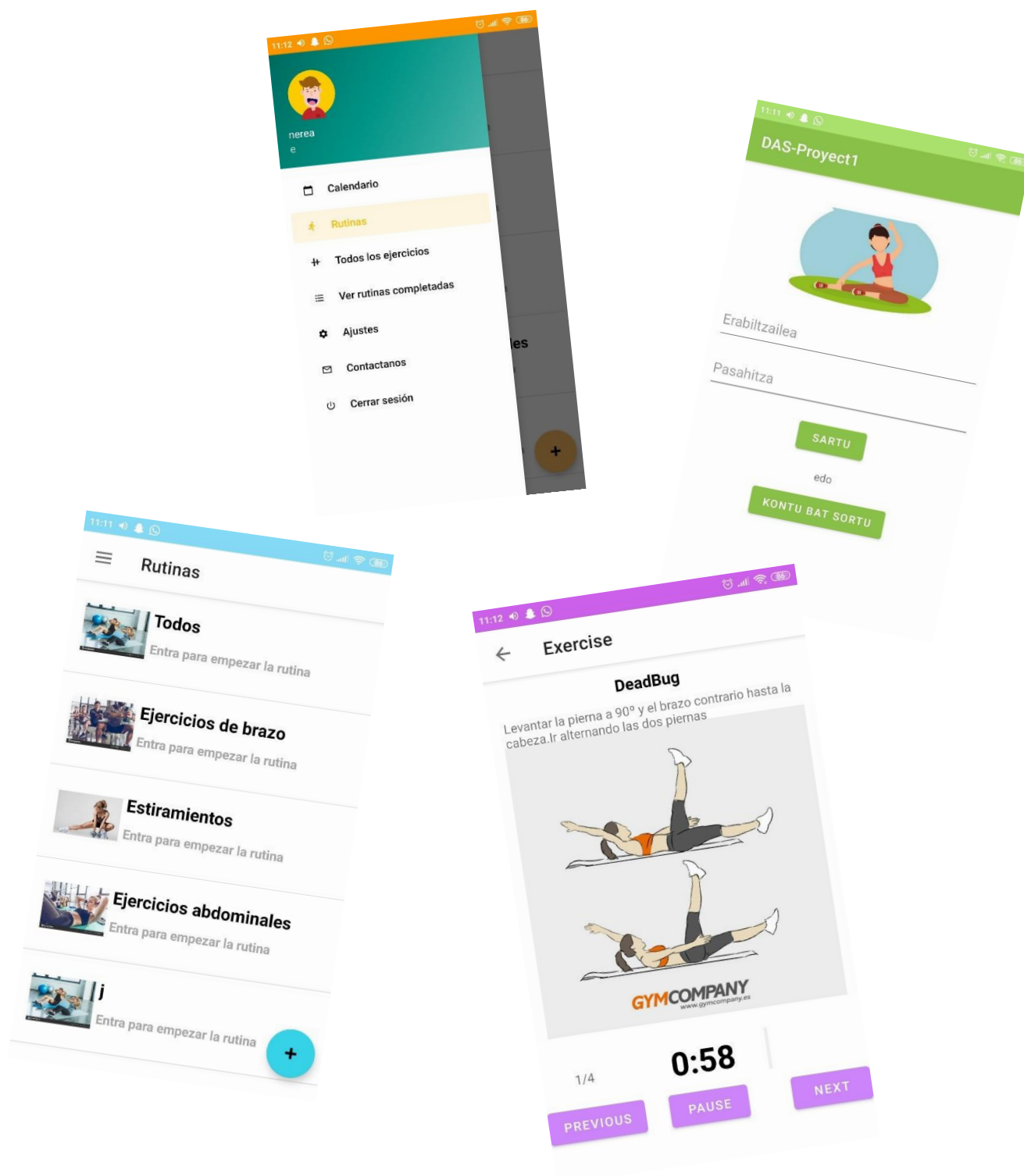


# DAS - ENTREGA INDIVIDUAL 1



Nerea Monasterio  
Curso 2020-2021  
Fecha de entrega: 21/03/2021

# ÍNDICE

<b>1- Descripción breve.....</b>	<b>3</b>
<b>2- Repositorio Git de la aplicación.....</b>	<b>3</b>
<b>3- Explicación de cada clase.....</b>	<b>3</b>
<b>3- Manual de usuario .....</b>	<b>8</b>
3.1- INICIAR SESIÓN.....	8
3.2- CREAR USUARIO .....	9
3.3- PAGINA PRINCIPAL AL INICIAR SESION - CALENDARIO .....	10
3.4- RUTINAS .....	11
3.5- RUT EJERCICIO .....	12
3.6- EJERCICIOS.....	13
3.7- RUTINAS COMPLETADAS .....	13
3.8- AJUSTES .....	14
3.10- CONTACTANOS .....	16
3.11- LOGOUT.....	16
<b>4 - Explicación de cómo se cumple con cada uno de los requisitos .....</b>	<b>17</b>
4.1- Uso de Listviews personalizados o de RecyclerView+CardView para mostrar listados de elementos con diferentes características.....	17
4.2-Usar una base de datos local, para listar, añadir y modificar elementos y características de cada elemento. ....	17
4.3- Uso de diálogos. ....	19
4.4- Usar notificaciones locales.....	20
4.5- Controlar que la aplicación se comporte correctamente y no se pierda información, aunque en medio de la ejecución recibamos una llamada telefónica, giremos el móvil, etc. No se puede impedir que la aplicación se gire. Controlar la pila de actividades para evitar incoherencias.....	20
4.6- Documentar la aplicación mediante comentarios en el propio código. ....	21
4.7- Permitir que una misma funcionalidad se comporte de manera distinta dependiendo de la orientación (o del tamaño) del dispositivo mediante el uso de Fragments (0,5 puntos). ....	21
4.8- Hacer la aplicación multiidioma y añadir la opción de cambiar de idioma en la propia aplicación (0,5puntos). ....	22
4.9- Uso de ficheros de texto (1 punto).....	22
4.10- Uso de Preferencias, para guardar las preferencias del usuario en cuanto a mostrar/esconder cierta información, elegir colores para la aplicación, o cualquier otra cosa relacionada con la visualización de la aplicación (1 punto).....	23
4.11- Crear estilos y temas propios, para personalizar fondos, botones, etc. (0,5 puntos).....	23
4.12- Usar intents implícitos para abrir otras aplicaciones, contactos, etc. (0,5 puntos). ....	24
4.13- Pantalla de login (y registro), para guardar credenciales de usuario (nombre de usuario – contraseña) en la base de datos local (1 punto). ....	24

## 1- Descripción breve

He creado una aplicación de rutinas de ejercicios. Para acceder a la app será necesaria la creación de una cuenta indicando usuario, email y contraseña. Una vez dentro de la app encontraremos un menú, que nos permitirá navegar por las diferentes ventanas que se ofrecen. Las ventanas disponibles son: un calendario que visualiza las rutinas completadas en el día seleccionado, otra ventana que muestra todas las rutinas disponibles para hacer, donde haciendo clic sobre una de ellas nos llevará a los ejercicios a realizar, otra ventana que muestra todos los ejercicios que podemos encontrar, otra que nos dice todas las rutinas que hemos completado hasta el momento, y otra donde podremos configurar varios ajustes.

## 2- Repositorio Git de la aplicación

La aplicación se encuentra disponible en: <https://github.com/nereamonas/DAS-Proyecto1>

## 3- Explicación de cada clase

Estos son los archivos del proyecto:

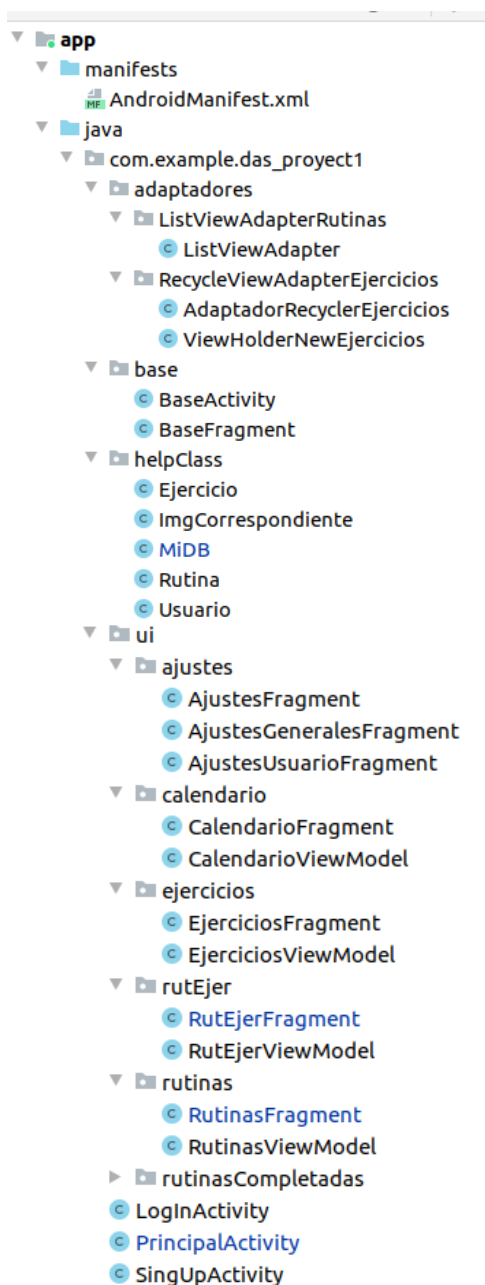


Imagen 1: Clases del proyecto 1

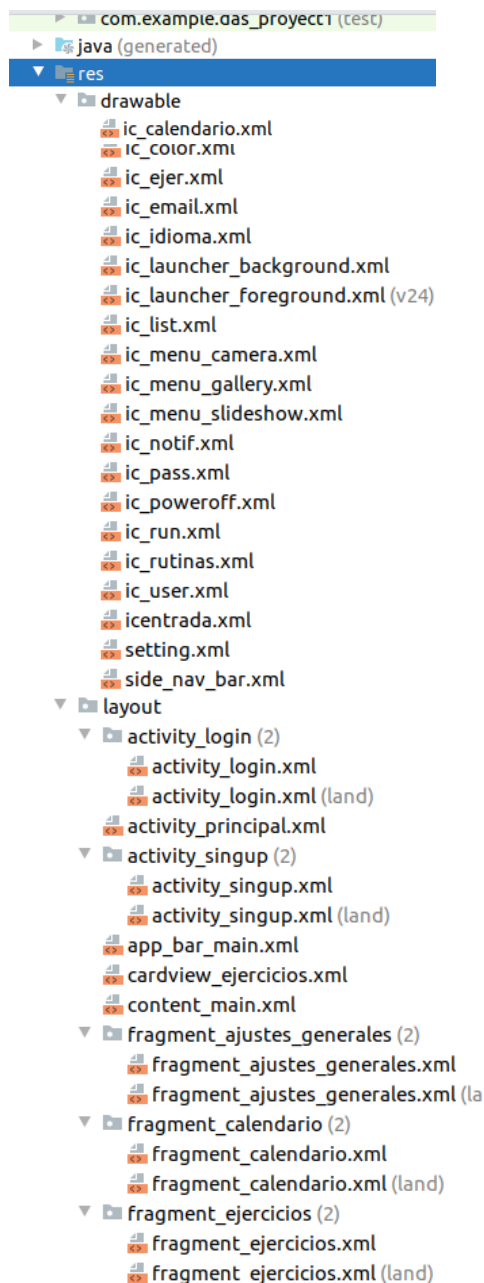


Imagen 2: Clases del proyecto 2

- ▼ fragment\_rutejerviewpager (2)
  - fragment\_rutejerviewpager.xml
  - fragment\_rutejerviewpager.xml (lar)
- ▼ fragment\_rutinas (2)
  - fragment\_rutinas.xml
  - fragment\_rutinas.xml (land)
  - fragment\_rutinas\_completadas.xml
  - listviewelement\_rutinas.xml
  - nav\_header\_main.xml
- ▼ menu
  - activity\_main\_drawer.xml
  - principal.xml
- ▼ mipmap
  - ▼ avatar (5)
    - avatar.png (hdpi)
    - avatar.png (mdpi)
    - avatar.png (xhdpi)
    - avatar.png (xxhdpi)
    - avatar.png (xxxhdpi)
  - ▶ bicyclecrunch (5)
  - ▶ deadbug (5)
  - ▶ giroruso (5)
  - ▶ ic\_launcher (6)
  - ▶ ic\_launcher\_round (6)
  - ▶ icono (5)
  - ▶ mountainclimber (5)
  - ▶ onearmtoetouchcrunch (5)
  - ▶ plancha (5)
  - ▶ planchaflexion (5)
  - ▶ reversecrunch (5)
  - ▶ rollingplank (5)
  - ▶ rutinaejerabdominal (5)
  - ▶ rutinaejerbrazo (5)
  - ▶ rutinaestiramiento (5)
  - ▶ sicsorsabs (5)
  - ▶ todos (5)
- ▼ navigation
  - mobile\_navigation.xml
- ▼ values
  - arrays.xml
  - colors.xml
  - dims.xml
  - ic\_launcher\_background.xml
  - ▼ strings (4)
    - strings.xml
    - strings.xml (en)
    - strings.xml (es)
    - strings.xml (eu)
  - ▼ themes (2)
    - themes.xml
    - themes.xml (night)
- ▼ xml
  - fragment\_ajustes.xml
  - fragment\_ajustes\_usuario.xml
- res (generated)
- Gradle Scripts

Imagen 3: Clases del proyecto 3

En el código de la aplicación está mejor explicado por pasos todas las acciones que realiza cada clase. Pero aquí un breve resumen de lo que hace cada una en general:

- ListAdapter → Es la clase adaptadora que se encargará de gestionar la ListView de las rutinas. Contendrá un ArrayList de rutinas.
- AdaptadorRecyclerEjercicios → Esta clase es el adaptador para crear la RecyclerView de todos los ejercicios. Será un ArrayList de ejercicios.
- ViewHolderNewEjercicios → Guardaremos los elementos que tiene cada elemento del RecyclerView. En este caso un título y una imagen.
- BaseActivity → Vamos a crear esta clase que será la base de las actividades. Cada vez q se cree una activity, en vez d heredar directamente de AppCompatActivity heredara de BaseActivity, así podemos editar el super.onCreate() y meter las modificaciones que necesitemos. En este caso, queremos gestionar el tema y el idioma. Lo que conseguiremos es, que al apagar la aplicación y volver a arrancarla, se mantengan todas las configuraciones que ha indicado el usuario en las preferencias o ajustes. Haciendo que estas preferencias se apliquen a todas las actividades cuando se crean.
- BaseFragment → Vamos a crear esta clase que será la base de los fragmentos. Cada vez q se cree un fragment, en vez d heredar directamente de Fragment heredara de BaseFragment, así podemos editar el super.onCreate() y meter las modificaciones que necesitemos. En este caso, queremos gestionar el tema y el idioma. Lo que conseguimos es, que al apagar la aplicación y volver a arrancarla, se mantengan todas las configuraciones que ha indicado el usuario en las preferencias. Haciendo que estas preferencias se apliquen a todos los fragmentos cuando se creen.
- Ejercicio → Clase de ayuda donde se guarda la información de un ejercicio. Su id, nombre, descripción, foto y duración.
- ImgCorrespondiente → Las imágenes se almacenan en R.mipmap. Pero esos elementos devuelve un numero identificador, y no puedo concatenar el nombre de la bd con R.mipmap para q m de directamente el numero que corresponde. Por eso he creado esta clase como ayuda para gestionarlo. Aunque no sería la manera más correcta.
- MiDB → Crearemos la conexión con la base de datos y se implementarán todos los métodos necesarios a lo largo de la aplicación que realicen select, insert, update o delete en la base de datos.
- Rutina → Clase de ayuda donde se guarda la información de una rutina. El id, nombre y referencia a la foto.
- Usuario → Clase de ayuda donde se guarda la información de un usuario. El nombre de usuario, email y contraseña.
- AjustesFragment → Clase donde se administran los ajustes principales de la aplicación. Qué hacer cuando ocurre un cambio de idioma, de tema y de activación o desactivación de las notificaciones.
- AjustesGeneralesFragment → Esta clase une el ajustesFragment y ajustesUsuarioFragment. Si está en vertical solo muestra ajustesFragment y si está en horizontal muestra los dos.
- AjustesUsuarioFragment → Clase donde se administran los ajustes del usuario. Qué hacer en el caso de que se modifique el nombre de usuario, email o la contraseña.
- CalendarioFragment → Administra el fragmento de calendario, su función es que cuando se haga clic en un día dentro del calendario, se muestre en la parte inferior las rutinas que ha completado el usuario en el día seleccionado, para mostrar esta lista cogerá de la memoria interna del teléfono, un fichero donde se irán guardando las rutinas que se han completado:
- CalendarioViewHolder → Todas estas clases ViewHolder, se añaden al crear un fragmento que pertenecerá al menú. Sirven para que, al entrar en el fragment, salgan las tres rallas que dan la opción a abrir el menú.
- EjerciciosFragment → Aquí se mostrarán todos los ejercicios que hay añadidos en la base de datos para ese usuario mediante un RecyclerView, para ello se ayudará de la clase AdaptadorRecyclerEjercicios.

- RutEjerFragment → Esta clase es la más grande y la que se encarga de gestionar más cosas. Cogerá de la base de datos todos los ejercicios que hay para la rutina seleccionada. Creará un temporizador con el tiempo de duración de cada ejercicio (que se indica en base de datos). Cuando el tiempo acabe, automáticamente saltará al siguiente ejercicio. Aunque este movimiento también se puede controlar con los botones que se encuentran en la parte inferior: 'Anterior', 'Pause' y 'Siguiente'. Al pasar de ejercicio saltará un toast si dichas notificaciones están activadas. Al terminar con todos los ejercicios realizará diferentes acciones: nos redirige a la ventana rutinas, salta un toast y una notificación local si estas notificaciones se encuentran activadas y escribe en un fichero externo en la memoria interna del telefono que se ha terminado la rutina:
- RutinasFragment → Creará una ListView de rutinas, que al pulsar en una te lleva al fragment RutEjerFragment, pasándole la rutina que el usuario ha seleccionado. También se puede eliminar una rutina haciendo una pulsación larga sobre ella. Y también se puede crear una nueva rutina mediante el botón + de la parte inferior derecha. Donde nos saltará un dialogo para añadir el nombre que le queremos dar a la rutina y posteriormente otro dialogo para seleccionar los ejercicios que queramos añadir.
- RutinasCompletadasFragment → Sera una lista, que muestra todas las rutinas que ha completado el usuario. Nombre de la rutina + fecha + hora. Esta información la coge del fichero externo que se genera en la memoria interna del telefono. Funciona igual que el calendario, pero cogiendo todas las fechas.
- LogInActivity → La actividad que se encarga de gestionar el inicio de sesión, si los datos introducidos por el usuario son correctos en base de datos, redirige a la ventana principal. En caso contrario muestra un mensaje de error.
- PrincipalActivity → Esta es la clase principal que se encarga de gestionar los fragmentos que pertenecen a ella. Nos indicará a que fragment hay que navegar o que hacer al hacer clic sobre cada elemento del menú.
- SingUpActivity → La actividad que se encarga de gestionar la creación de un usuario. Si los datos introducidos por el usuario son correctos y se puede crear una cuenta, redirige a la ventana principal. En caso contrario muestra un mensaje de error.
- Carpeta drawable → Aquí he creado muchos iconos que se usarán en la aplicación, el icono de email, de cerrar sesión, de ajustes, de idioma...
- Carpeta layout → Aquí guardaremos todas las plantillas que usaremos en la aplicación:
  - o Activity\_login.xml → Plantilla de la página de login (LogInActivity.java).
  - o Activity\_principal.xml → Plantilla de la ventana principal (PrincipalActivity.java).
  - o Activity\_singup.xml → Plantilla de la ventana para crear una nueva cuenta (SingUpActivity.java).
  - o App\_bar\_main.xml → Plantilla del principal. Que tendrá en la parte superior un toolbar.
  - o Cardview\_ejercicios.xml → Plantilla de un elemento del cardview. Mostrará el título de un ejercicio con su foto correspondiente.
  - o Content\_main.xml → Plantilla del principal. Que contiene un fragmento que irá rotando según el que elijamos abrir en el menú. Se usará un navigation\_host.
  - o Fragment\_ajustes\_generales → Plantilla que muestra los fragmentos de ajustes (AjustesGeneralesFragment.java).
  - o Fragment\_calendario.xml → Plantilla que muestra el calendario con el textView de las rutinas completadas ese día (CalendarioFragment).
  - o Fragment\_ejercicios.xml → Plantilla que muestra un recycleView donde cada elemento será un cardview. Mostrará todos los ejercicios disponibles (EjerciciosFragment).
  - o Fragment\_rutejerviewpager.xml → Plantilla que muestra la información de un ejercicio, su título, descripción, foto, botones.... (RutEjerFragment.java).
  - o Fragment\_rutinas.xml → Plantilla que muestra una listView con todas las rutinas (RutinaFragment.java).

