

## Pla de Qualificació en Tecnologia Mòbil

**Nombre de la PLA:** Descarga y persistencia de datos

### Objetivos de la PLA:

- Aprender a descargar datos desde un servidor.
- Aprender a guardar los datos en una base de datos ORM.
- Aprender a comprobar si la aplicación tiene acceso a internet o no.
- Aprender a incluir librerías de terceros a una aplicación.

### Competencias vinculadas a la PLA:

Competencias técnicas	Soft Skills
<ul style="list-style-type: none"><li>• Programar en lenguaje Java</li><li>• Utilizar el entorno de desarrollo Android Studio y el Android SDK.</li></ul>	<ul style="list-style-type: none"><li>• Trabajo en equipo</li><li>• Resolución de problemas</li><li>• Búsqueda, gestión y uso de la información</li></ul>

### Horas de dedicación

Asíncronas	60
Totales	60

**Modalidad PLA:** Proyecto

### Instrucciones para realizar la PLA:

Durante esta tarea vamos a descargar de internet los libros que mostrará nuestra app MyBooks y utilizaremos la base de datos online Firebase para gestionar los datos .

### EJERCICIO 1. 10% Configurar Firebase

En primer lugar, con tu cuenta de Google, crea un nuevo proyecto en Firebase (<http://firebase.google.com>) y añade Firebase al proyecto creado en la PLA 1.

En el servidor de Firebase, importa el JSON adjunto a la práctica, el cual permitirá tener una lista de libros en la base de datos de nuestro servidor (Database → Import JSON).

## EJERCICIO 2. 20% Añadir Firebase al proyecto

El siguiente paso es añadir el cliente Firebase a la aplicación de Android, para poder leer datos del servidor.

Primero añadiremos las dependencias que permitirán hacer login a la base de datos y descargar los datos:

dependencies {

compile 'com.google.firebase:firebase-database:9.4.0'

compile 'com.google.firebase:firebase-auth:9.4.0'

}

Añade el permiso para conectarse a internet en el Manifest e inicializa las clases FirebaseAuth y FirebaseDatabase de Firebase en la actividad principal en el método “onCreate”:

**mAuth** = FirebaseAuth.getInstance();

**database** = FirebaseDatabase.getInstance();

Una vez está Firebase inicializado, lo primero sería hacer login al servidor Firebase. Para poder hacer login desde la aplicación a Firebase, primero habrá que añadir un usuario al servidor y autorizar el login mediante email y contraseña, en el apartado “Auth” de la consola Firebase:

The image displays two screenshots of the Firebase Authentication console interface.

**Top Screenshot: USERS Tab**

- Header: Authentication | WEB SETUP ?
- Sub-headers: USERS | SIGN-IN METHOD | EMAIL TEMPLATES
- Search bar: Search by exact email address (email@domain.com) or user UID. [ADD USER] [Refresh]
- Table:

Email	Providers	Created	Signed In	User UID
loran@uoc.edu	[Email icon]	6 Aug 2016	8 Aug 2016	BgvmcDlaNPD83Y199X3XFUn1f13

Displays up to 500 users ⓘ Rows per page: 50 1-1 of 1 < >

**Bottom Screenshot: SIGN-IN METHOD Tab**

- Header: Authentication | WEB SETUP ?
- Sub-headers: USERS | SIGN-IN METHOD | EMAIL TEMPLATES
- Section: Sign-in providers
- Table:

Provider	Status
[Email icon] Email/Password	Enabled
[Google icon] Google	Disabled
[Facebook icon] Facebook	Disabled
[Twitter icon] Twitter	Disabled
[GitHub icon] GitHub	Disabled
[User icon] Anonymous	Enabled

## Pla de Qualificació en Tecnologia Mòbil

El método “onCreate” de la aplicación principal, utiliza el método “signInWithEmailAndPassword” de FirebaseAuth para hacer login en el servidor y si funciona correctamente, pide la lista de libros introducida en el servidor con el método “addValueEventListener” que permitirá recibir la lista de libros cada vez que se produzcan cambios:

