

## Clases y métodos.

- Usuario
  - Atributos: Nombre, mail, apodo, teléfono y clave son de tipo String.
  - Atributo puntos es una lista de Ints que contiene cada uno de las calificaciones que se le han asignado al usuario.
  - Atributo publicaciones es una lista de objetos Publicacion.
  - Atributo notificaciones es una lista de lista de Strings.
  - Usuario
  - **AgregarCalificacion(int):** *void*; método que permite calificar a otro usuario, cuando es implementado en el main, las notas van de 1 a 5.
  - **AgregarNotificacion(Usuario):** *void*; método implementado para enviar una notificación a un usuario, es enviada por un usuario y recibida por otro; las notificaciones pueden ser vistas y eliminadas en el menú, accediendo a Perfil.
  - **AgregarPublicacion(Publicacion):** *void*; Recibe una publicación y la agrega a la lista de publicaciones.
  - **CalcularPuntos():** *int*; método que calcula el promedio de las calificaciones obtenidas por un usuario.
  - **EliminarNotificaciones():** *void*; metodo que funciona como un “botón” para eliminar las publicaciones de la lista de notificaciones que tiene un usuario.
  - **EliminarPublicacion(int):** *void*; método que permite eliminar una publicación hecha por el usuario, de la lista de publicaciones.
  - **EntregarPublicacion(int):** *Publicacion*; Método que entrega una publicación a partir de su ID.
  - **Getapodo():** *String*; método que retorna el apodo del usuario.
  - **Getcontraseña():** *String*; método que retorna la contraseña del usuario.
  - **Getmail():** *String*; método que retorna el mail del usuario.
  - **MostrarContacto():** *void*; método que muestra toda la información del contacto para poder imprimirla en consola al necesitar saber la información del “dueño” de una publicación o de todos los usuarios.
  - **VerMisPublicaciones():** *void*; permite ver en consola todas las publicaciones del usuario (alumno o personal), mostrandolas todas.
  - **verNotificaciones():** *void*; permite ver en consola todas las notificaciones que tiene el usuario, mostrando el remitente, el mail del remitente y el mensaje de notificación.
- Administrador
  - Administrador
- Alumno
  - Atributo carrera es de tipo string.
  - Alumno
- Personal
  - Atributo cargo es de tipo String.
  - Personal
- Publicación
  - Atributos: Encabezado, comentario y apodo son strings.

- Atributo ID es de tipo int.
- Publicacion
- **GetId():** *int*; método que solo retorna el ID de una publicación determinada.
- **GetApodo():** *String*; método que solo retorna el apodo de quien creó una publicacion.
- **MostrarPublicacion():** *void*; es un método que permite mostrar la información de una publicación, mostrando en consola el ID, Encabezado y Comentario de la publicación.
- Encontrado:
  - Atributo categoría es de tipo string.
  - Encontrado: tiene los detalles de cada publicación en la categoría de Encontrado, cuenta con los mismos métodos y funcionamiento que la clase Perdido; tiene un Encabezado (título de la publicación), Comentario (descripción de la publicación), miCategoría (Si pertenece a Ropa, Librería, etc), un ID (identificador único) y el apodo de quien creo la publicación.
  - **GetCategoria():** *String*; es un Getter que retorna la categoría de un objeto encontrado que fue asignada en la creación de la publicación. Puede ser Ropa, Útiles, Electrónica, Documentos u Otros.
  - **MostrarPublicacion():** *void*; método que hereda la base de la clase madre, publicaciones y que le agrega la categoría de la publicación, usado en VerMisPublicaciones para mostrar toda la información de la publicación.
- Perdido
  - Atributo categoría es de tipo string.
  - Perdido: tiene los detalles de cada publicación en la categoría de Perdido, cuenta con los mismos métodos y funcionamiento que la clase Encontrado; tiene un Encabezado (título de la publicación), Comentario (descripción de la publicación), miCategoría (Si pertenece a Ropa, Librería, etc), un ID (identificador único) y el apodo de quien creo la publicación.
  - **GetCategoria():** *String*; es un Getter que retorna la categoría de un objeto perdido que fue asignada en la creación de la publicación. Puede ser Ropa, Útiles, Electrónica, Documentos u Otros.
  - **MostrarPublicacion():** *void*; método que hereda la base de la clase madre, publicaciones y que le agrega la categoría de la publicación, usado en VerMisPublicaciones para mostrar toda la información de la publicación.
- Main
  - Atributo alumnos, personals, administradores, encontradas y perdidos, son listas de objetos de cada tipo respectivamente (Alumno, Personal, Administrador, Encontrado, Perdido).
  - Atributo cuentaPublicaciones es de tipo Int.
  - **GenerarTXT():** *void*; metodo que crea un archivo .txt para guardar las operaciones que se realizan en el programa.
  - **AdicionarInfo(String):** *void*; metodo que recibe un string para luego guardarlo en un archivo .txt que se genera al iniciar la aplicación. Este método se utiliza cada vez que se quiere guardar una operación que se realizó en el programa.

