.txt
```

Ilustración 3: Carga de datos

1. Menú principal

Este es el primer menú desplegado luego de indicar la ruta de los archivos que contienen la base de datos. Este menú contiene 3 opciones, dos de ellas derivan a los menús principales del sistema y la tercera permite salir del mismo.

```
Menú principal
1. Opciones operativas
2. Opciones generales
3. Salir

Seleccione una opción:
```

Ilustración 4: Menú principal

2. Menú operativo

Para acceder a este menú se debe seleccionar la opción 1 en el menú principal. Una vez seleccionada se desplegará el menú en consola. Este menú está compuesto de 4 opciones enfocadas al análisis y carga de datos del sistema.

```
Menú operativo
1. Mostrar parqueos
2. Mostrar bicicletas
3. Cargar usuarios
4. Estadísticas
5. Volver
6. Salir

Seleccione una opción:
_
```

Ilustración 5: Menú operativo

2.1. Mostrar parqueos

Al seleccionar esta opción se imprimirá en consola una serie de parqueos. Estos datos se encuentran registrados en el archivo indicado al inicio del programa, en este caso “parqueos.txt” y poseen la estructura que se muestra en la siguiente imagen:

```
##### Parqueos #####
```

Nombre: "ASJ"	Direccion: "direccion ASJ"	Provincia: San Jose	Ubicacion x: -147.5057	Ubicacion y: -479.76776
Nombre: "BHE"	Direccion: "direccion BHE"	Provincia: Heredia	Ubicacion x: 269.88977	Ubicacion y: 87.74979
Nombre: "CLI"	Direccion: "direccion CLI"	Provincia: Limón	Ubicacion x: 961.5822	Ubicacion y: 6.8928
Nombre: "DGU"	Direccion: "direccion DGU"	Provincia: Guanacaste	Ubicacion x: -774.5799	Ubicacion y: -339.7484
Nombre: "ECA"	Direccion: "direccion ECA"	Provincia: Cartago	Ubicacion x: 231.30699	Ubicacion y: -475.09106
Nombre: "FAL"	Direccion: "direccion FAL"	Provincia: Alajuela	Ubicacion x: -96.06201	Ubicacion y: 118.84984
Nombre: "GLI"	Direccion: "direccion GLI"	Provincia: Limón	Ubicacion x: 872.0149	Ubicacion y: -297.37646
Nombre: "HCA"	Direccion: "direccion HCA"	Provincia: Cartago	Ubicacion x: 525.9391	Ubicacion y: -407.2789
Nombre: "ISJ"	Direccion: "direccion ISJ"	Provincia: San Jose	Ubicacion x: -273.7766	Ubicacion y: -255.28615
Nombre: "JAL"	Direccion: "direccion JAL"	Provincia: Alajuela	Ubicacion x: -327.55865	Ubicacion y: 181.98529
Nombre: "KPU"	Direccion: "direccion KPU"	Provincia: Puntarenas	Ubicacion x: 93.34434	Ubicacion y: -802.46
Nombre: "LPU"	Direccion: "direccion LPU"	Provincia: Puntarenas	Ubicacion x: 656.88666	Ubicacion y: -748.678

Ilustración 6: Mostrar parqueos

2.2. Mostrar bicicletas

Al seleccionar esta opción se desplegará la lista de parqueos registrados en el sistema y se solicitará el nombre de uno de ellos. Posteriormente se listarán las bicicletas que se encuentren en dicho parqueo.

Además, se puede seleccionar la opción “#” para mostrar todas las bicicletas registradas en el sistema o “transito” para mostrar las bicicletas que actualmente están alquiladas.

La ejecución de esta opción se ve de la siguiente manera:

```
##### Parqueos #####
```

Nombre: "ASJ"	Direccion: "direccion ASJ"	Provincia: San Jose	Ubicacion x: -147.5057	Ubicacion y: -479.76776
Nombre: "BHE"	Direccion: "direccion BHE"	Provincia: Heredia	Ubicacion x: 269.88977	Ubicacion y: 87.74979
Nombre: "CLI"	Direccion: "direccion CLI"	Provincia: Limón	Ubicacion x: 961.5822	Ubicacion y: 6.8928
Nombre: "DGU"	Direccion: "direccion DGU"	Provincia: Guanacaste	Ubicacion x: -774.5799	Ubicacion y: -339.7484
Nombre: "ECA"	Direccion: "direccion ECA"	Provincia: Cartago	Ubicacion x: 231.30699	Ubicacion y: -475.09106
Nombre: "FAL"	Direccion: "direccion FAL"	Provincia: Alajuela	Ubicacion x: -96.06201	Ubicacion y: 118.84984
Nombre: "GLI"	Direccion: "direccion GLI"	Provincia: Limón	Ubicacion x: 872.0149	Ubicacion y: -297.37646
Nombre: "HCA"	Direccion: "direccion HCA"	Provincia: Cartago	Ubicacion x: 525.9391	Ubicacion y: -407.2789
Nombre: "ISJ"	Direccion: "direccion ISJ"	Provincia: San Jose	Ubicacion x: -273.7766	Ubicacion y: -255.28615
Nombre: "JAL"	Direccion: "direccion JAL"	Provincia: Alajuela	Ubicacion x: -327.55865	Ubicacion y: 181.98529
Nombre: "KPU"	Direccion: "direccion KPU"	Provincia: Puntarenas	Ubicacion x: 93.34434	Ubicacion y: -802.46
Nombre: "LPU"	Direccion: "direccion LPU"	Provincia: Puntarenas	Ubicacion x: 656.88666	Ubicacion y: -748.678

```
-----
-Ingresa el nombre del parqueo que desea consultar
-'#' Para consultar todas las bicicletas
-o 'transito' para consultar las bicicletas en tránsito
Opción:
ASJ

##### Bicicletas #####
```

Identificador: "011-AE"	tipo: "AE"	Parqueo: "ASJ"
Identificador: "012-TR"	tipo: "TR"	Parqueo: "ASJ"
Identificador: "014-TR"	tipo: "TR"	Parqueo: "ASJ"
Identificador: "016-TR"	tipo: "TR"	Parqueo: "ASJ"

Ilustración 7: Mostrar bicicletas

Estos datos se encuentran precargadas en el sistema y están almacenadas en el archivo indicado al inicio del programa, en este caso “bicicletas.txt”.

Nota: El nombre del parque debe ser idéntico al indicado por consola, esto quiere decir que las mayúsculas y minúsculas deben coincidir

2.3. Cargar usuarios

Al seleccionar esta opción se imprimirá en consola una serie de usuarios. Estos datos se encuentran registrados en el archivo indicado al inicio del programa, en este caso “usuarios.txt” y poseen la estructura que se muestra en la siguiente imagen:

```

Seleccione una opción:
3

##### Usuarios #####

Cedula: 662664284      nombre: "Valentina Ochoa"
Cedula: 292634569      nombre: "Estefania Rios"
Cedula: 381595807      nombre: "Jordi Amador"
Cedula: 590114029      nombre: "Maria Cano"
Cedula: 354776885      nombre: "Patricio Feijo"
Cedula: 481212646      nombre: "Rafael Herrero"
Cedula: 369637462      nombre: "Jhon Smith"
Cedula: 579565026      nombre: "Cristina Mejias"
Cedula: 183196389      nombre: "Celeste Fernandez"
Cedula: 229844674      nombre: "Agustin Gomez"

-----

```

Ilustración 8: Mostrar usuarios

2.4. Estadísticas

Al seleccionar esta opción se desplegará un submenú con 4 opciones relacionadas al análisis de los datos registrados. Dicho menú tiene la siguiente estructura:

```

Seleccione una opción:
4

Menú estadísticas
1. Top 5 usuarios con más viajes
2. Top 5 parqueos con más viajes
3. Top 3 bicicletas con más kilómetros recorridos
4. Resumen
5. Volver
6. Salir
Seleccione una opción:
_

```

Ilustración 9: Menú estadísticas

2.4.1. Top 5 usuarios

Despliega una lista de los 5 usuarios con más viajes completados:

```

##### Top 5 usuarios #####

Usuario: 662664284      Cantidad de viajes completados: 3
Usuario: 481212646      Cantidad de viajes completados: 3
Usuario: 229844674      Cantidad de viajes completados: 2
Usuario: 381595807      Cantidad de viajes completados: 1
Usuario: 590114029      Cantidad de viajes completados: 1
-----

```

Ilustración 10: Top 5 usuarios

2.4.2. Top 5 parqueos

Despliega una lista de los 5 parqueos con más viajes ya sea que este haya sido el parqueo de salida o de llegada:

```
##### Top 5 parqueos #####

Parqueo: ASJ      Cantidad de viajes salida/llegada: 4
Parqueo: HCA      Cantidad de viajes salida/llegada: 4
Parqueo: BHE      Cantidad de viajes salida/llegada: 3
Parqueo: FAL      Cantidad de viajes salida/llegada: 3
Parqueo: ISJ      Cantidad de viajes salida/llegada: 3

-----
```

Ilustración 11: Top 5 parqueos

2.4.3. Top 3 bicicletas

Despliega una lista de las 3 bicicletas que más distancia han recorrido:

```
##### Top 3 bicicletas #####

Bicicleta: 009-TR      Cantidad de Km recorridos: 1488.731
Bicicleta: 006-AE      Cantidad de Km recorridos: 1211.1625
Bicicleta: 001-TR      Cantidad de Km recorridos: 1071.5897

-----
```

Ilustración 12: Top 3 bicicletas

2.4.4. Resumen

Muestra un resumen de los datos facturados con la siguiente información: Viajes terminados, total de kilómetros recorridos por las bicicletas, total facturado en colones.

```
##### Resumen #####

Cantidad viajes terminados: 13
Total de kilometros: 9537.183
Total facturado: 9833194.0

-----
```

Ilustración 13: Resumen

2.5. Volver

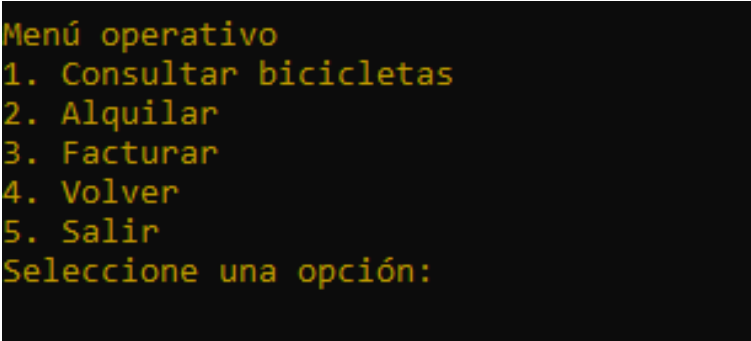
Vuelve al menú principal.

2.6. Salir

Termina la ejecución del programa.

3. *Menú general*

Para acceder a este menú se debe seleccionar la opción 2 en el menú principal. Una vez seleccionada se desplegará el menú en consola. Este menú está compuesto de 3 opciones enfocadas alquiler de bicicletas.



```
Menú operativo
1. Consultar bicicletas
2. Alquilar
3. Facturar
4. Volver
5. Salir
Seleccione una opción:
```

Ilustración 14: Menú general

3.1. Consultar bicicletas

Esta opción le permite al usuario consultar cual es el parqueo más cercano a su posición y sus bicicletas.

Para ello, se le solicitará al usuario que indique su posición (x, y), se calculará cual es el parqueo más cercano y se le mostrará al usuario su información y la información de sus bicicletas tal y como se muestra en la siguiente imagen:

```
##### Consulta de bicicletas #####
Indique x:
671
Indique y:
1022

-----

El parqueo más cercano es:

Parqueo: "BHE"
Direccion: "direccion BHE"
Provincia: Heredia
X: 269.88977
Y: 87.74979

-----

##### Bicicletas #####

Identificador: "004-TR"      tipo: "TR"      Parqueo: "BHE"
Identificador: "010-AE"      tipo: "AE"      Parqueo: "BHE"
Identificador: "015-TR"      tipo: "TR"      Parqueo: "BHE"

-----
```

Ilustración 15: Consulta de bicicletas

3.2. Alquilar

Esta opción le permite al usuario alquilar una bicicleta. Primeramente, se le mostraran los usuarios registrados en el sistema de los cuales deberá seleccionar uno indicando su cedula tal y como se muestra en la siguiente imagen:

```
##### Alquiler de bicicletas #####

##### Usuarios #####

Cedula: 662664284      nombre: "Valentina Ochoa"
Cedula: 292634569      nombre: "Estefania Rios"
Cedula: 381595807      nombre: "Jordi Amador"
Cedula: 590114029      nombre: "Maria Cano"
Cedula: 354776885      nombre: "Patricio Feijo"
Cedula: 481212646      nombre: "Rafael Herrero"
Cedula: 369637462      nombre: "Jhon Smith"
Cedula: 579565026      nombre: "Cristina Mejias"
Cedula: 183196389      nombre: "Celeste Fernandez"
Cedula: 229844674      nombre: "Agustin Gomez"

