

## Tarea 3

Se ha realizado una aplicación que emula un perfil de una red social. Los comandos que se pueden emplear desde su ventana principal son:

- {Ir | Navegar} a [Nombre de usuario] (por ejemplo, Vicente).
  - La acción nos lleva al perfil del usuario seleccionado.
- {Ocultar | Mostrar} foto.
  - La acción muestra u oculta la foto de perfil.
- {Aumentar | Reducir} {letra | fuente | tamaño de letra | tamaño de fuente}.
  - Aumenta o reduce la fuente de la etiqueta del nombre de usuario.
- Volver/Atrás/Retroceder.
  - En caso de haber abierto el perfil de otro usuario, retrocede al anterior visitado.
- {Escribir | Añadir | Dictar comentario}
  - Comienza una secuencia de dictado que termina cuando el usuario de la aplicación dice “Fin dictado”.
- Borrar comentario
  - Elimina el texto escrito para añadir un comentario.

Comentarios adicionales:

- Además del formulario principal, se han creado otros dos: uno para iniciar sesión con el usuario indicado y otro para crear usuarios nuevos. Para hacer pruebas, se ha dejado para que la aplicación se inicialice desde el formulario principal con el primer usuario encontrado, pero si se desea, este comportamiento se puede modificar desde la clase Program:

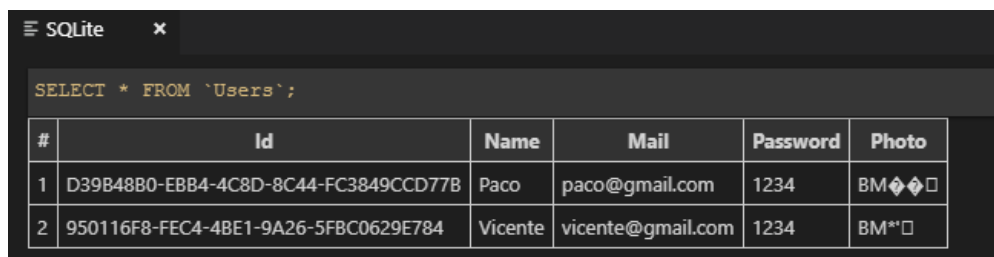
```

---try
---{
---    // Uncomment this code to initialize the application from the Login form.
---    //Application.Run(new LoginForm(serviceProvider));

---    // Code to initialize the application with the first user registered.
---    Application.Run(
---        new ProfileForm(
---            serviceProvider,
---            serviceProvider.CreateScope().ServiceProvider
---            .GetService<PersistenceContext>().Users.Include(u => u.OwnerPosts).FirstOrDefault());
---    }

```

- La aplicación hace uso de una BD Sqlite que se crea dentro de la carpeta “DIM\_Voice\bin\Debug”. En caso de que el fichero no exista, se crea automáticamente añadiendo dos usuarios: Vicente y Paco (gracias a la combinación de los métodos de EF Core HasData() y EnsureCreated()). La base de datos se puede inspeccionar, por ejemplo, empleando la extensión de “SQLite” de Visual Studio Code.



The screenshot shows the SQLite extension interface in Visual Studio Code. At the top, there's a tab labeled 'SQLite' and a close button. Below it, the SQL query 'SELECT \* FROM `Users`;' is entered. The results are displayed in a table with 6 columns: '#', 'Id', 'Name', 'Mail', 'Password', and 'Photo'.

#	Id	Name	Mail	Password	Photo
1	D39B48B0-EBB4-4C8D-8C44-FC3849CCD77B	Paco	paco@gmail.com	1234	BM♦♦□
2	950116F8-FEC4-4BE1-9A26-5FBC0629E784	Vicente	vicente@gmail.com	1234	BM**□

- Existe un proyecto de arranque adicional: DictationTest. En este proyecto se han hecho pruebas para emular el dictado por voz a un text box. La IU es mucho más simple. Los comandos que aquí se pueden emplear son los dos últimos de los descritos en la aplicación.