- **AgregarCalificacion(Usuario):** *void*; Permite calificar otros usuarios en un rango de 1 a 5, es hecha por un usuario hacia otro usuario. Es necesario ingresar el apodo del usuario a evaluar.
- **CrearPublicacion(List<String>, String):** *bool*; método que permite crear las publicaciones, preguntando en consola los datos necesarios para crearla, es necesario que el usuario ingrese el encabezado, comentario, categoría, etc. de la publicación para que pueda ser ingresada y creada correctamente. Las publicaciones creadas, siempre tienen un apodo asociado, que corresponde a un usuario determinado, el apodo es un identificador único de cada usuario.
- **EliminarCualquierPublicacion():** *void*; método que permite eliminar cualquier publicación a partir de su ID, es un método usado por el administrador.
- **EliminarPublicacionUsuario(Usuario):** *void*; método usado por un Usuario para borrar una de sus publicaciones según su ID.
- **eliminarUsuario():** *void*; método usado por el administrador para eliminar un usuario, eliminando todas sus publicaciones también.
- **EntregarUsuario(String):** *Usuario*, metodo que retorna el usuario que tiene el apodo que se ingresa.
- **GetAdministrador(String):** *Administrador*; método que retorna un objeto de tipo administrador buscandolo según el apodo.
- **GetAlumno(String):** *Alumno*; método que retorna un objeto de tipo alumno buscandolo según el apodo.
- **GetPersonal(String):** *Personal*; metodo que retorna un objeto de tipo personal buscandolo según el apodo.
- **guardarAdministrador(Administrador):** *void*; guarda usuarios de tipo Administrador en la lista administradores.
- **guardarAlumno(Alumno):** *void*; Guarda usuarios de tipo Alumno en la lista alumnos.
- **guardarEncontrado(Encontrado, Usuario):** *void*; guarda las publicaciones de tipo encontrado en la lista de encontradas y agrega la publicación al usuario en la lista encontrados.
- **guardarPerdido(Perdido, Usuario):** *void*; guarda las publicaciones de tipo perdido en la lista de perdidos y agrega la publicación al usuario en la lista perdido.
- **guardarPersonal(Personal):** *void*; guarda usuarios de tipo Personal en la lista personals.
- **mostrarInformacion(int):** *void*; permite mostrar la información del contacto que realiza una publicación.
- **RevisarApodo(String):** *bool*; método para verificar que el apodo es único y no está en uso, se revisa al momento de crear un usuario.
- **RevisarMail(String):** *bool*; método para verificar que el mail es único y no está en uso, se revisa al momento de crear un usuario.
- **VerPublicacionesCategoriaE(String):** *void*; Permite ver las publicaciones de objetos por categoría (Ropa, Útiles, etc), en la "categoría" de Encontrados.
- **VerPublicacionesCategoriaP(String):** *void*; Permite ver las publicaciones de objetos por categoría (Ropa, Útiles, etc), en la "categoría" de Perdidos.

- **VerPublicacionesEncontradas():** *void*; Permite ver todas las publicaciones de “categoría” Encontradas. (la diferencia con el anterior, es que este no filtra por categoría como Ropa, solo muestra todas las publicaciones encontradas)
- **VerPublicacionesPerdidas():** *void*; Permite ver todas las publicaciones de “categoría” Pérdidas. (la diferencia con el anterior, es que este no filtra por categoría como Ropa, solo muestra todas las publicaciones pérdidas)
- **verUsuario():** *void*; método que permite ver todos los usuarios de cada tipo, mostrando los usuarios de tipo Alumnos y Personal.

## Uso del programa

Al iniciar el programa al usuario le aparecerá un menú en el cual tendrá la opción de crear un usuario, ingresar o salir del programa. En el caso de crear un usuario, le dará la opción de ser de 2 tipos, alumno o personal, en caso de ser alumno le pedirá los datos necesarios para crearlo en el sistema y revisará con el método RevisarMail y RevisarApodo si es que ya existe algún usuario con ese apodo o ya existir le retornará que ya existe que no lo puede crear y en el caso del mail funciona de la misma forma, si el mail y el apodo no existen el usuario se creará.

La opción dos del menú es ingresar, para esto le pedirá que ingrese el tipo de usuario que es alumno, personal o administrador, luego le pedirá el apodo y verificará con el método RevisarApodo si existe o no el usuario, de no existir le dirá que no existe y volverá al menú principal, si existe le pedirá la contraseña y la verificará con el método GetContraseña de alumno, de ser incorrecta el programa lo indicará y volverá al menú principal, si es correcta el usuario ingresará.