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
  
    setContentView(R.layout.activity_book_list);  
  
    mAuth = FirebaseAuth.getInstance();  
    database = FirebaseDatabase.getInstance();  
  
    // ===== INICIO CODIGO A COMPLETAR =====  
  
    // ===== FIN CODIGO A COMPLETAR =====  
}
```

### EJERCICIO 3. 20% Añadir una base de datos al proyecto

En este ejercicio, se añadirá una base de datos a nuestra aplicación con la ayuda de la librería SugarORM con el fin de tener la lista de libros del servidor disponible de forma local en caso de que no haya conexión a internet.

Añade la librería SugarORM/Realm (**elige una**), tal como se explica en el módulo 6 de los apuntes de la asignatura y cambia el modelo BookItem para que sea del tipo “SugarRecord”/”RealmObject”. En la clase BookContent.java añade dos métodos para simplificar las peticiones a la base de datos: uno para obtener todos los elementos de la base de datos y otro para comprobar si un libro ya existe en ella:

```
public static List<BookItem> getBooks(){  
    // ===== INICIO CODIGO A COMPLETAR =====  
    // ===== FIN CODIGO A COMPLETAR =====  
}  
  
public static boolean exists(BookItem bookItem) {
```

## Pla de Qualificació en Tecnologia Mòbil

```
// ===== INICIO CODIGO A COMPLETAR =====  
// ===== FIN CODIGO A COMPLETAR =====  
}
```

Cuando se reciba la lista de libros del servidor con FirebaseDatabase, comprueba que el libro recibido no exista y si es nuevo, añadirlo a esta. Los elementos se reciben en formato “DataSnapshot”, por lo tanto, primero habrá que convertirlos a una lista de libros con un “GenericTypeIndicator”:

```
GenericTypeIndicator<ArrayList<BookContent.BookItem>>
```

Y después se añadirán a la base de datos.

Una vez obtenida la lista de libros, crea un método en el adapter para actualizar la lista con los datos recibidos del servidor:

```
public class SimpleItemRecyclerViewAdapter  
    extends RecyclerView.Adapter<SimpleItemRecyclerViewAdapter.ViewHolder> {  
  
    private List<BookContent.BookItem> mValues;  
  
    public SimpleItemRecyclerViewAdapter(List<BookContent.BookItem> items) {  
        mValues = items;  
    }  
  
    public void setItems(List<BookContent.BookItem> items) {  
        // ===== INICIO CODIGO A COMPLETAR =====  
        // ===== FIN CODIGO A COMPLETAR =====  
    }  
    // ... resto del código  
}
```

En el caso de que se produzca un error al obtener la lista de libros del servidor, es decir, que llegue el resultado al método “onCancelled”, carga la lista de libros de la base de datos en el adapter. Ese caso indica que se ha producido un error y el usuario no tiene acceso a internet, o bien, el servidor no está funcionando correctamente.

### EJERCICIO 4. 15% Actualizar la lista de items

## Pla de Qualificació en Tecnologia Mòbil

En este ejercicio, añadiremos a la librería la opción de refrescar la lista de libros al estirar de la lista hacia abajo. Esta librería se encuentra actualmente dentro de la librería v4 de support de Android, por lo tanto si todavía no has añadido esta librería al proyecto, añádela al fichero gradle y, a los dos layouts de la lista de libros del proyecto añade la clase “SwipeRefreshLayout”.

En el método “onCreate” de la actividad principal del proyecto, se deberá obtener la referencia a la clase e implementar el método para actualizar los datos:

```
swipeContainer = (SwipeRefreshLayout) findViewById(R.id.swipeContainer);  
// Setup refresh listener which triggers new data loading  
swipeContainer.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {  
    @Override  
    public void onRefresh() {  
        // ===== INICIO CODIGO A COMPLETAR =====  
        // ===== FIN CODIGO A COMPLETAR =====  
    }  
});
```

### EJERCICIO 5. 15% Mostrar el detalle de un libro

Para este último ejercicio, se conectará la lista de libros obtenida del servidor con la actividad del detalle del libro.

Modifica el método “onBindViewHolder” del adapter para pasar la posición de libro actual a la actividad de detalle (el resto del código puedes dejarlo tal y como se ha autogenerado) a través del Intent:

```
@Override  
public void onBindViewHolder(final ViewHolder holder, final int position) {  
    // ...  
    // ===== INICIO CODIGO A COMPLETAR =====  
    holder.mView.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            // ...  
        }  
    });  
    // ===== FIN CODIGO A COMPLETAR =====  
}
```

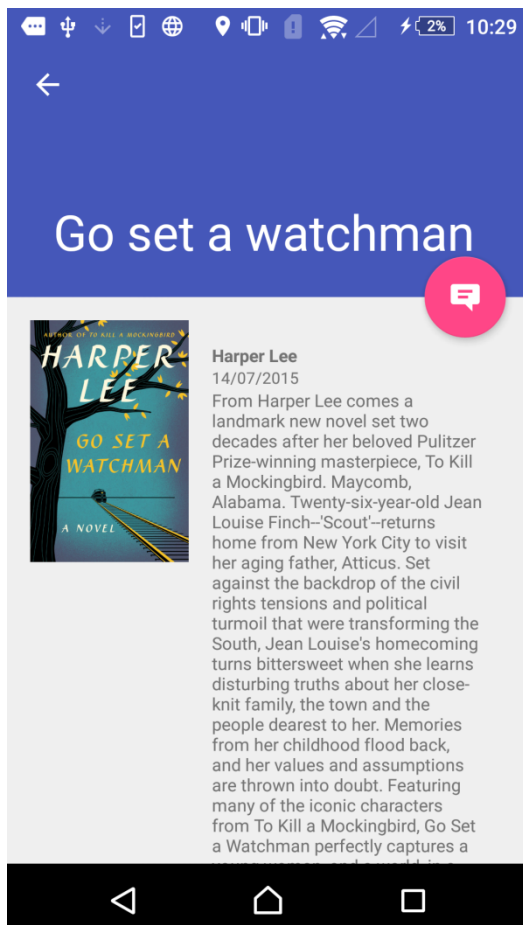
## Pla de Qualificació en Tecnologia Mòbil

En el fragment del detalle de los libros “BookDetailFragment” habrá que leer la posición del libro seleccionado de los argumentos y obtener todos los datos del libro:

@Override

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
  
    if (getArguments().containsKey(ARG_ITEM_ID)) {  
        // ===== INICIO CODIGO A COMPLETAR =====  
        // ===== FIN CODIGO A COMPLETAR =====  
    }  
}
```

Una vez tenemos los datos del libro seleccionado, completa los campos mostrados en este fragment con los detalles correspondientes, tal y como se muestra en la siguiente imagen:



## **EJERCICIO Equipo. 20% UX**

Veremos los vídeos de la siguiente lista durante la primera semana:

[https://www.youtube.com/playlist?list=PLAwTw4SYaPmQ\\_E3Z9WvTUgQ8rpjZqel3](https://www.youtube.com/playlist?list=PLAwTw4SYaPmQ_E3Z9WvTUgQ8rpjZqel3)

Una vez los hayáis visto, crearéis un nuevo tablero de Trello (probablemente crearé equipos nuevos para este ejercicio) en el que basándoos en lo mostrado en los vídeos, busquéis una utilidad a la aplicación MyBooks que estamos haciendo con la que tendría sentido publicarla en la PlayStore, por ejemplo:

- App de libros de segunda mano, en la que el usuario anónimo puede acceder a la lista de libros que tiendas de segunda mano o particulares que quieren venderlos una vez leídos, los comparten en la app. Podemos mostrar un mapa y aprovechar la geolocalización para que los usuarios puedan buscar a alguien cerca, también se podría contemplar la opción de como monetizar la app, si sería mejor hacer una versión gratis para particulares y otra de pago para tiendas de segunda mano...

