Tecnologías utilizadas.

.

Retrofit2



Retrofit es la seguridad de tipos de HTTP client para Android y Java. El Retrofit facilita la conexión al servicio web REST mediante la traducción de API a interfaces Java.

Java



Java es un **lenguaje de programación** utilizado para crear software compatible con una gran diversidad de sistemas operativos.

Spring boot

Java Spring Framework (Spring Framework) es un popular marco de trabajo empresarial de código abierto que sirve para crear aplicaciones autónomas de producción que se ejecutan en una máquina virtual Java (JVM).

Github



GitHub es un sitio "social coding". Te permite subir repositorios de código para almacenarlo en el sistema de control de versiones Git.

PostgreSQL



PostgreSQL es un sistema de código abierto empleado para gestionar bases de datos de tipo relacional, aquellas que administran datos relacionados entre sí, esto es, tablas integradas por registros (filas) y campos (columnas) en la que cada registro tiene una única identidad, conocida como clave.

Requisitos funcionales

* + **Registro de Usuario**

La aplicación debe permitir a los usuarios registrarse y crear una cuenta con su información personal, incluyendo nombre de usuario, dirección de correo electrónico ,contraseña y una foto de perfil.

* + **Inicio de Sesión**

La aplicación debe permitir a los usuarios iniciar sesión en su cuenta utilizando su dirección de correo electrónico o nombre de usuario y contraseña.

* + **Perfil de Usuario**

La aplicación debe permitir a los usuarios actualizar y editar su información de perfil, incluyendo su nombre, correo electrónico y la foto de perfil.

* + **Búsqueda de Usuarios y Comunidades**

La aplicación debe permitir a los usuarios buscar usuarios y comunidades por nombre.

* + **Comentarios**

La aplicación debe permitir a usuarios dejar comentarios en cualquier publicación

* + **Notificaciones**

La aplicación debe permitir a los usuarios recibir notificaciones sobre nuevos mensajes enviados por sus amigos,avisos de nuevas publicaciones de amigos y la cantidad de “me gustas” de sus publicaciones.

Requisitos no funcionales

* **Fiabilidad**

Se debe evitar que el usuario experimente fallos en la aplicación por sesión.

* **Eficiencia**

La aplicación debe mostrar información que pide el usuario y mostrarla en menos de 5 o 10 segundos.

La aplicación no debe consumir mucha capacidad de almacenamiento ni batería,tampoco debe ralentizar el móvil.

* **Rendimiento**

La aplicación debe ser rápida y responder sin contratiempos, en cualquier condición de conectividad de red o en cualquier dispositivo móvil.

* **Seguridad**

La contraseña debe tener un mínimo de 8 caracteres,máximo 15,al menos una letra mayúscula, al menos un dígito y sin espacios en blanco.

La aplicación debe garantizar la seguridad de los datos del usuario, utilizando técnicas de cifrado para el uso de contraseñas.

* **Disponibilidad**

La aplicación debe estar disponible y accesible para los usuarios en todo momento, sin interrupciones ni tiempo de inactividad prolongado.

* **Escalabilidad**

La aplicación debe ser escalable y capaz de manejar grandes volúmenes de tráfico y usuarios sin afectar negativamente el rendimiento.

* **Compatibilidad**

La aplicación debe ser compatible con una amplia gama de dispositivos móviles y tablets, debe funcionar tanto en Android como en IOS.

* **Usabilidad**

La aplicación debe ser fácil de usar y navegar para los usuarios, con una interfaz de usuario intuitiva y tener un diseño visualmente atractivo y simple.

Informar al usuario cuando se ha producido un error.

Permite recuperar la cuenta en el caso de que se haya olvidado de su nombre de usuario o contraseña.

Permitir al usuario cambiar los datos de su cuenta en cualquier momento.

* **Mantenibilidad**

La aplicación debe ser fácil de mantener y actualizar, con un código bien estructurado y documentado, para permitir cambios y correcciones de errores de manera eficiente.

* **Privacidad**

La aplicación debe cumplir con las leyes y regulaciones de privacidad de datos aplicables, y proporcionar opciones claras y transparentes para que los usuarios controlen sus datos personales.