En el caso de ingresar como alumno o personal les aparecerá el mismo menú2, que ingresará 4 opciones y el usuario deberá seleccionar la que quiere, en caso de ser incorrecta la opción le indicará y tendrá que volver a probar.

1. ver mi perfil, al seleccionar esta opción al usuario le aparecerá su apodo y calificación y le aparecerá un menú en el cual tiene seis opciones, si ingresa una opción mal le indicará y volverá al menú 2:
  - a. ver sus notificaciones, muestra todas las notificaciones del usuario y le pregunta si quiere borrarlas o no, si ingresa algo incorrecto le indicará y volverá al menú 2, si ingresa no volverá al menú 2 y si ingresa si borrará todas las notificaciones del usuario.
  - b. enviar notificación a un usuario, le pedirá el apodo del usuario a quien quiere enviar la notificación, de no existir le indicará y volverá al menú 2, de existir le pedirá el texto de la notificación y le enviará a la notificación entregando un mensaje de éxito.
  - c. calificar a otro usuario, le pedirá el apodo del usuario a quien quiere calificar, de no existir le indicará y volverá al menú 2, de existir le pedirá la nota con que lo quiere calificar, de 1 a 5, de ingresar mal un número le indicará y volverá al menú 2, de ingresarla bien le indicará y volverá al menú 2.
  - d. ver mis publicaciones, le mostrará al usuario todas las publicaciones que ha creado. de no tener notificaciones le indicará.
  - e. eliminar una publicación, le pedirá ingresar el id de la publicación que quiere eliminar, de no existir esa publicación le indicará y volverá al menú 2, si existe la borrará y le indicará al usuario que se ha eliminado.
  - f. volver, vuelve de la opción perfil a el menú2, en caso de no querer realizar ninguna de las opciones anteriores.
2. Crear publicación, le permite al usuario crear una publicación, para esto le pide que ingrese un encabezado y un comentario, si es de tipo perdido o encontrado y a que categoría de las que se muestran pertenece, de no existir la categoría le indicará y volverá al menú 2, si existe la categoría la publicación se creará y le indicará un mensaje de éxito.

3. ver publicaciones, le aparecerán dos opciones, ver las encontradas o ver las pérdidas, si ingresa algo mal le indicará y volverá al menú 2, si ingresa bien le mostrará 3 opciones
  - a. ver todas las de encontrado/perdido, que muestran todas las publicaciones de encontrado/perdido,
  - b. ver publicaciones por categoria, te pide ingresar a qué categoría de las que se muestran pertenece, de no existir la categoría le indicará y volverá al menú 2, de existir te mostrará todas las publicaciones de la categoría indicada.
  - c. Mostrar contacto de una publicación, te pedirá el id de la publicación que quiere ver su contacto, si ingresa mal el id le indicará y volverá al menú 2, si lo ingresa bien le mostrará el nombre, apodo, mail, calificación de la persona que publicó esa publicación.

4. Salir, vuelve al menú principal.

En caso de ser un administrador le aparecerá un menú 2 diferente, en este tendrá 5 opciones, de ingresar una mal le indicará y volverá al menú 2:

1. ver publicaciones, le aparecerán dos opciones, ver las encontradas o ver las pérdidas, si ingresa algo mal le indicará y volverá al menú 2, si ingresa bien le mostrará 3 opciones
  - a. ver todas las de encontrado/perdido, que muestran todas las publicaciones de encontrado/perdido,
  - b. ver publicaciones por categoria, te pide ingresar a qué categoría de las que se muestran pertenece, de no existir la categoría le indicará y volverá al menú 2, de existir te mostrará todas las publicaciones de la categoría indicada.
  - c. Mostrar contacto de una publicación, te pedirá el id de la publicación que quiere ver su contacto, si ingresa mal el id le indicará y volverá al menú 2, si lo ingresa bien le mostrará el nombre, apodo, mail, calificación de la persona que publicó esa publicación.
2. Eliminar publicaciones, le pedirá ingresar el id de la publicación que quiere eliminar, de no existir esa publicación le indicará y volverá al menú 2, si existe la borrará y le indicará al usuario que se ha eliminado. El administrador puede eliminar cualquier publicación.
3. Eliminar usuario, le pedirá ingresar el apodo del usuario que quiere eliminar, de no existir ese usuario le indicará y volverá al menú 2, si existe borrará al usuario y todas sus publicaciones, le indicará al administrador que se ha eliminado el usuario indicado.
4. ver usuarios, le muestra todos los usuarios creados en el sistema.
5. Salir, vuelve al menú principal.