**Documentación del Diagrama de Clases**

Las clases se han dividido en 3 tipos:

- Entidades:

* **Usuario**

Los atributos con los que cuenta son:

* ID (integer)
* usuario (string)
* contraseña(string)
* nombre(string)
* apellido(string)
* dni(integer)
* direccion(string)
* fecha(date)
* telefono(integer)
* email(string)
* supervisor(boolean)

Los métodos que contiene son:

* +modificar\_clave(): void.

Este método permite modificar la clave del usuario.

* **Admin**

Los atributos con los que cuenta son:

* ID (integer)
* usuario (string)
* contraseña(string)

Los métodos que contiene son:

* +modificar\_clave(): void.

Este método permite modificar la clave del usuario.

* **Reclamos**

Los atributos con los que cuenta son:

* ID (integer)
* resumen(string)
* fecha(date)
* estado(string)

Los métodos que contiene son:

* +generar\_id(): IDint

Este método permite generar el id del reclamo como dato int

* +get\_id(): int

Este método permite acceder al id de reclamo.

* +get\_resumen():string

Este método permite acceder a la string resumen.

* +set\_resumen():string

Este método permite colocar o modificar el string resumen.

* +get\_fecha():date

Este método permite acceder a la fecha.

* +set\_fecha():void

Este método permite colocar o modificar la date fecha.

* +get\_estado():string

Este método permite acceder a la string estado.

* +set\_estado():string

Este método permite colocar o modificar el string estado.

* **Eventos**

Los atributos con los que cuenta son:

* ID(integer)
* resumen(string)
* imagen(string)
* comentarios:(dictionary)

Los métodos que contiene son:

* +generar\_id():ID int

Este método genera la ID correspondiente al evento creado.

* +get\_id():int

Este método permite acceder a la id del evento.

* +get\_resumen():string

Este método permite acceder a él string resumen

* +set\_resumen():void

Este método permite colocar o modificar el string resumen.

* +get\_comentarios():string

Este método permite acceder al diccionario con todos los comentarios.

* +genera\_comentario\_ID():id\_coment int

Este método genera la ID correspondiente a cada uno de los comentarios realizados por cada evento.

* +set\_comentario(comentarios):{}

Este método permite colocar o modificar los comentarios realizados sobre un evento.

* +get\_imagen():string

Este método permite acceder a la string con la ruta correspondiente a la imagen asignada al evento.

* +set\_imagen():void

Este método permite colocar o modificar la ruta de la imagen asignada al evento.

-Controladores

* ControladorEventos

Los métodos que contiene son:

* +obtener\_eventos():void

Este método accede a todos los eventos disponibles.

* +buscar\_evento(ID\_evento):void

Este método busca entre los eventos por el ID correspondiente.

* +crear\_evento(usuario.supervisor):obj

Este método verifica que el usuario esté autorizado para generar crear un evento y en caso de estar habilitado crea dicho evento .

* +modificar\_evento(ID\_evento,usuario.supervisor):boolean

Este método verifica que el usuario esté autorizado para realizar modificaciones y luego realiza las modificaciones solicitadas.

* +eliminar\_evento(ID\_evento,usuario.supervisor):void

Este método verifica que el usuario esté autorizado para eliminar un evento y luego elimina el evento solicitado.

* +modificar\_comentario(ID\_comentario,usuario.supervisor):boolean

Este método verifica que el usuario esté autorizado para realizar modificaciones y luego realiza las modificaciones solicitadas.

* +eliminar\_comentario(ID\_comentario,usuario.supervisor)void

Este método verifica que el usuario esté autorizado para eliminar comentarios y luego elimina el comentario solicitado.

* +buscar\_comentario(comentarios.ID):comentario string

Este método permite la búsqueda de comentarios por medio de su ID.

* +crear\_comentario():json

Este método crea un nuevo comentario en el evento correspondiente.

* ControladorReclamos

Los métodos que contiene son:

* +obtener\_reclamos(usuario.supervisor):void

Este método permite acceder a todos los reclamos.

* +buscar\_reclamo(ID\_reclamo):void

Este método busca un reclamo específico por medio de su ID.

* +borrar\_reclamo(ID\_reclamo):void

Este método elimina un reclamo específico por medio de su ID.

* +modificar\_reclamo(ID\_reclamo):void

Este método modifica un reclamo específico por medio de su ID.

* +crear\_reclamo(usuario.supervisor): obj

Este método crea un reclamo.

* ControladorUsuarios

Los métodos que contiene son:

* +crear\_usuario():obj

Este método crea un usuario nuevo.

* +modificar\_usuario(id\_usuario):void

Este método modifica un usuario por medio de su ID.

* +eliminar\_usuario(id\_usuario):void

Este método elimina un usuario por medio de su ID.

* +obtener\_usuarios():void

Este método perimte acceder a todos los usuarios.

* +buscar\_usuario(ID.usuario):void

Este método busca un usuario específico por su ID.

-Interfaces

* AdministrarUsuarios

Los métodos que contiene son:

* +administrar\_usuarios():void

Este método le permite al admin crear modificar o eliminar a un objeto usuario.

* +mostrar\_usuarios():void

Este método permite mostrar a todos los usuarios o buscar un usuario específico por su ID.

* VistaReclamosSupervisor

Los métodos que contiene son:

* +mostrar\_reclamos():void

Este método permite ver todos los reclamos o buscar un reclamo específico por su ID.

* +borrar\_reclamo(ID\_reclamo,usuario.supervisor):void

Este método elimina un reclamo específico.

* +cambiar\_estado(ID\_reclamo):void

Este método cambia el string estado de un reclamo específico.

* VistaReclamosUsuario

Los métodos que contiene son:

* +mostrar\_reclamos():void

Este método le permite al usuario ver todos los reclamos que realizó.

* +crear\_reclamo(usuario.supervisor):void

Este método le permite crear un reclamo al usuario.

* VistaEventoUsuario

Los métodos que contiene son:

* +mostrar\_eventos():void

Este método muestra al usuario todos los eventos.

* +crear\_comentario():void

Este método le permite crear un comentario al usuario.

* VistaEventoSupervisor

Los métodos que contiene son:

mostrar\_eventos():void

Este método permite mostrar todos los eventos o buscar un evento específico.

+administrar\_eventos(usuario.supervisor):void

Este método le permite al supervisor crear,modificar o eliminar eventos.

+mostrar\_comentarios():void

Este método le permite ver todos los comentarios o buscar un comentario específico.

+administrar\_comentarios(usuario.supervisor):void

Este método le permite al supervisor crear,modificar o eliminar comentarios.