Descripción casos de uso

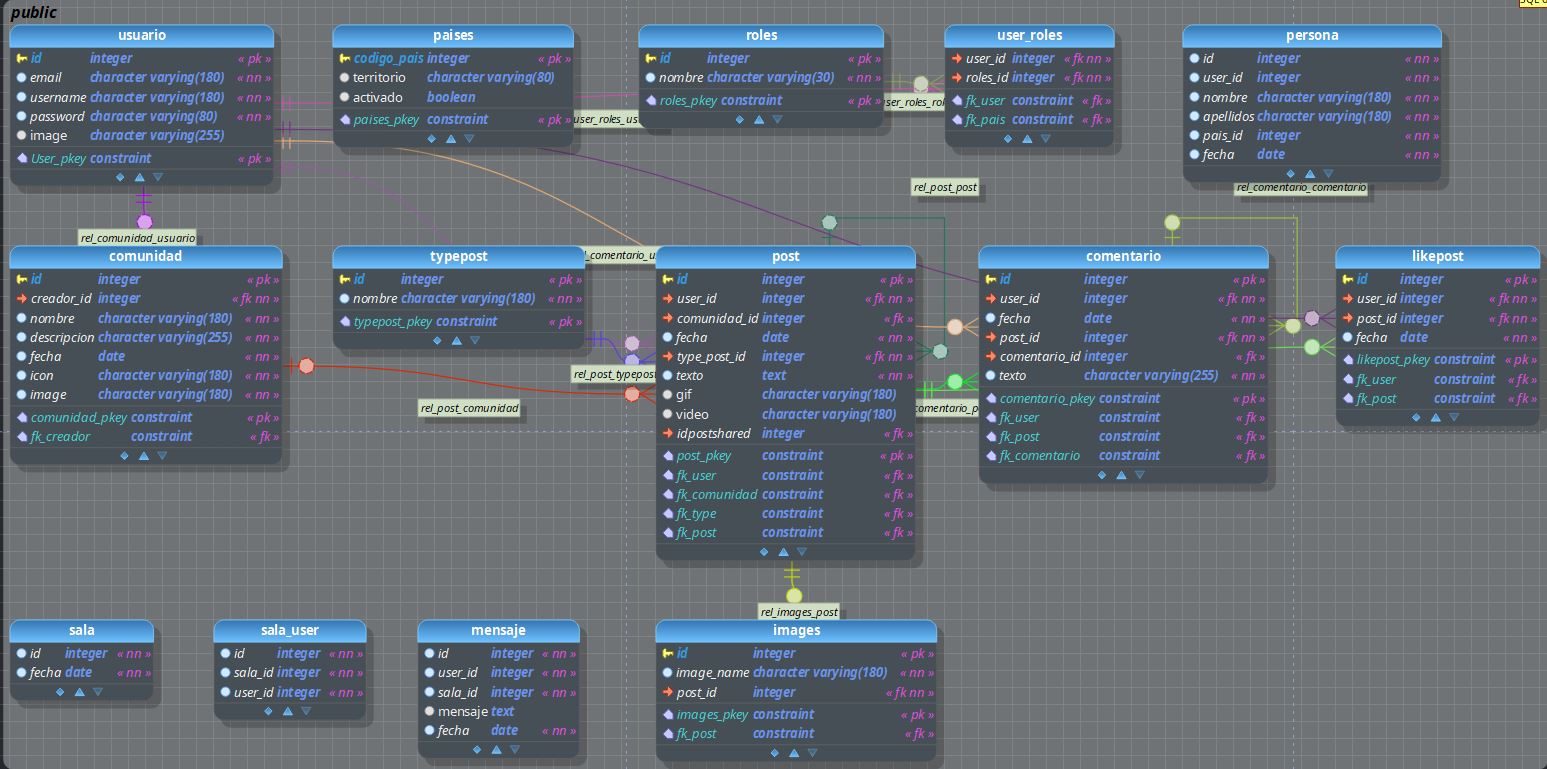
| CU-01 | Login | | |
| --- | --- | --- | --- |
| PreCondición | Poseer una cuenta | | |
| Descripción | El sistema deberá comprobar que los datos introducidos en un formulario son correctos. | | |
| Secuencia normal | Paso | Acción | |
| 1 | El sistema solicita introducir el usuario y la contraseña | |
| 2 | El usuario introduce el usuario y la contraseña | |
| 3 | El usuario hace clic en el botón “Log in”. | |
| 4 | El sistema verifica que el usuario y la contraseña sean correctas. | |
| PostCondición | Se pasa de la actividad de login a la actividad principal | | |
| Excepciones | Paso | Acción | |
| 1 | Si el usuario introducido no es correcto | |
| E.1 | El sistema informa de la situación |
| E.2 | El sistema impide que entres a la aplicación |

| CU-02 | Registro | | |
| --- | --- | --- | --- |
| PreCondición |  | | |
| Descripción | El sistema deberá comprobar que los datos introducidos en un formulario son correctos. | | |
| Secuencia normal | Paso | Acción | |
| 1 | El sistema solicita introducir un nombre de usuario,contraseña,correo electrónico y una imagen. | |
| 2 | El usuario introduce toda la información que le pide la aplicación | |
| 3 | El usuario hace clic en el botón “Sign up”. | |
| 4 | El sistema verifica que el usuario y la contraseña sean correctas. | |
| PostCondición | Se pasa de la actividad de registro a la actividad principal | | |
| Excepciones | Paso | Acción | |
| 1 | Si el usuario no es válido o ya existe | |
| E.1 | El sistema informa de la situación. |
| E.2 | El sistema impide que puedas completar la creación de la cuenta |