- Fragment rutinas completadas.xml → Plantilla con un textView que se mostrarán todas las rutinas completadas por el usuario (RutinasCompletadasFragment.java).
- Listviewelement rutinas.xml → Plantilla de un elemento de la listview. Será el título, descripción y foto de una rutina.
- Nav\_header\_main.xml → Plantilla de la parte superior del menú. Donde aparece una foto, el nombre de usuario y el correo.
- Carpeta menú →
  - Activity main\_drawer.xml → Los elementos que tendrá el menú. Cada uno de ellos hará referencia a un nav.
- Carpeta mipmap → Aquí guardaremos las imágenes que se usen en la aplicación, mayormente todas las de los ejercicios y las rutinas.
- Carpeta navigation:
  - Mobilenavigation.xml → En este archivo se almacenan todas las navegaciones entre los fragment.
- Carpeta values:
  - Arrays.xml → Se guardarán los string-array de las preferencias. X ejemplo, podremos editar en las preferencias el idioma. Pues aquí guardaremos que valores podrá coger.
  - Carpeta strings → En esta carpeta se guardarán todas las traducciones que se usarán en la aplicación. Se creará un fichero String.xml por idioma. Tendremos 4, el default, castellano, euskera, ingles.
  - Carpeta themes → En themes.xml guardaremos los diferentes temas creados. He creado cuatro temas que tienen de base los colores: morado, naranja, verde y azul.
- Carpeta xml → Aquí se guardarán las plantillas de las preferencias o ajustes.
  - Fragment ajustes.xml → La plantilla de ajustes generales.
  - Fragment ajustes\_usuario.xml → La plantilla de los ajustes de usuario.

### 3- Manual de usuario

Aquí se realizará una breve descripción con imágenes de lo que el usuario puede hacer en cada una de las ventanas disponibles en la aplicación:

#### 3.1- INICIAR SESIÓN

El usuario deberá insertar su nombre de usuario y contraseña y hacer clic en 'Iniciar sesión' para poder acceder a su cuenta. Si los datos introducidos son correctos, iniciaremos sesión y nos llevará a la pestaña principal de la aplicación. Si, al contrario, los datos introducidos no son correctos, saltará un mensaje de error, avisando al usuario de que los datos introducidos no son correctos.

También tendremos la opción de crear una nueva cuenta si aún no disponemos de ninguna, haciendo clic en 'Crear una cuenta', que nos llevará a la ventana de creación de cuenta.



Imagen 4: inicio de sesión

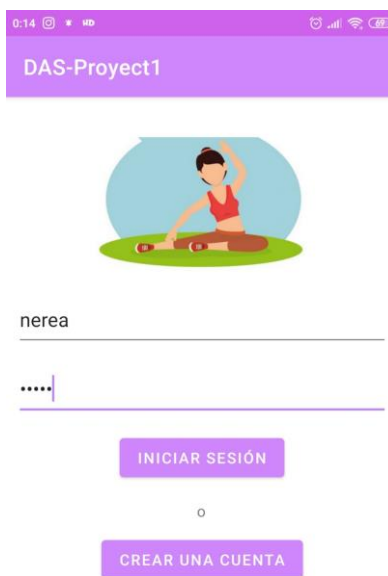


Imagen 5: inicio de sesión

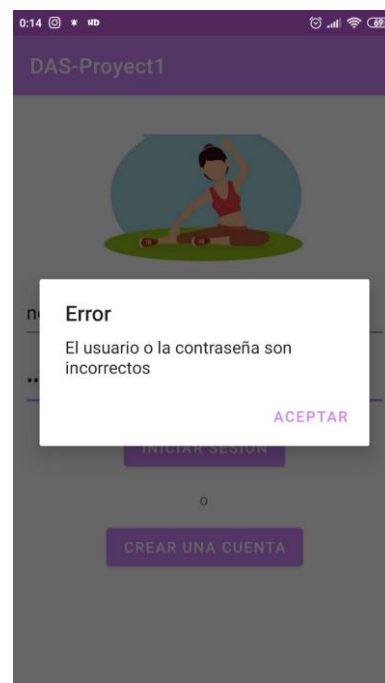


Imagen 6: Error en inicio de sesión



Imagen 7: Inicio de sesión horizontal



### 3.2- CREAR USUARIO

Esta es la ventana para crear una nueva cuenta. El usuario deberá indicar un nombre de usuario, un email, y una contraseña, la cual se deberá indicar dos veces. Una vez indicados los datos, se deberá dar al botón de 'Crear'. Si las contraseñas no coinciden se mostrará un mensaje de error indicando que las contraseñas no coinciden. Por otro lado, si el usuario no se ha podido crear correctamente, mayormente porque ese nombre de usuario ya estará ocupado por otro usuario, se mostrará otro mensaje indicando que no se ha podido crear el usuario. Si la cuenta se crea correctamente, se iniciará sesión con esa nueva cuenta y se dirigirá a la ventana principal de la aplicación.



DAS-Project1

Usuario

Email

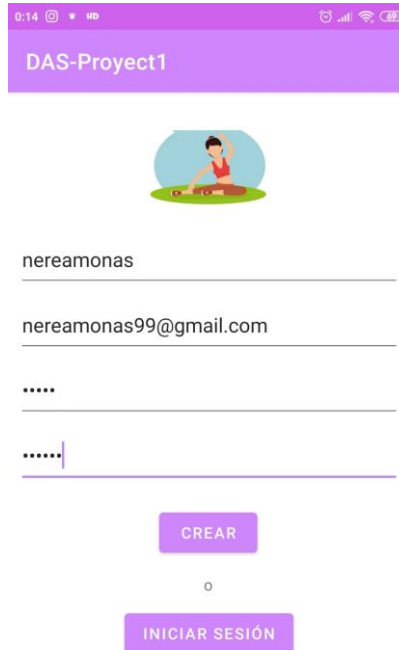
Contraseña

Repite la contraseña

CREAR

INICIAR SESIÓN

Imagen 8: Crear usuario



DAS-Project1

nereamonas

nereamonas99@gmail.com

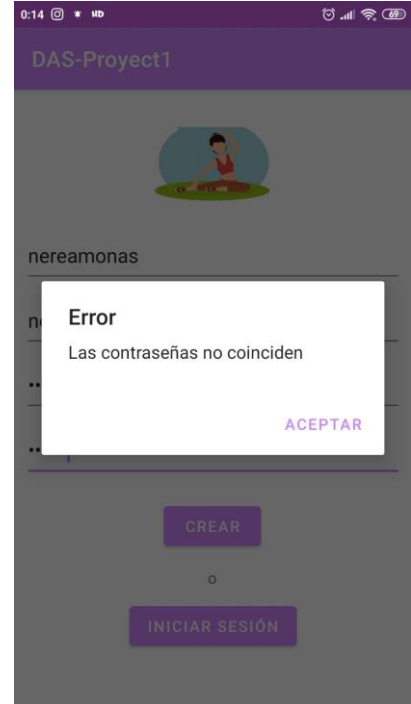
.....

.....

CREAR

INICIAR SESIÓN

Imagen 9: Crear usuario



DAS-Project1

nereamonas

Error

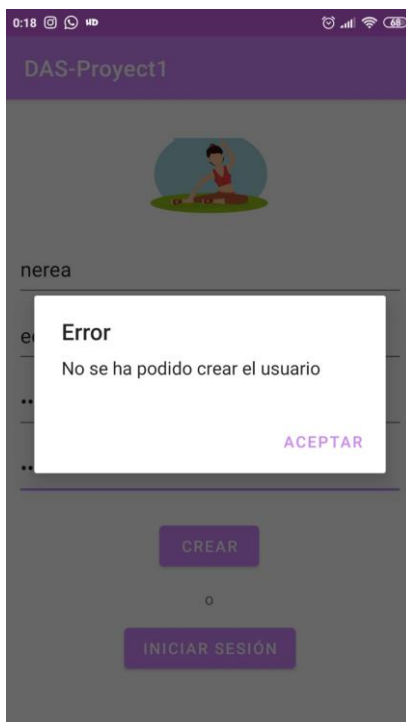
Las contraseñas no coinciden

ACEPTAR

CREAR

INICIAR SESIÓN

Imagen 10: Error las contraseñas no coinciden



DAS-Project1

nerea

Error

No se ha podido crear el usuario

ACEPTAR

CREAR

INICIAR SESIÓN

Imagen 11: Error al crear usuario



DAS-Project1

Usuario

Email

Contraseña

Repite la contraseña

CREAR

INICIAR SESIÓN

Imagen 12: Crear usuario horizontal

### 3.3- PAGINA PRINCIPAL AL INICIAR SESION- CALENDARIO

Una vez iniciemos sesión esta es la pestaña que veremos. Se mostrará un calendario y en la parte inferior se listarán todas las rutinas que se han completado ese día. Si en el calendario seleccionamos otro día, esta lista se actualizará mostrando las rutinas completadas en el nuevo día seleccionado.

En la parte superior izquierda, tenemos el botón del menú. Al hacer clic sobre ese botón, se desplegará el menú. Ese menú, estará accesible en la gran mayoría de las ventanas que veremos a continuación. Las cuales podemos acceder haciendo clic desde este menú.



Imagen 13: Calendario

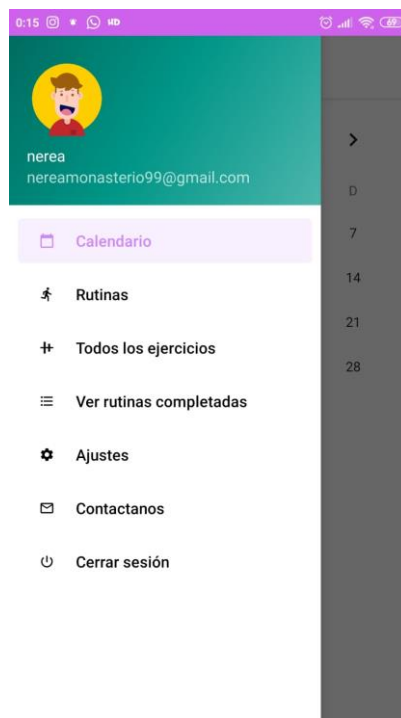


Imagen 14: Menú



Imagen 15: Calendario

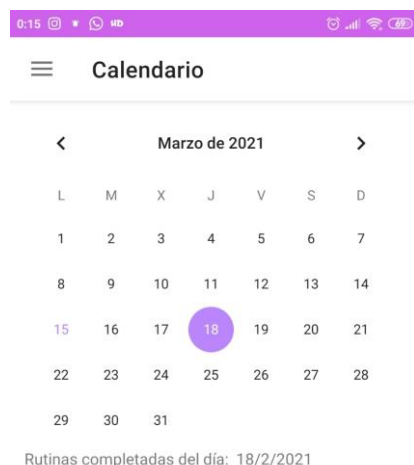


Imagen 16: Calendario



Imagen 17: Calendario horizontal

### 3.4- RUTINAS

Si elegimos la ventana de rutinas, se nos mostrará una lista con todas las rutinas disponibles. Cada rutina contará con x ejercicios.

