**Universidad ORT Uruguay**

Facultad de Ingeniería

Escuela de Tecnología

Obligatorio (1)

Programación 3  
- Documentación -

Hombre con barba y bigote

Descripción generada automáticamente

Lucas Divenuto-293644

  
Sofía Fernández – 292523

N3C  
Docente: Plinio Gañi.

11/05/2023

### Índice

Documento de análisis

[Índice 0](#_Toc134700793)

[Requerimientos funcionales 1](#_Toc134700794)

[Usuario 1](#_Toc134700795)

[Login() 1](#_Toc134700796)

[Usuarios y contraseñas precargados: 1](#_Toc134700797)

[Cabañas 1](#_Toc134700798)

[Add() 1](#_Toc134700799)

[FindById() 1](#_Toc134700800)

[FindAll() 1](#_Toc134700801)

[FindCabanaNombre() 1](#_Toc134700802)

[FindCabanaMax() 2](#_Toc134700803)

[FindCabanaTipo() 2](#_Toc134700804)

[FindCabanasHabilitadas() 2](#_Toc134700805)

[Tipos 2](#_Toc134700806)

[Add() 2](#_Toc134700807)

[FindById() 2](#_Toc134700808)

[FindAll() 2](#_Toc134700809)

[FindTipoByNombre() 2](#_Toc134700810)

[Remove() 3](#_Toc134700811)

[Update() 3](#_Toc134700812)

[Mantenimientos 3](#_Toc134700813)

[Add() 3](#_Toc134700814)

[FindMantenimientosCabana() 3](#_Toc134700815)

[FindMantenimientosFechas() 3](#_Toc134700816)

[Parámetros 3](#_Toc134700817)

[ValorParametro() 3](#_Toc134700818)

# Requerimientos funcionales

## Usuario

### Login()

**Ubicación**: Datos/Repositorios/RepositorioUsuario

**Parámetros**: string email, string password

**Función**: Permite el ingreso al sistema para poder acceder a todas las funcionalidades. Recibe dos parámetros de tipo string que representan un usuario y una contraseña.

### Usuarios y contraseñas precargados:

|  |  |
| --- | --- |
| **Usuario** | **Contraseña** |
| sofi@gmail.com | 1234abC |
| lucas@gmail.com | 7654abC |
| admin@gmail.com | Qwerty1 |

## Cabañas

### Add()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: Cabana obj

**Función**: Ejecuta el agregado de una nueva cabaña a la base de datos. Antes de agregar se realiza la validación del objeto con la cual se chequean los requerimientos del nombre y la descripción.

### FindById()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: int id

**Función**: Busca y trae la cabaña de interés según su identificador numérico. Retorna un único objeto o una excepción si no se encuentra.

### FindAll()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: no requiere

**Función**: Busca y trae todas las cabañas. Retorna una lista.

### FindCabanaNombre()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: string nombre

**Función**: Busca y trae las cabañas cuyo nombre contenga el string pasado por parámetros (case-insensitive). Retorna una lista de objetos. Se incluyen los datos del Tipo.

### FindCabanaMax()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: int maxPersonas

**Función**: Busca y trae las cabañas cuyo número máximo de huéspedes es mayor o igual al ingresado por parámetros. Retorna una lista de objetos. Se incluyen los datos del Tipo.

### FindCabanaTipo()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: int idTipo

**Función**: Busca y trae las cabañas cuyo id de Tipo asociado coincide con el pasado por parámetros. Retorna una lista de objetos. Se incluyen los datos del Tipo.

### FindCabanasHabilitadas()

**Ubicación**: Datos/Repositorios/RepositorioCabana

**Parámetros**: no requiere

**Función**: Busca y trae las cabañas habilitadas. Retorna una lista de objetos. Se incluyen los datos del Tipo.

## Tipos

### Add()

**Ubicación**: Datos/Repositorios/RepositorioTipo

**Parámetros**: Tipo obj

**Función**: Ejecuta el agregado de un nuevo Tipo de cabaña a la base de datos. Antes de agregar se realiza la validación del objeto con la cual se chequean los requerimientos del nombre y la descripción.

### FindById()

**Ubicación**: Datos/Repositorios/ RepositorioTipo

**Parámetros**: int id

**Función**: Busca y trae el Tipo de interés según su identificador numérico. Retorna un único objeto o una excepción si no se encuentra.

### FindAll()

**Ubicación**: Datos/Repositorios/ RepositorioTipo

**Parámetros**: no requiere

**Función**: Busca y trae todos los Tipos. Retorna una lista.

### FindTipoByNombre()

**Ubicación**: Datos/Repositorios/ RepositorioTipo

**Parámetros**: string nombre

**Función**: Busca y trae el tipo cuyo nombre sea igual al pasado por parámetros (case-insensitive). Retorna un objeto solo o null.

### Remove()

**Ubicación**: Datos/Repositorios/ RepositorioTipo

**Parámetros**: int id

**Función**: Busca y elimina el tipo según el identificador numérico pasado por parámetros. Primero realiza una búsqueda del objeto con FindById y luego se chequea que no esté en uso mediante un recuento de las cabañas que lo utilizan.

### Update()

**Ubicación**: Datos/Repositorios/ RepositorioTipo

**Parámetros**: Tipo obj

**Función**: Actualiza la información del objeto que es pasado por parámetros. Antes de actualizar se vuelve a ejecutar la validación correspondiente.

## Mantenimientos

### Add()

**Ubicación**: Datos/Repositorios/RepositorioMantenimiento

**Parámetros**: Mantenimiento obj

**Función**: Ejecuta el agregado de un nuevo Mantenimiento a la base de datos. Antes de agregar se realiza la validación del objeto con la cual se chequean que la fecha sea posterior a hoy (Validar()) y que no se agreguen más de 3 Mantenimientos con la misma fecha (consulta LinQ).

### FindMantenimientosCabana()

**Ubicación**: Datos/Repositorios/RepositorioMantenimiento

**Parámetros**: int idCabana

**Función**: Busca todos los Mantenimientos de una Cabaña en particular según el identificador número de la Cabaña pasada por parámetros. Retorna una lista.

### FindMantenimientosFechas()

**Ubicación**: Datos/Repositorios/RepositorioMantenimiento

**Parámetros**: DateTime startDate, DateTime endDate, int idCabana

**Función**: Busca todos los Mantenimientos de una Cabaña en particular y entre las fechas ingresadas (inclusive) y las ordena de forma descendiente según su costo. Solo compara las fechas sin incluir las horas. Retorna una lista.

## Parámetros

### ValorParametro()

**Ubicación**: Datos/Repositorios/RepositorioParámetro

**Parámetros**: string nombre

**Función**: Busca el parámetro de interés según el nombre ingresado por parámetros de la función. Selecciona para el retorno solo el valor asociado a ese nombre. Retorna un único valor o null.

# Estructura propuesta

Ver imagen adjunta o el archivo astah-UML.

Diagrama

Descripción generada automáticamente