| CU-03 | Búsqueda | | |
| --- | --- | --- | --- |
| PreCondición | Haber iniciado sesión con un usuario | | |
| Descripción | El sistema buscará usuarios según lo introducido en un campo de texto | | |
| Secuencia normal | Paso | Acción | |
| 1 | El sistema solicita introducir un nombre de usuario | |
| 2 | El usuario introduce toda la información que le pide la aplicación | |
| 3 | El usuario hace clic en el botón “Buscar”. | |
| 4 | El sistema muestra todos los resultados que haya encontrado | |
| PostCondición | El usuario podrá ver el perfil del usuario que haya seleccionado | | |
| Excepciones | Paso | Acción | |
| 1 | Si el usuario no existe | |
| E.1 | El sistema informa de la situación. |

| CU-04 | Añadir publicación | | |
| --- | --- | --- | --- |
| PreCondición | Haber iniciado sesión con un usuario | | |
| Descripción | Proceso para añadir una publicación a la aplicación | | |
| Secuencia normal | Paso | Acción | |
| 1 | Ingresar en la actividad para crear una nueva publicación | |
| 2 | El sistema solicitará una serie de datos en un formulario | |
| 3 | Seleccionar un tipo de publicación(Imagen,video,Gif) | |
| 4 | El usuario rellena el formulario | |
|  | 5 | El usuario hace clic sobre “Crear” | |
| PostCondición | El sistema insertará la publicación en la base de datos y estará accesible para todos los usuarios. | | |
| Excepciones | Paso | Acción | |
| 1 | En caso de que algún campo no sea válido | |
| E1 | El sistema informará del error ocurrido |
| E2 | La publicación no se insertará en la base de datos |

Modelo de datos



La tabla User es una entidad que almacena todos los usuarios de la aplicación.

* id:Identificador del usuario,es un valor autoincremental.
* UserName:Nombre de la cuenta.
* password:Contraseña del usuario.
* email:Correo Electrónico del usuario.
* image:Avatar del usuario.

La tabla persona tiene una relación OneToOne con la entidad User donde contiene los datos personales de cada usuario.

* id:Identificador de la tabla,valor autoincremental
* Nombre:Campo que indica el nombre del usuario.
* Apellidos:Campo que indica el apellido del usuario.
* Fecha Nacimiento:Campo que indica la fecha de nacimiento del usuario.
* user\_id:Clave Clave secundaria cuyo valor corresponde a la clave primaria de la entidad User
* pais\_id:Clave secundaria cuyo valor corresponde a la clave primaria de la entidad País.

La tabla de Comunidad guardará toda la información de las comunidades la aplicación:

* id:Identificador de la tabla,valor autoincremental.
* Nombre:El nombre de la comunidad.
* Fecha:Almacena la fecha de creación de la comunidad..
* Descripción:Campo descriptivo.Contiene una breve definición de la comunidad.
* icon:Almacena el nombre del avatar de la comunidad.
* imagen:Almacena la imagen de la comunidad.
* creador\_id:Clave Clave secundaria cuyo valor corresponde a la clave primaria de la entidad User.

La tabla de Post guardará toda la información relacionada con las publicaciones que suban los usuarios:

* id:Identificador de la tabla,valor autoincremental.
* Fecha:Almacena la fecha de la subida de la publicación.
* Texto:Campo que almacena el mensaje del post.
* Video:Campo que almacena el nombre del video si el post es de tipo “Video”.
* Gif:Campo que almacena el nombre del gif si el post es de tipo “Gif”.
* Tipo:Campo numérico.Contiene el tipo de la publicación.
* user\_id:Clave Clave secundaria cuyo valor corresponde a la clave primaria de la entidad User
* comunidad\_id:Clave secundaria cuyo valor corresponde a la clave primaria de la entidad Comunidad.

La tabla TypePost contiene información sobre de qué tipo puede ser una publicación:

* id:Identificador de la tabla,valor autoincremental
* Nombre:Campo que indica el nombre del usuario.

La entidad images contiene todas las imágenes de las publicaciones:

* id:Identificador de la tabla,valor autoincremental.
* imageName: Campo que contiene el nombre nombre de la imagen.
* post\_id:Clave secundaria cuyo valor corresponde a la clave primaria de la entidad Post.

La tabla de Comentario guardará toda la información relacionada con las comentarios que suban los usuarios:

* id:Identificador de la tabla,valor autoincremental.
* Fecha:Almacena la fecha de la subida del comentario.
* Texto:Campo que almacena el mensaje del post.
* user\_id:Clave Clave secundaria cuyo valor corresponde a la clave primaria de la entidad User.
* post\_id:Clave secundaria cuyo valor corresponde a la clave primaria de la entidad Post.
* comentario\_id:Clave secundaria cuyo valor corresponde a la clave primaria de la entidad Comentario.

Guia de estilos

Tipografía

Colores

* #5f5fb0
* #262626
* #efefef
* #9596a3
* #747383
* #7c7c7c
* #c5c5c8
* #d4d4df
* #6c74cc
* #151515
* #bfbfc2
* #7b83f2
* #343434
* #6b6b6e
* #3b3c66
* #848cf4
* #8c84a4
* #24243d

Iconos