Si hacemos clic sobre una rutina, nos dirigirá a la ventana donde se mostrarán los ejercicios que tiene la rutina seleccionada. Esta ventana la explicaremos a continuación más detalladamente.

Si hacemos una pulsación larga sobre una rutina, nos mostrará un mensaje indicando si queremos borrar dicha rutina. En el caso de decir que no, la rutina no se eliminará, pero en el caso de decir que si, la rutina se eliminará de la base de datos y de la lista.

En la parte inferior derecha contamos con un botón con el símbolo +. Con este botón podremos crear una nueva rutina personalizada. Primero nos indicará que le asignemos un nombre a la rutina. Si le damos al 'ok', nos abrirá otra alerta, donde se listarán todos los ejercicios que hay disponibles para que agregues a tu rutina personalizada, podrás seleccionar tantos ejercicios como desees. Una vez elegidos todos los ejercicios le volvemos a dar al 'ok' y ya se habrá creado nuestra rutina. Si accedemos a ella haciendo clic, veremos que cuenta con los ejercicios que le acabamos de añadir.

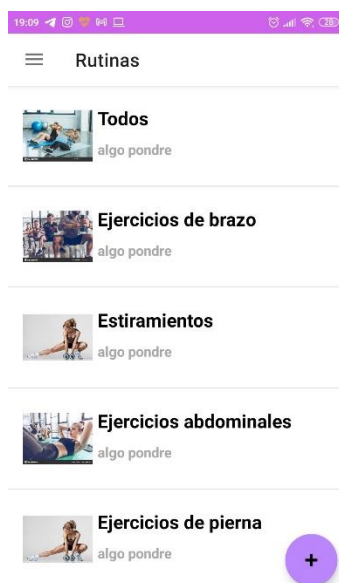


Imagen 18: Rutinas

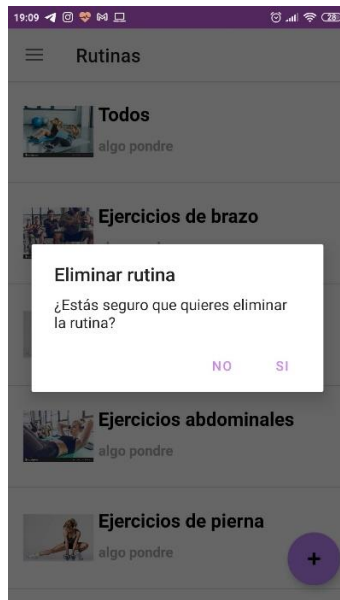


Imagen 19: Eliminar rutina



Imagen 20: Rutinas horizontal

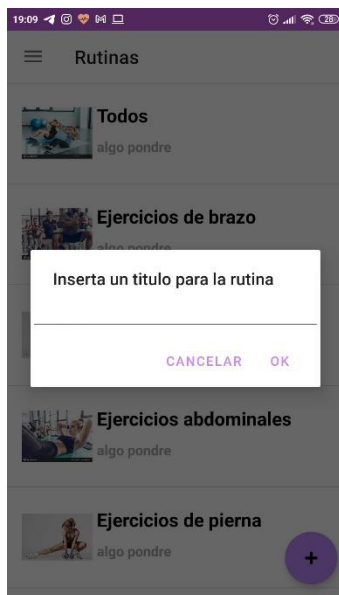


Imagen 21: Insertar título rutina nueva

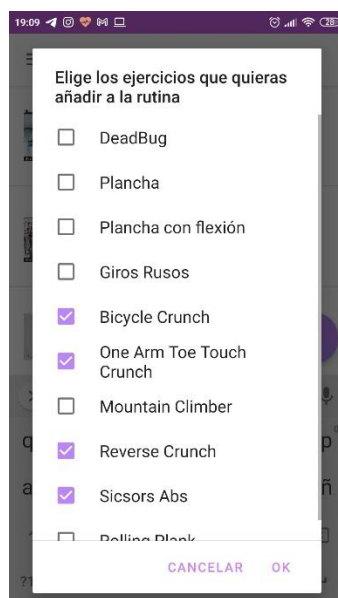


Imagen 21: Elegir ejercicios a añadir

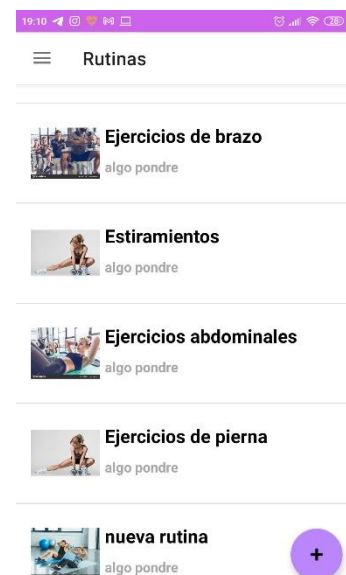


Imagen 22: Nueva rutina añadida

### 3.5- RUT EJERCICIO

Como hemos dicho cada rutina cuenta con x ejercicios. Al hacer clic sobre una rutina esta es la pestaña que se mostrará. Primero se mostrará el primer ejercicio de la rutina, indicando su título, una breve descripción, una foto del ejercicio a realizar, un numero indicando en que ejercicio estamos y cuantos ejercicios tiene la rutina, un temporizador, que indicará el tiempo que hay que estar haciendo dicho ejercicio, un botón de volver atrás (al ejercicio anterior o en caso de ser el primero a la ventana de rutinas), un botón de pause, para pausar el contador y un botón de siguiente, para pasar al siguiente ejercicio. Si no tocamos nada, cuando el contador termina, automáticamente saltará al siguiente elemento, y nos mostrará un Toast (En caso de que estén habilitadas este tipo de notificaciones en los ajustes de la aplicación. Por defecto están desactivadas). Se repetirá este proceso sucesivamente hasta completar todos los ejercicios de la rutina. Cuando se completan todos, nos saltará una notificación (En caso de tener las notificaciones activadas en la pestaña ajustes, por defecto están desactivados), nos guardará en un fichero en la memoria externa que hemos completado esa rutina en tal fecha y tal hora y nos dirigirá a la pestaña de rutinas saltando un Toast (en caso de estar activados) indicando que se ha completado la rutina.

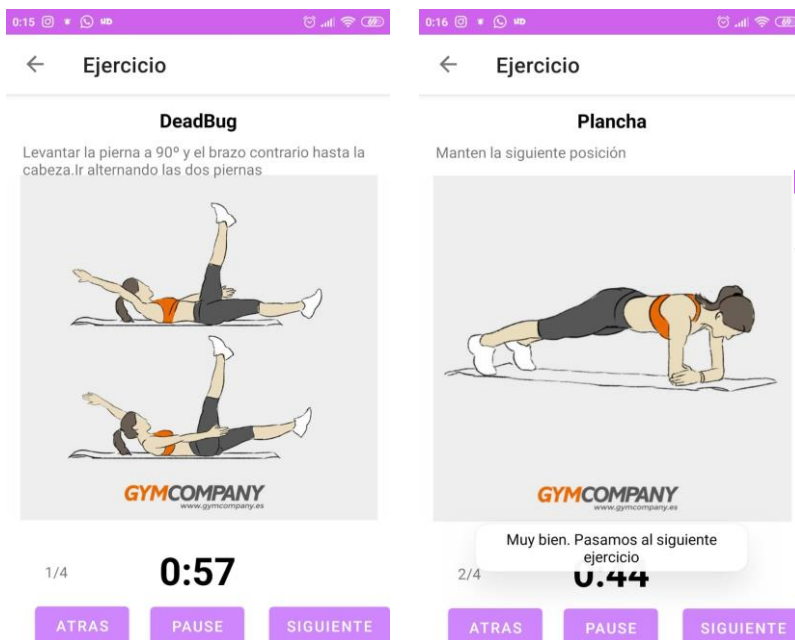


Imagen 23: Ejercicio

Imagen 24: Saltar ejercicio



Imagen 25: Ejercicio horizontal

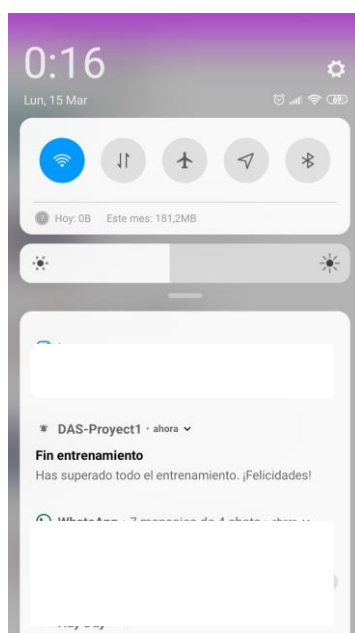


Imagen 26: Notificación local

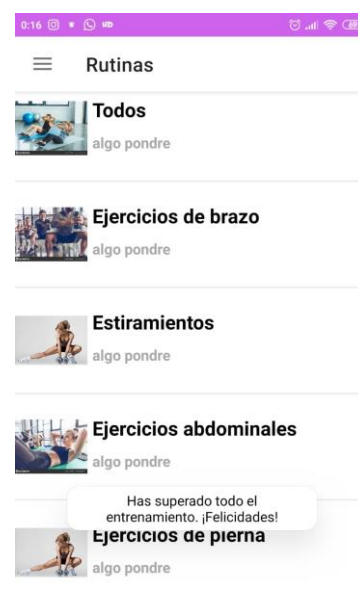


Imagen 27: Superada la rutina

### 3.6- EJERCICIOS

El siguiente apartado del menú es 'Todos los ejercicios'. Aquí se listarán todos los ejercicios disponibles.



Imagen 28: Todos los ejercicios

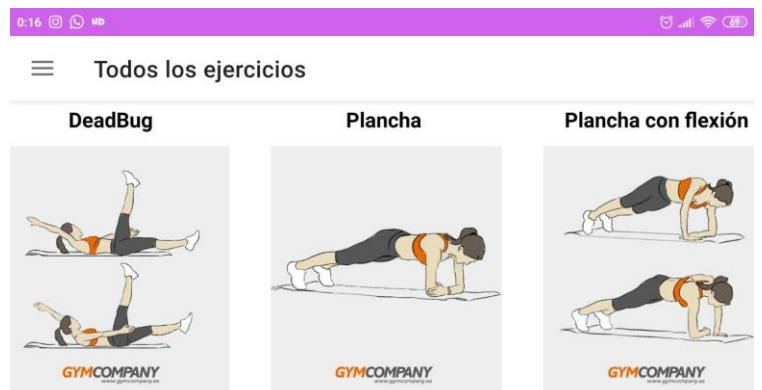


Imagen 29: Todos los ejercicios horizontal

### 3.7- RUTINAS COMPLETADAS

Como en la ventana de 'Calendario', se listarán todas las rutinas completadas por ese usuario, pero en este caso todas, no solo la del día seleccionado.