-----

Ingrese la cedula del usuario o (#) para cancelar el alquiler:
292634569
```

Ilustración 16: Alquiler 1

Seguidamente se le mostrarán los parqueos registrados en el sistema, el usuario deberá seleccionar el parque de salida y el parqueo de llegada. Una vez seleccionados los parqueos se mostrarán las bicicletas que se encuentren en el parqueo de salida de las cuales deberá seleccionar una ingresando su identificador cuando se le solicita. Este procedimiento se ve de la siguiente manera:

```
##### Parqueos #####
Nombre: "ASJ"   Direccion: "direccion ASJ"   Provincia: San Jose   Ubicacion x: -147.5057   Ubicacion y: -479.76776
Nombre: "BHE"   Direccion: "direccion BHE"   Provincia: Heredia   Ubicacion x: 269.88977   Ubicacion y: 87.74979
Nombre: "CLI"   Direccion: "direccion CLI"   Provincia: Limón     Ubicacion x: 961.5822   Ubicacion y: 6.8928
Nombre: "DGU"   Direccion: "direccion DGU"   Provincia: Guanacaste   Ubicacion x: -774.5799   Ubicacion y: -339.7484
Nombre: "ECA"   Direccion: "direccion ECA"   Provincia: Cartago   Ubicacion x: 231.30699   Ubicacion y: -475.09106
Nombre: "FAL"   Direccion: "direccion FAL"   Provincia: Alajuela   Ubicacion x: -96.06201   Ubicacion y: 118.84984
Nombre: "GLI"   Direccion: "direccion GLI"   Provincia: Limón     Ubicacion x: 872.0149   Ubicacion y: -297.37646
Nombre: "HCA"   Direccion: "direccion HCA"   Provincia: Cartago   Ubicacion x: 525.9391   Ubicacion y: -407.2789
Nombre: "ISJ"   Direccion: "direccion ISJ"   Provincia: San Jose   Ubicacion x: -273.7766   Ubicacion y: -255.28615
Nombre: "JAL"   Direccion: "direccion JAL"   Provincia: Alajuela   Ubicacion x: -327.55865   Ubicacion y: 181.98529
Nombre: "KPU"   Direccion: "direccion KPU"   Provincia: Puntarenas   Ubicacion x: 93.34434   Ubicacion y: -802.46
Nombre: "LPU"   Direccion: "direccion LPU"   Provincia: Puntarenas   Ubicacion x: 656.88666   Ubicacion y: -748.678

-----

Ingrese el nombre del parqueo de salida o (#) para cancelar el alquiler:
CLI

Ingrese el nombre del parqueo de llegada o (#) para cancelar el alquiler:
DGU

##### Bicicletas #####

Identificador: "003-TR"   tipo: "TR"   Parqueo: "CLI"
Identificador: "006-AE"   tipo: "AE"   Parqueo: "CLI"
Identificador: "019-TR"   tipo: "TR"   Parqueo: "CLI"

-----

Ingrese el identificador de la bicicleta o (#) para cancelar el alquiler:
019-TR
```

Ilustración 17: Alquiler 2

Una vez seleccionados todos los datos se generará el alquiler y se mostrará un resumen de su información en pantalla:

```
¡Se ha generado el alquiler!

-----
Codigo: 13
Cedula: 292634569
Salida: CLI
Llegada: DGU
Bicicleta: 019-TR
-----
```

Ilustración 18: Resumen alquiler

3.3. Facturar

Esta opción permite al usuario facturar un alquiler activo, es decir, indicar que se ha completado el viaje. Para ello se el mostrarán al usuario todos los alquileres que se encuentren activos.

Para facturar el alquiler se debe indicar el código cuando se solicite. Una vez indicado, se procederá al facturar dicho alquiler, se generará una factura y se mostrará en consola de la siguiente manera:

```
##### Facturación #####

##### Alquileres Activos #####
Codigo: 13 usuario: 292634569 Salida: CLI Llegada: DGU Bicicleta: 019-TR Estado: activo

-----
Ingrese el identificador del alquiler o (#) para cancelar la facturación:
13

¡El alquiler 13 ha sido facturado!

