

## Proyecto 2º Trimestre

Diseñar una aplicación similar a la que se muestra en los videos e imágenes compartidos.

La App puede ser de otra temática o tener otro diseño, pero las funcionalidades han de ser las mismas.

Además, tendrá que ser una App totalmente funcional y lista para descargar por los usuarios, por lo que hay que tener en cuenta el mínimo detalle para que no haya fallos y se consiga una buena experiencia de usuario.

### Login:

- **Login** de usuario y contraseña. Las credenciales correctas son éstas:
  - aficionado - 123456
  - jugador - qwerasdf
  - directivo - asdfasdf
- El campo de contraseña deberá mostrar el botón para poder visualizar el texto escrito.
- Cuando alguno de los dos campos está vacío, se muestra el error en rojo sobre los mismos **EditText**
- Si la contraseña o el usuario son incorrectos, se muestra el error con un **AlertDialog**
- **Acceso invitados.** Un botón que simplemente permite entrar al usuario en la aplicación.
- **Recordatorio de login:** Si el usuario es correcto, se almacena su nombre en **preferencias**. La próxima vez que se abra la App, se debe mostrar en el **EditText** el nombre de usuario con el que entró la última vez.
  - Si se entra como invitado, no hay que recordar nada
  - *Nota: Al hacer login correctamente, se abre la Actividad Principal. Ten en cuenta que desde la Actividad Principal no se podrá volver a login pulsando el botón atrás del teléfono.*

### MainActivity:

- Consta de un **menú lateral** con varias opciones
- Al entrar, se carga un **fragment** inicial.
- Mostrar una cabecera en el menú lateral. Como mínimo, esta cabecera debe mostrar el nombre o tipo de usuario que ha accedido
- Extra: investiga como mostrar la fecha actual en la cabecera del menú lateral
- Extra: Si pulsamos el botón atrás y estamos en el primer fragment de la pila, se muestra un cuadro de diálogo que pregunta si queremos salir. Si se pulsa salir, se cierra la App.

### Fragment Inicio:

- Se muestra un mensaje de bienvenida y una imagen.
- En el mensaje de bienvenida, aparece el nombre de usuario con el que se ha iniciado sesión.

- Si se ha iniciado sesión como invitado, el mensaje se oculta
- En la Toolbar se muestra un menú con una única opción para hacer logout. Al pulsar, se muestra un cuadro de diálogo que pregunta si quiere hacer logout. El logout debe abrir **loginactivity** y finalizar cualquier actividad de la App, de forma que loginactivity quede en el top de la pila de actividades.
- Extra: Huevo de pascua. Si se pulsa 10 veces el logo, se gana un premio. Se mostrará un cuadro de diálogo con una lista de premios. Al seleccionar uno, se genera un código aleatorio y se muestra en otro cuadro de diálogo, para que el usuario pueda recoger su premio.
- Extra: Huevo de pascua: Una vez que se ha ganado un premio, ya no volverá a funcionar el huevo de pascua, aunque se salga de la App.

### Equipo:

- La plantilla muestra un listado de data class con distinta información. Aunque los data class tendrá varios campos, en la lista solamente se mostrarán los más importantes. En la App de ejemplo, se muestra la imagen, el nombre y la posición.
  - Se debe crear una data class para el elemento del listado
  - La imagen puede ser de tipo Int y almacenar recursos drawable (R.drawable.imagenjugador) o ser de tipo String para guardar una url y cargar la imagen usando la librería Glide (la veremos en clase)
  - La data class tiene que ser de tipo **Parcelable (lo veremos en clase)** para poder pasar el objeto entre fragments como si se tratase de un tipo simple, como Int.
- La plantilla usará una lista en formato de rejilla.
- El listado cambiará el color de fondo de las celdas dependiendo de algún dato. En la App de ejemplo, el fondo de las letras cambia de color dependiendo de la posición del jugador.

### Vista en detalle:

- Al pulsar un elemento de la vista, se abre otro fragment para mostrar toda la información del ítem.
  - Se pasa toda la información usando Parcelable, como se ha indicado anteriormente.
  - El fragment se integra en la navegación, de forma que se puede volver atrás.

### Galería:

- La galería muestra una serie de imágenes paginadas haciendo uso de un **ViewPager (lo veremos también en clase)**
- Extra: Junto a la galería, mostrar un listado de imágenes pequeñas. Al pulsar sobre una de estas imágenes, nos lleva directamente a esa imagen en la galería principal.
- **\*\*Nota:** Es fácil realizarlo con un RecyclerView horizontal

### Contacto:

- El contacto está formado por una serie de botones de contacto y un formulario de contacto.
- El formulario de contacto manda un correo a [info@escuelaestech.es](mailto:info@escuelaestech.es) y [secretaria@escuelaestech.es](mailto:secretaria@escuelaestech.es).
  - El formulario no deja mandar nada a no ser que estén todos los campos rellenos.
  - Cuando está todo correcto, se muestra un cuadro de diálogo para preguntar si está seguro de mandarlo. Si es que sí, se abrirá la App de correo con todos los datos que el usuario ha escrito, para poder mandarlos a los correos mencionados.
  - Tener en cuenta el tipo de dato que se ha de escribir en cada campo para mostrar el teclado adecuado.
- Extra: en el campo donde el usuario escribe su correo electrónico, investiga qué código hay que usar para comprobar que el correo introducido tiene formato de correo electrónico. Si no tiene el formato adecuado, no podrá enviarse.
- Los botones de contacto son 2 + 2 extras
  - Botón de correo, al pulsar abre la app de correo con la dirección indicada.
  - Botón teléfono, que al pulsar llama al centro
- Extra: Incorpora un botón para abrir una conversación en WhatsApp con el número indicado.
- Extra: Incorpora un botón que abra la App de Google Maps con una ubicación específica

### Otras funcionalidades:

- Se pueden añadir otras funcionalidades que desees y/o consideres necesarias. Se contabilizará también como puntuación extra.
- Si se desarrollan funcionalidades no vistas en clase, el código tendrá que venir acompañado de comentarios explicativos de cómo funciona.

### Criterios de evaluación

Código bien ordenado y estructurado. Nombres apropiados de recursos, clases y variables.

Diseño: Uso de las vistas y layouts apropiados.

Funcionalidades del proyecto.

Para corregir la práctica es indispensable que el proyecto compile y se ejecute correctamente. En el momento que se rompa la app la corrección se dará por finalizada.

**Puntuación:** 70% parte obligatoria, 30% parte extra

- **Parte obligatoria: 70%**

- Login:	25%
- Activity y Navigation:	10%
- Fragment Inicio:	5%
- Fragment Galería:	10%
- Fragment Plantilla y Jugador:	30%
- Fragment Contacto:	20%

- **Parte Extra:**
  - **Fecha en la cabecera del menú** **5%**
  - **Mostrar dialog al salir** **15%**
  - **Pulsar imagen 10 veces para premio** **20%**
  - **No poder ganar más premios tras ganar uno** **5%**
  - **Mini lista de imágenes en la galería** **35%**
  - **Botones Mapa y Whatsapp en contacto:** **10%**
  - **Comprobar formato del correo en el formulario de contacto:** **10%**