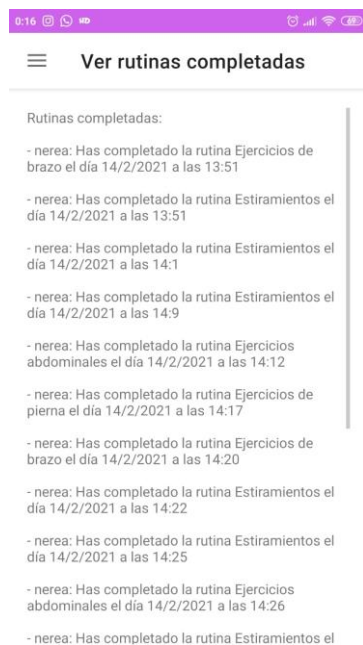


Imagen 29: Rutinas completadas



Imagen 30: Rutinas completadas horizontal

### 3.8- AJUSTES

Aquí podremos configurar diferentes aspectos de la aplicación. Esto será lo que veremos al abrirlo:

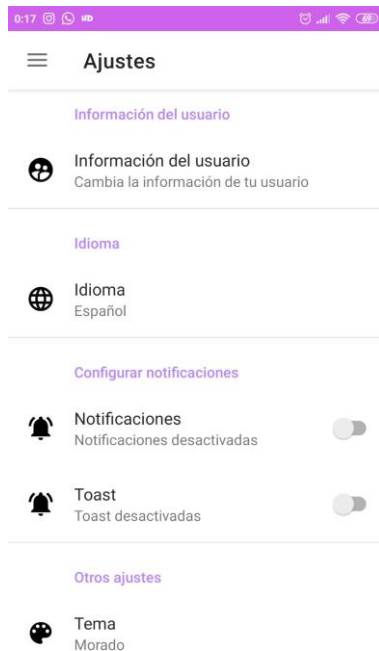


Imagen 31: Ajustes

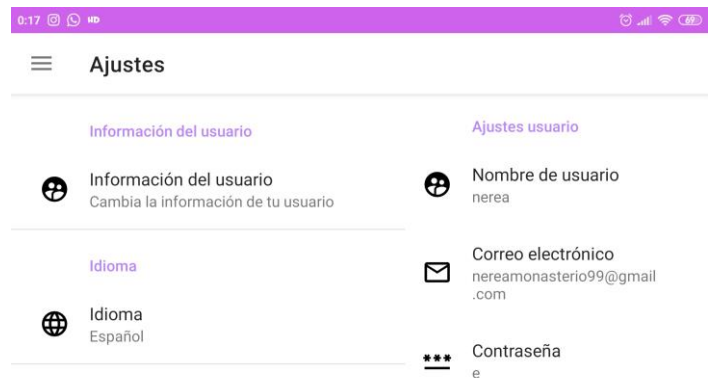


Imagen 32: Ajustes horizontal

Empezamos por la información de usuario, si nos encontramos en la versión vertical y pulsamos sobre información de usuario, nos abrirá otra ventana para modificar la información de nuestro usuario, el nombre de usuario, el email y la contraseña. Si estamos en la opción horizontal, no hace falta clicar sobre información de usuario ya que nos aparece directamente a la derecha del móvil.

Haciendo clic sobre el nombre de usuario, podremos modificar su valor. En el caso de que no se haya podido modificar (seguramente porque ya exista un usuario con ese nombre) saltará un toast indicando que no se ha podido actualizar, y en el caso de que se haya modificado correctamente saltará otro toast de que se ha modificado correctamente. Esto aplicará para los otros dos casos, el cambio de email y el cambio de contraseña.

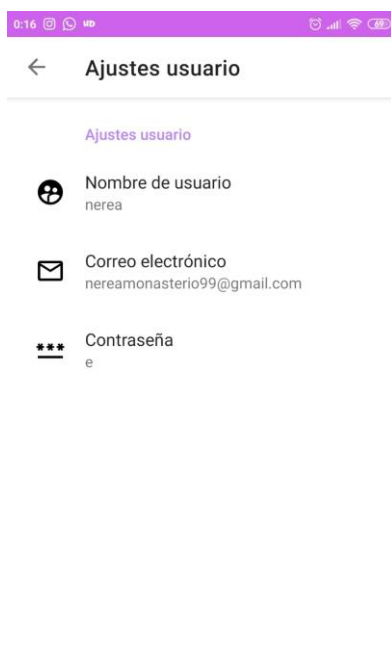


Imagen 33: Ajustes usuario



Imagen 34: Cambiar nombre usuario

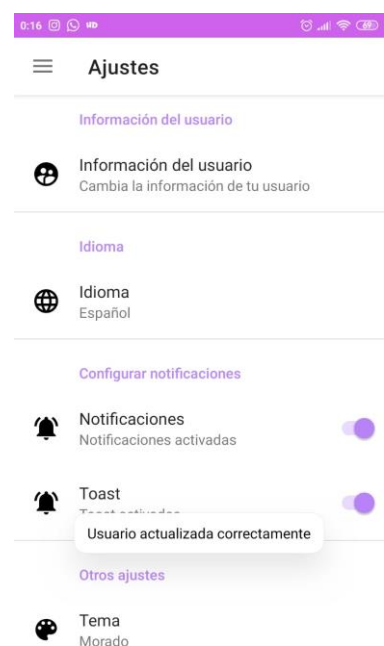


Imagen 35: Usuario actualizado



La siguiente opción por modificar es el idioma. La aplicación está disponible en Castellano, Euskera e Inglés. Al cambiar el idioma, toda la aplicación cambiará el idioma, e incluso si la cerramos y la volvemos a abrir se mantendrá la configuración del idioma. (El nombre de las rutinas y de los ejercicios se cogen de la base de datos, por lo que no se modificarán con el cambio de idioma).

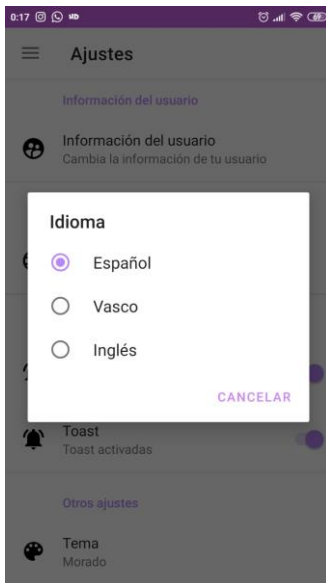


Imagen 36: Cambiar idioma

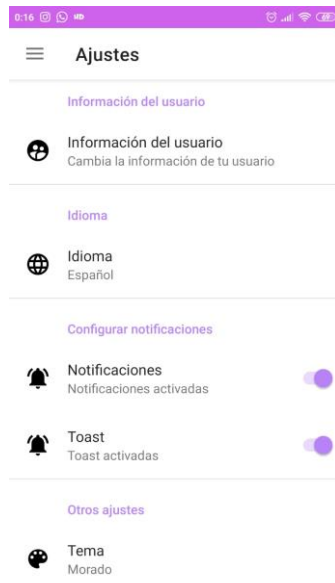


Imagen 37: Español

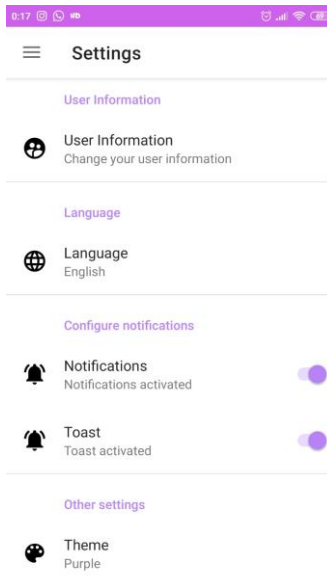


Imagen 38: Inglés

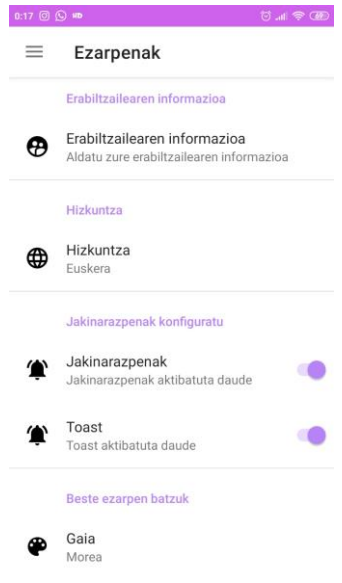


Imagen 39: Euskera

Ahora activaremos o desactivaremos dos tipos de notificaciones. La primera, las notificaciones del teléfono y por otro lado los toast.

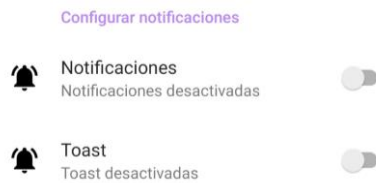


Imagen 40: Notificaciones desactivadas

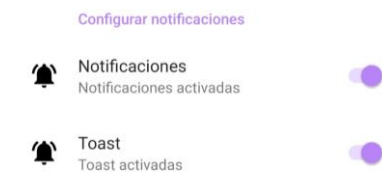


Imagen 41: Notificaciones activadas

Y por último, se podrá cambiar el tema de la aplicación, es decir, el color. Estará disponible en cuatro colores diferentes: morado, naranja, verde y azul. Esta configuración se mantendrá aun si cerramos la aplicación y la volvemos a abrir.

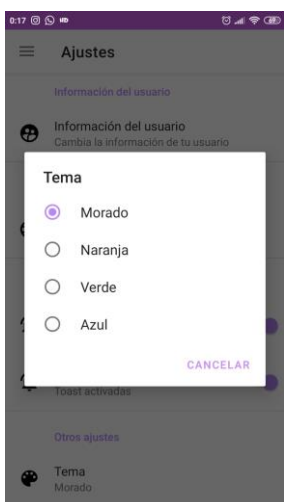


Imagen 42: Cambio tema

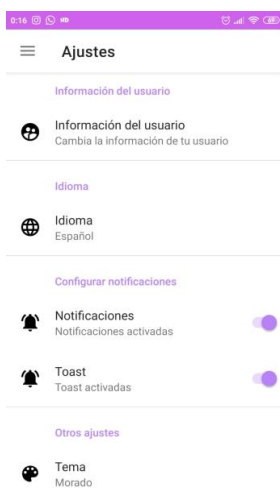


Imagen 43: Tema morado



Imagen 44: Tema verde

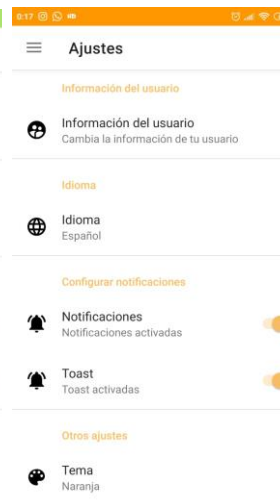


Imagen 45: Tema naranja



Imagen 46: Tema azul

### 3.10- CONTACTANOS

La siguiente ventana del menú es contáctanos. Se abrirá una pestañita para elegir con que aplicación queremos mandar el correo (Gmail, Outlook...) y al abrir la aplicación, nos mostrará automáticamente el email destino, un asunto y un mensaje. Así el usuario podrá ponerse en contacto con nosotros para cualquier duda o valoración de la aplicación.