Se ha generado la siguiente factura

##### Factura #####
Codigo: 15
Nombre empresa: CletasYa
Sitio web: cletasya.cr
Contacto: 27123456
Usuario: 292634569
Parqueo Salida: CLI    Parqueo Llegada: DGU
idBicicleta: 019-TR    tipoBicicleta: tradicional
Distancia recorrida: 1770.4291
Tarifa x Kilometro: 420.0
Total colones: 743580.2

-----
```

Ilustración 19: Facturar alquiler

3.4. Volver

Vuelve al menú principal.

3.5. Salir

Termina la ejecución del programa.

Documentación general

1. Descripción del problema

Se deberá implementar una aplicación de escritorio para administrar la gestión de alquiler de bicicletas utilizando el lenguaje Haskell y la Programación Funcional. El programa deberá desplegar un menú al usuario con dos submenús o grupo de funciones (el menú debe mostrarse recurrentemente después de cada opción realizada -a excepción del salir- y tener la opción de subir al menú principal desde los menús secundarios) y salir

***Nota:** La descripción anterior fue extraída de la especificación correspondiente a esta tarea programada

2. Diseño del programa:

El programa fue desarrollado en el lenguaje de programación Haskell. La interfaz del programa será únicamente a través de la consola de comandos y mediante entrada del teclado.

Existen 2 archivos que componen el programa:

- **Main.hs:** Contiene todas las funcionalidades del programa.
- **Estructuras.hs:** Contiene los tipos de datos (estructuras) que se utilizan para la manipulación de datos

2.1. Decisiones de diseño

Antes y durante el desarrollo del proyecto se tomaron las siguientes decisiones:

- El programa será desarrollado en lenguaje Haskell
- Se utilizará vscode como entorno de desarrollo.
- La persistencia de datos se hará mediante uso de archivos de texto plano (sin embargo, archivos con extensión csv también funcionan)
- La navegación por menus se hará mediante índices indicados en el mismo.

- El programa se desarrollará únicamente para ser ejecutado en consola.
- Únicamente se utilizarán librerías ya incluidas en el lenguaje.
- Se utilizará GitHub como control de repositorio.

2.2. Algoritmos usados

A continuación, se encuentra un listado con las declaraciones de los algoritmos utilizados en los programas:

```
menuPrincipal :: (Integer, FilePath, FilePath, FilePath) -> IO b
menuOperativo :: (Integer, FilePath, FilePath, FilePath) -> IO ()
menuGeneral :: (Integer, FilePath, FilePath, FilePath) -> IO ()
menuEstadisticas :: (Integer, [Parqueo], [Bicicleta], [Usuario]) -> IO ()
cargarParqueos :: FilePath -> IO [Parqueo]
separaParqueos :: [[Char]] -> [Parqueo]
showParqueo :: Parqueo -> [Char]
showParqueos :: [Parqueo] -> IO()
getParqueoCercano :: (Float, Float, [Parqueo], Parqueo) -> IO String
calcularDistanciaParqueo :: (Float, Float, Parqueo) -> Float
seleccionarParqueoS :: ([Parqueo], [Bicicleta]) -> IO String
seleccionarParqueoL :: ([Parqueo], String) -> IO String
existeParqueo :: ([Parqueo], String) -> Bool
getParqueoXNombre :: Monad m => (String, [Parqueo]) -> m Parqueo
getTop5Parqueos :: ([Factura], [Parqueo]) -> IO ()
getViajesXParqueo :: ([Factura], [Parqueo]) -> [[String]]
getViajesXParqueoAux :: Num p => (String, [Factura]) -> p
cargarAlquileres :: FilePath -> IO [Alquiler]
separarAlquileres :: [[Char]] -> [Alquiler]
alquilar :: ([Parqueo], [Bicicleta], FilePath) -> IO ()
getAlquiler :: Monad m => (Integer, [Alquiler]) -> m Alquiler
seleccionarAlquiler :: [Alquiler] -> IO String
existeAlquiler :: [Alquiler], Integer, String -> Bool
facturarAlquiler :: [Alquiler], Integer, [Char] -> IO ()
facturarAlquilerAux :: [Alquiler], Integer, [Char] -> IO ()
```