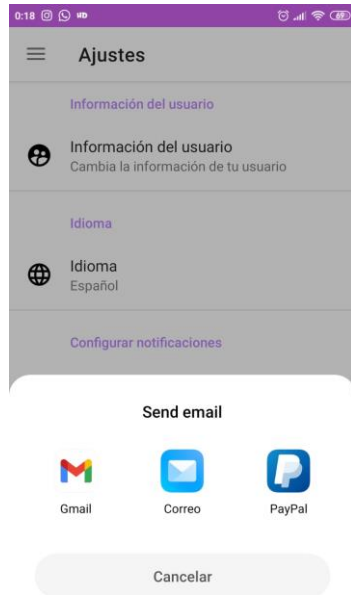


Imagen 47: Contáctanos

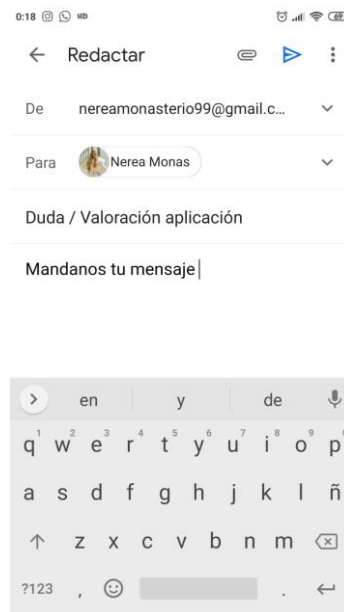


Imagen 48: Abrir aplicación correo

### 3.11- LOGOUT

Para concluir, tenemos el botón de cerrar sesión. Donde al hacer clic nos mostrará una alerta, para asegurarnos de si realmente quiere o no cerrar sesión. En el caso de decir que no, se cerrará la alerta y se quedará en la pestaña en la que estaba. Di al contrario hace clic en cerrar sesión, la sesión que está actualmente iniciada se cerrará y nos dirigirá a la ventana de inicio de sesión.

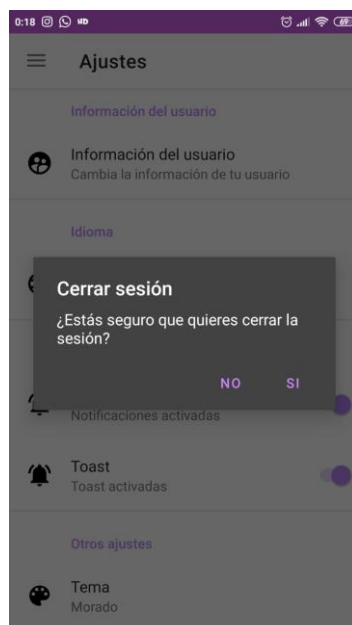


Imagen 49: Cerrar sesión



## 4- Explicación de cómo se cumple con cada uno de los requisitos

4.1- Uso de Listviews personalizados o de RecyclerView+CardView para mostrar listados de elementos con diferentes características.

Se hará uso de ListView en la ventana de rutinas y una RecyclerView+CardView en la ventana donde se muestran todos los ejercicios:

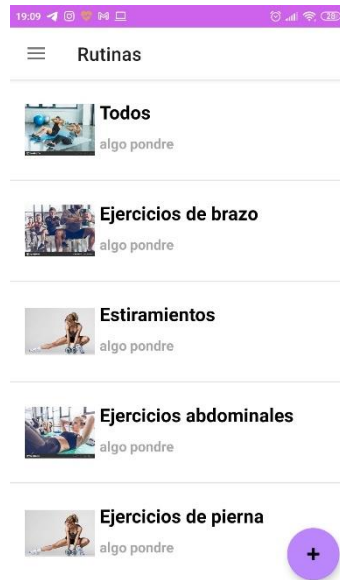


Imagen 50: ListView

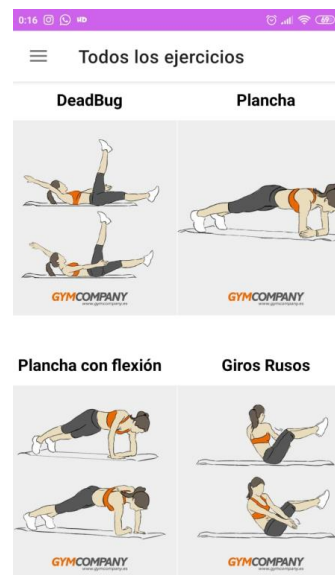


Imagen 51: RecyclerView+CardView

4.2-Usar una base de datos local, para listar, añadir y modificar elementos y características de cada elemento.

Se creará una base de datos con 5 tablas:

- 1- usuario → En esta tabla se guardarán los datos del usuario, su nombre de usuario, mail y contraseña.
  - user TEXT NOT NULL UNIQUE
  - email TEXT
  - pass TEXT
- 2- rutina → Se guardará la información de la rutina. Un id, el nombre y la referencia a la foto
  - id INTEGER PRIMARY KEY AUTOINCREMENT
  - nombre TEXT
  - foto TEXT
- 3- ejercicio → Se guardará la información de un ejercicio. Un id, el nombre, una breve descripción, la referencia a la foto y la duración (el tiempo que hay que estar realizando el ejercicio).
  - id INTEGER PRIMARY KEY AUTOINCREMENT
  - nombre TEXT
  - descripcion TEXT
  - foto TEXT
  - duracion TEXT
- 4- rutEjer → Una rutina, puede tener muchos ejercicios, y un ejercicio puede estar en diferentes rutinas. Como es una relación NM, crearemos una clase aparte para unir estos elementos. Guardaremos el id de la rutina y el id del ejercicio
  - idRut INTEGER
  - idEjer INTEGER

- FOREIGN KEY(idRut) REFERENCES rutina(id)
  - FOREIGN KEY(idEjer) REFERENCES ejercicio(id)
- 5- userRut→ Un usuario, puede tener muchas rutinas, y una misma rutina, la pueden tener asociada diferentes usuarios. Como es una relación NM, crearemos una clase aparte para unir estos elementos. Guardaremos el nombre de usuario y el id de la rutina.
- idUser TEXT
  - idRut INTEGER
  - FOREIGN KEY(idUser) REFERENCES usuario(user)
  - FOREIGN KEY(idRut) REFERENCES rutina(id) )

Haciendo uso de estas tablas, se realizarán distintas acciones desde la aplicación. Estas son las más importantes:

- Se listarán todas las rutinas disponibles del usuario en una ListView.
- Se listarán todos los ejercicios disponibles para ese usuario en una RecyclerView+CardView.
- Se listarán todos los ejercicios que forman parte de una rutina.
- Se podrán eliminar rutinas.
- Se podrán crear nuevas rutinas y asociar ejercicios a la rutina recién creada.
- Se podrá crear un usuario.
- Se realizarán comprobaciones para saber si el usuario insertado es el correcto.
- Se podrá modificar el nombre de usuario.
- Se podrá modificar el email del usuario.
- Se podrá modificar la contraseña del usuario.
- Añadir una rutina
- Añadir un ejercicio
- Añadir un usuario
- Añadir un usuario-rutina
- Añadir un rutina-ejercicio

Nombre	Tipo	Esquema
Tablas (7)		
android_metadata		CREATE TABLE android_metadata (locale TEXT)
ejercicio		CREATE TABLE ejercicio( id INTEGER PRIMARY KEY AUTO
id	INTEGER	"id" INTEGER PRIMARY KEY AUTOINCREMENT
nombre	TEXT	"nombre" TEXT
descripcion	TEXT	"descripcion" TEXT
foto	TEXT	"foto" TEXT
duracion	TEXT	"duracion" TEXT
rutEjer		CREATE TABLE rutEjer( idRut INTEGER, idEjer INTEGER, F
idRut	INTEGER	"idRut" INTEGER
idEjer	INTEGER	"idEjer" INTEGER
rutina		CREATE TABLE rutina( id INTEGER PRIMARY KEY AUTOIN
id	INTEGER	"id" INTEGER PRIMARY KEY AUTOINCREMENT
nombre	TEXT	"nombre" TEXT
foto	TEXT	"foto" TEXT
sqlite_sequence		CREATE TABLE sqlite_sequence(name,seq)
userRut		CREATE TABLE userRut( idUser TEXT, idRut INTEGER, FOI
idUser	TEXT	"idUser" TEXT
idRut	INTEGER	"idRut" INTEGER
usuario		CREATE TABLE usuario( user TEXT NOT NULL UNIQUE, er
user	TEXT	"user" TEXT NOT NULL UNIQUE
email	TEXT	"email" TEXT
pass	TEXT	"pass" TEXT

Imagen 52: Base de datos

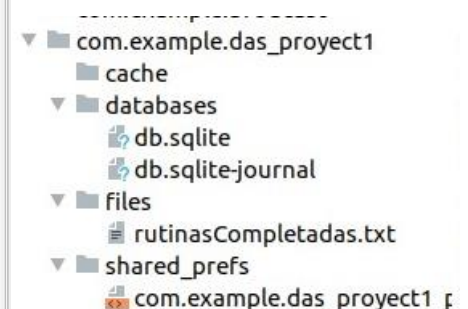


Imagen 53: Localización base de datos

#### 4.3- Uso de diálogos.

Se hará uso de diversos tipos de diálogos.

- Diálogos informativos → Los cuales solo muestran un mensaje al usuario
  - o Para indicar que los datos ingresados para iniciar sesión no son correctos
  - o Para indicar que al crear un usuario las contraseñas no coinciden
  - o Para indicar que, al crear un usuario, dicho usuario no ha podido ser creado

**Error**  
El usuario o la contraseña son incorrectos

ACEPTAR

Imagen 54: Dialogo info 1

**Error**  
Las contraseñas no coinciden

ACEPTAR

Imagen 55: Dialogo info 2

**Error**  
No se ha podido crear el usuario

ACEPTAR

- Diálogos informativos con botones → Muestran un mensaje al usuario y dependiendo del botón elegido, se realizará una acción u otra
  - o Para cerrar sesión, el usuario podrá decir si 'Sí' o 'No' y dependiendo de la respuesta se cerrará o no la sesión.
  - o Para eliminar una rutina, el usuario podrá finalmente decidir entre 'Sí' o 'No' y dependiendo de la respuesta se borrará o no.

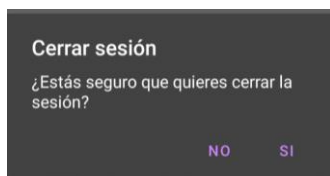


Imagen 57: Dialogo botones 1

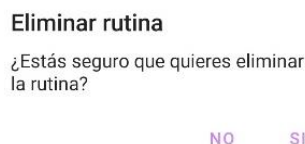


Imagen 58: Dialogo botones 2

- Diálogos con un EditText → Muestra un dialogo donde el usuario deberá insertar un texto.
  - o Para crear una nueva rutina, se deberá indicar el nombre que se le quiere asignar a dicha rutina. Por esta razón se ha añadido un EditText.
  - o Para modificar el usuario, email o contraseña.

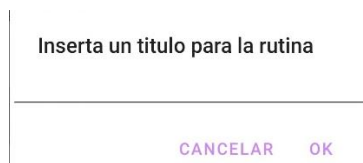


Imagen 59: Dialogo editText 1

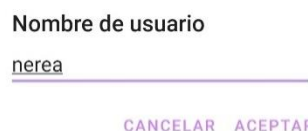
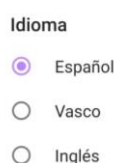


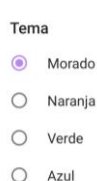
Imagen 60: Dialogo editText 2

- Diálogos con radio button → Se le mostrará al usuario diferentes opciones y podrá seleccionar una de ellas.
  - o Para cambiar el idioma de la aplicación
  - o Para cambiar el tema(colores) de la aplicación.



CANCELAR

Imagen 61: Dialogo radioButton 1



CANCELAR

Imagen 62: Dialogo radioButton 2

- Diálogos con check box → Se le mostrará al usuario una lista con diferentes elementos y podrá seleccionar tantos elementos como desee.
  - o Al crear una nueva rutina, el usuario podrá elegir que ejercicios puede meter en dicha rutina.

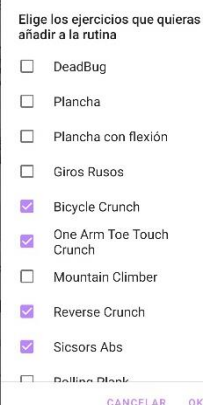


Imagen 63: Dialogo checkBox

#### 4.4- Usar notificaciones locales.

Cuando el usuario completa una rutina, saltará una notificación local indicando que la rutina ha sido completada con éxito. Para que dicha notificación llegue, deberá estar activada la opción de recibir notificaciones locales en el menú de ajustes.

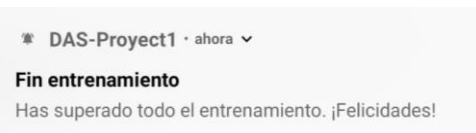


Imagen 64: Notificación local

4.5- Controlar que la aplicación se comporte correctamente y no se pierda información, aunque en medio de la ejecución recibamos una llamada telefónica, giremos el móvil, etc. No se puede impedir que la aplicación se gire. Controlar la pila de actividades para evitar incoherencias.

La aplicación se comportará correctamente en todos los casos y al girar la pantalla, se guardará la información. Aquí algunos ejemplos explicando cómo se ha evitado la pérdida de datos.

- En la ventana de ajustes, se puede cambiar tanto el idioma como el tema. Pero cuando se rotaba la pantalla, o se cerraba la aplicación y se volvía a abrir esta información se perdía, y volvía a la opción por defecto. Para evitar esto se han creado dos clases, una que hereda de Fragment y otra que hereda de AppCompatActivity. En estas clases, modificaremos el método onCreate, cogiendo el valor del idioma y del tema indicado en los ajustes y aplicandoselos al fragment/activity actual. Una vez modificado el método onCreate, todas las clases de nuestra aplicación en vez de heredar directamente de Fragment u AppCompatActivity, heredarán de la clase intermedia que hemos creado modificando el método onCreate. De esta manera, al crearse cada uno de los Fragmentos/Actividades, se creará con el valor guardado en los ajustes.
- Otro claro ejemplo es la ventana donde se muestra el ejercicio a realizar con el contador. Deberemos implementar el método onSaveInstanceState, para poder guardar el estado del ejercicio que estamos realizando. Su posición, es decir, tenemos una lista de ejercicios que forman parte de la rutina, pues cual es el ejercicio que se está mostrando actualmente en pantalla. Y por otro lado el tiempo del contador. De esta manera, cada vez que giremos la pantalla se invocará automáticamente a este método y se guardarán estos dos valores. Los cuales, al cargar de nuevo el fragmento al girarse, los cogerá y actualizará la página en base a los valores recogidos.

Por otro lado, cabe destacar, que se ha pasado información entre fragmentos. Para ello, al llamar a un nuevo intent, se añadirá lo siguiente: `intent.putExtra("usuario", user);` Así le estamos pasando al intent el extra usuario. Se podrá coger desde el nuevo intent con: `this.usuario = getIntent().getExtras().getString("usuario");`

En el caso de tener que pasar información entre fragmentos se realizará haciendo uso de un Bundle y Navigation:

```
Bundle bundle = new Bundle();
bundle.putString("usuario", this.usuario);
Navigation.findNavController(this,R.id.nav_host_fragment).navigate(R.id.action_global_ajustes
GeneralesFragment, bundle,options);
```

Y desde el nuevo fragmento, se recuperará esa información con: `getArguments().getString("usuario");`

También se ha controlado la pila de actividades para evitar incoherencias. La página de inicio de sesión y crear una cuenta, al saltar a otro Intent, se eliminarán de la pila. Así una vez que entremos en el Intent principal solo tendremos ese. Dentro del Intent principal, navegaremos con fragmentos que usarán un `singleTop`, es decir, si el último elemento de la pila es el fragmento que acabamos de volver a abrir, no cargará otro encima, sino que recuperará el actual. De todas maneras, hay una excepción en el caso del Fragmento ejercicios, donde se muestra el ejercicio que hay q realizar. Cuando ese fragmento finaliza, no se podrá volver a él dándole al botón de atrás. Por otro lado, cuando cerramos sesión y nos dirige a la página de inicio de sesión, el Intent principal se eliminará de la pila. Si no fuera así, al darle atrás, estaríamos dentro del usuario registrados, lo cual no tendría sentido.

#### 4.6- Documentar la aplicación mediante comentarios en el propio código.

Todo el código de la aplicación incluye comentarios para hacerlo más entendible.

#### 4.7- Permitir que una misma funcionalidad se comporte de manera distinta dependiendo de la orientación (o del tamaño) del dispositivo mediante el uso de Fragments (0,5 puntos).

La mayoría de las ventanas cuentan con dos plantillas. Una para la versión vertical y otra para la versión horizontal, donde en algunas de ellas, los elementos que contienen cambiarán de lugar.

Pero dos de ellas se comportarán de manera diferente.

- 1- Cuando mostramos todos los ejercicios, en la versión vertical, se mostrará la RecyclerView+CardView en dos columnas, al contrario, si ponemos la orientación en modo horizontal, veremos tres columnas. Para ello, en el código se deberá ver la orientación actual del móvil para funcionar de una u otra manera

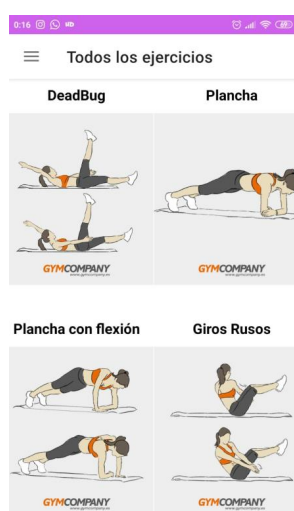


Imagen 65: Modo vertical

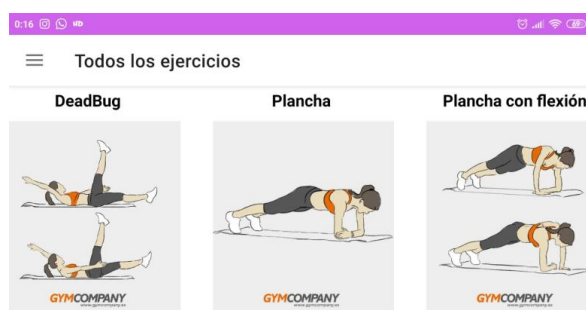


Imagen 66: Modo horizontal

- 2- La ventana de ajustes también se comportará de diferente manera dependiendo de la orientación del móvil. Por un lado, en el modo horizontal, se mostrarán los ajustes principales, y haciendo clic en información del usuario, nos abrirá otra ventana para editar los datos del usuario. En cambio, en la versión horizontal, la pantalla está dividida en dos y se muestran directamente los dos fragmentos, por lo que no habrá que hacer clic en información del usuario puesto que se visualizará directamente en pantalla.

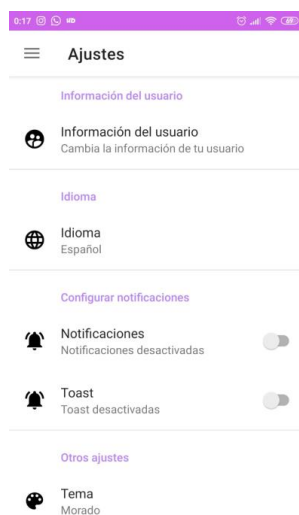


Imagen 67: Modo vertical A

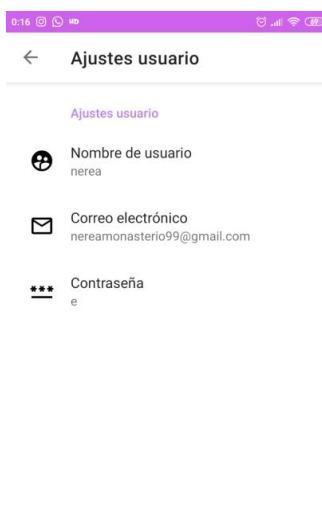


Imagen 68: Modo vertical B

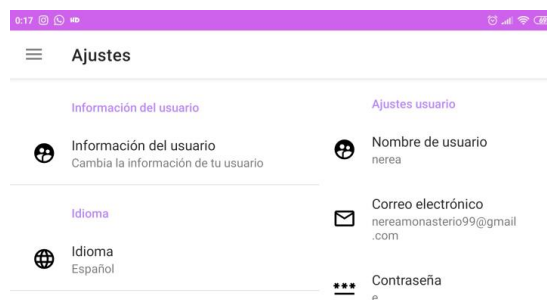


Imagen 69: Modo horizontal A+B

#### 4.8- Hacer la aplicación multiidioma y añadir la opción de cambiar de idioma en la propia aplicación (0,5puntos).

La aplicación está disponible en tres idiomas: Castellano (por defecto), Euskera e Inglés. El idioma se puede cambiar desde la ventana de ajustes y aunque rotemos la aplicación o la cerremos y volvamos a abrirla, no se perderá el idioma guardado.