```

showAlquilerActivo :: [Alquiler] -> IO()
cargarFacturas :: FilePath-> IO [Factura]
separarFacturas :: [[Char]] -> [Factura]
facturar :: ([Bicicleta], [Parqueo]) -> IO ()
printFactura :: Factura -> IO ()
resumen :: [Factura] -> IO ()
resumenAux :: (Show b, Num b) => ([Factura], b, Float, Float) -> IO ()
cargarBicicletas :: FilePath-> IO [Bicicleta]
separaBicicletas :: [[Char]] -> [Bicicleta]
showBicicletas :: ([Bicicleta],String,Integer) -> IO()
consultarBicicletas :: ([Parqueo], [Bicicleta]) -> IO ()
getBicicletasParqueo :: ([Bicicleta], String) -> [Bicicleta]
seleccionarBicicleta :: [Bicicleta] -> IO String
existeBicicleta :: ([Bicicleta], String) -> Bool
bicicletaUbicacion :: ([Bicicleta], String, [Char]) -> IO ()
bicicletaUbicacionAux :: ([Bicicleta], String, [Char]) -> IO ()
getTipoBicicleta2 :: (String, [Bicicleta]) -> String
getTop3Bicicletas :: ([Factura], [Bicicleta]) -> IO ()
getDistanciaBici :: ([Factura], [Bicicleta]) -> [[String]]
getDistanciaBiciAux :: (String, [Factura]) -> Float
mostrarBicicletas :: [Bicicleta] -> IO ()
cargarUsuarios :: FilePath-> IO [Usuario]
separaUsuarios :: [[Char]] -> [Usuario]
showUsuario :: Usuario -> [Char]
showUsuarios :: [Usuario] -> IO()
seleccionarUsuario :: [Usuario] -> IO String
existePersona :: ([Usuario], Integer) -> Bool
getTop5Usuarios :: ([Factura], [Usuario]) -> IO ()
getViajesXUsuario :: ([Factura], [Usuario]) -> [[String]]
getViajesXUsuarioAux :: Num p => (Integer, [Factura]) -> p
toLines :: String -> [String]
separarPorComas :: ([Char], [Char]) -> [[Char]]
getInput :: IO String
getProvincia :: [Char] -> [Char]
getNombreArchivo :: IO String
lowerString :: [Char] -> [Char]
calcularDistancia :: Floating a => (a, a, a, a) -> a
getDistaciaRecorrida :: Monad m => (String, String, [Parqueo]) -> m Float
getTarifa :: [Char] -> Float
imprimirListaTop :: ([[[Char]]], [Char], [Char]) -> IO ()
salir :: IO b
main :: IO b

```

3. Librerías utilizadas

Para el desarrollo del programa únicamente se utilizaron librerías nativas del lenguaje Haskell las cuales se enlistan a continuación:

- **Data.Char:** Se importaron las siguientes funciones
 - *isDigit*: Saber si un char es un numero
 - *toLower*: Traducir un char a minúscula
- **Data.List:** Se utilizó la función *sortBy* para organizar listas
- **System.Exit:** Se utilizó la función *exitSuccess* para cerrar el programa.
- **System.Directory:** se utilizó la función *doesFileExist* para saber si un directorio existe o no.
- **Control.DeepSeq y Control.Exception:** Se utilizaron las funciones *force* y *evaluate* para forzar el cierre de un programa. Esto porque al intentar añadir nuevas entradas a un archivo el sistema no lo permitía por que dicho archivo seguía abierto.

***Nota:** Esta solución fue extraída del siguiente hilo de StackOverflow:

<https://stackoverflow.com/questions/22166912/how-to-close-a-file-in-haskell>

4. Análisis de resultados

En la siguiente tabla se muestra una lista de objetivos para los cuales se evaluará si fueron cumplidos o no, en caso de no cumplirse también se indicará la razón del porqué.

Objetivo	Alcanzado	Razón
Aprender la sintaxis y semántica del lenguaje Haskell	Si	
Cumplir con las funcionalidades del programa solicitadas	Si	
Obtener una alta rigidez durante la ejecución	Si	
Documentación interna del programa	Si	
Documentación externa del programa	Si	
Extra: Aplicar persistencia de datos	Si	

Ilustración 20: Objetivos alcanzados

En el siguiente enlace se puede encontrar el repositorio que contiene el proyecto

<https://github.com/JPHuntV/P2-Lenguajes-S2-2022.git>