Cabe destacar, que está toda la aplicación disponible en estos tres idiomas excepto los nombres de los ejercicios y de las rutinas, puesto que esta información la coge de la base de datos y son esos nombres los que están guardados.

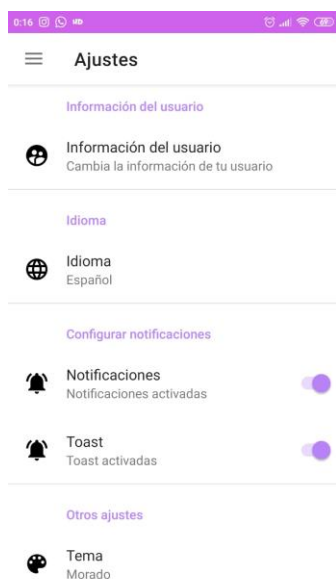


Imagen 70: Español

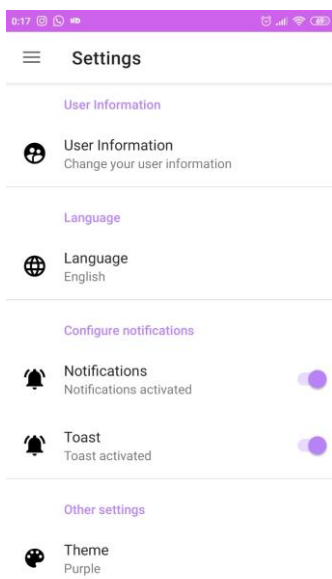


Imagen 71: Inglés

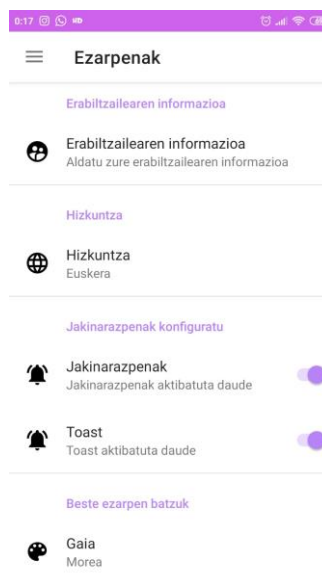


Imagen 72: Euskera

#### 4.9- Uso de ficheros de texto (1 punto).

Se hará uso un fichero externo en la memoria interna del móvil, para escribir y leer.

Por un lado, cada vez que el usuario finalice una rutina, se escribirá en el fichero el usuario que ha completado la rutina, el nombre de la rutina y la fecha y hora en la que se ha finalizado.

Y por otro lado, se leerá información del fichero en dos momentos. El primero, en el calendario, donde al hacer clic en un día, se mostrará en la parte inferior que rutinas se han completado dicho día por el usuario que tiene iniciada la sesión. Esa información se recogerá de este fichero. Y también, en el apartado de 'todas las rutinas completadas' donde se realizará la misma acción, pero cogiendo todas las rutinas completadas por el usuario, sin condiciones de fechas.

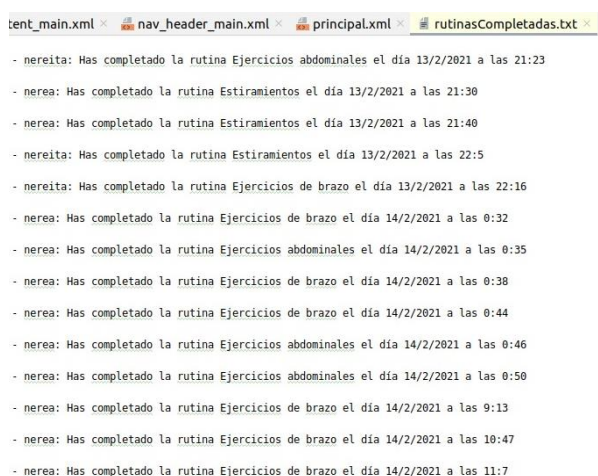


Imagen 73: Contenido fichero

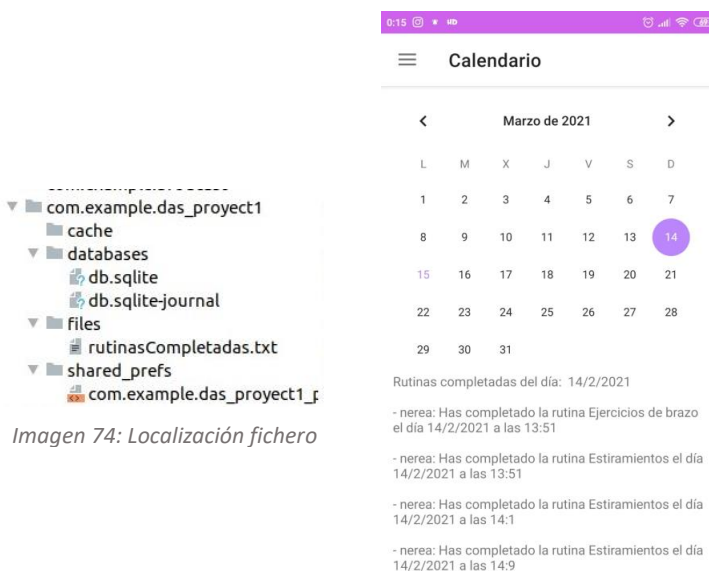


Imagen 75: Leer fichero app



4.10- Uso de Preferencias, para guardar las preferencias del usuario en cuanto a mostrar/esconder cierta información, elegir colores para la aplicación, o cualquier otra cosa relacionada con la visualización de la aplicación (1 punto).

Se ha creado una ventana de preferencias o ajustes, donde se permitirá al usuario realizar las siguientes acciones:

- Información del usuario → Se podrá ver la información del usuario y si se desea editar dicha información. Los valores que se pueden modificar son: el nombre de usuario, el email y la contraseña.
- Idioma → Se permite cambiar el idioma entre Castellano, Euskera y Ingles
- Notificaciones → Se puede configurar si se desean recibir notificaciones o si se quieren desactivar. Hay opción de activar o desactivar las notificaciones locales y los mensajes toast.
- Tema → Por último se permite cambiar el tema, a elegir entre 4 temas diferentes (morado, naranja, verde y azul).

Cabe destacar que la información almacenada en esta ventana se guardará y se aplicará en toda la aplicación, aun si se cierra y se vuelve a abrir desde 0.

Estos ajustes se guardarán en un fichero que se crea en el dispositivo:

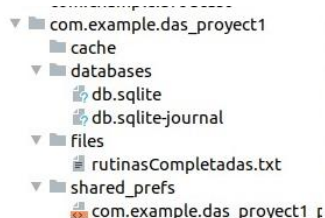


Imagen 77: Localización preferencias

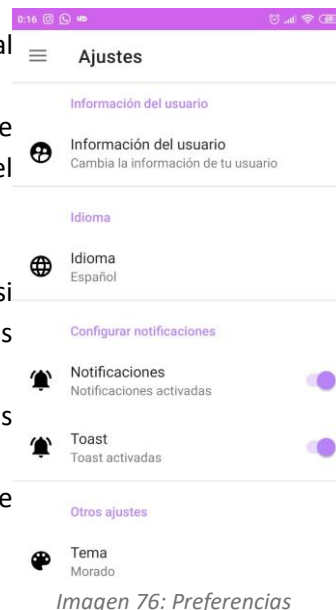


Imagen 76: Preferencias

4.11- Crear estilos y temas propios, para personalizar fondos, botones, etc. (0,5 puntos).

Como hemos dicho, se han creado tres nuevos temas, que cambian de colores diferentes elementos de la aplicación. Contamos con 4 temas, cada uno de ellos orientado a un color.

- Morado
- Naranja
- Verde
- Azul

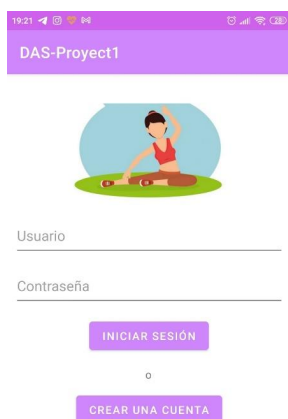


Imagen 78: Tema morado



Imagen 79: Tema naranja



Imagen 80: Tema verde



Imagen 81: Tema azul

#### 4.12- Usar intents implícitos para abrir otras aplicaciones, contactos, etc. (0,5 puntos).

Se ha usado un intent implícito para abrir la aplicación de correo. Si el usuario quiere contactar con los administradores de la aplicación, bien por una duda, para dar una valoración, o por algún fallo encontrado, podrá ponerse en contacto con nosotros haciendo clic en el menú 'Contáctanos'.

Cuando se abra la aplicación de correo elegida (Gmail, Outlook, correo...) saldrá por defecto un email destino, un asunto y un mensaje.

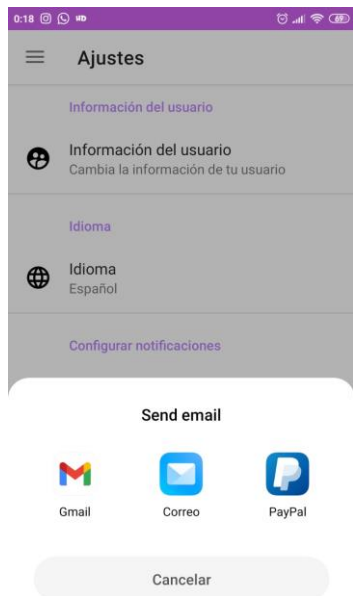


Imagen 82: Elegir aplicación con la que abrir



Imagen 83: Abrir aplicación

#### 4.13- Pantalla de login (y registro), para guardar credenciales de usuario (nombre de usuario – contraseña) en la base de datos local (1 punto).

Se ha creado una ventana de login y otra de registro que hacen uso de una base de datos local. Donde el usuario podrá registrarse o iniciar sesión para entrar a su propio apartado de la aplicación. En la base de datos, como hemos explicado anteriormente se almacenará el nombre de usuario, el email y la contraseña.



Imagen 84: Ventana login



Imagen 85: Ventana registro