Se crearía un documento con los casos de uso, la experiencia de usuario, el usuario potencial, etc... utilizando el modelo que se presenta en los vídeo (no hace falta utilizar prezi).

### **Recursos de aprendizaje vinculados:**

<b>Documentos/ Recursos para la adquisición de conocimientos</b>	<b>Herramientas</b>
<ul style="list-style-type: none"><li>• Módulo 6: Persistencia de datos.</li><li>• Módulo 9: Uso de librerías</li></ul> Complementarios: <ul style="list-style-type: none"><li>• Documentación oficial de Android</li><li>• Documentación oficial de Firebase</li></ul>	<ul style="list-style-type: none"><li>• Android Studio</li></ul>

## Criterios de evaluación

- La práctica se debe resolver **de forma individual**
- **El código fuente se debe comentar**

## Competencias técnicas

Competencias	Dimensiones	Indicadores de aprendizaje
Programar en lenguaje Java	Utilizar los elementos propios de la programación orientada a objeto	Usar correctamente la sintaxis del lenguaje Java
		Programar en Java mediante los elementos propios del lenguaje, como clases, propiedades, variables, estructuras de control de flujo, funciones...
	Realizar el diseño y la arquitectura lógica de aplicaciones móviles	Identificar los objetos, clases, atributos, métodos, relaciones e interfaces de los componentes software a utilizar.
		Utilizar las características propias de la programación orientada a objeto, como la herencia y el polimorfismo
	Utilizar las APIs y los entornos de desarrollo propios de Android (Android Studio)	Codificar las clases con las características detalladas en el diseño, haciendo uso de Android Studio para las activities, los fragments, layouts, etc.
	Utilizar servicios y APIs para conectar la aplicación con elementos externos y con un back-end, para poder así acceder a datos externos o almacenar información de la aplicación.	Realizar comunicaciones síncronas a elementos externos o un back-end
Utilizar herramientas de gestión y control de versiones de código: GIT		Realizar comunicaciones asíncronas a elementos externos o un back-end
		Intercambiar información mediante XML o JSON
		Gestionar ramas o branches
Adoptar una estrategia de desarrollo incremental (Scrum) para hacer más ágil el proceso de diseño y programación de la app.	Desarrollar y gestionar un proyecto utilizando metodología ágil.	Participar de manera eficiente en un equipo ágil a través de sus distintos roles
		Conocer y utilizar la tecnología y las herramientas que permiten soportar los distintos métodos y técnicas ágiles
Orientar el desarrollo de aplicaciones móviles en negocio para poder monetizar nuestra	Diseñar modelos de negocio y marketing orientados a aplicaciones móviles	Entender los diferentes modelos de negocio de la industria de las apps móviles
		Conocer las características de un



## Pla de Qualificació en Tecnologia Mòbil

aplicación		plan de marketing para un producto móvil
------------	--	--

### Soft Skills

Soft Skills	Dimensiones	Escala de evaluación			
Trabajo en equipo		0%	30%	70%	100%
		No alcanza ningún indicador	Alcanza sólo algunos indicadores	Alcanza la mayor parte de los indicadores	Alcanza todos los indicadores
	Planificar y gestionar el tiempo para cumplir los objetivos fijados por el equipo.	<ul style="list-style-type: none"> <li>- Actúa de acuerdo con los criterios establecidos por el equipo.</li> <li>- Realiza las tareas encargadas en el tiempo previsto.</li> <li>- Se coordina con el resto de miembros que forman parte del equipo.</li> </ul>			
	Promueve el beneficio del equipo	<ul style="list-style-type: none"> <li>- Participa activamente en la toma de decisiones sobre el funcionamiento y las actividades del grupo, favoreciendo la distribución equitativa del trabajo.</li> <li>- Comparte los conocimientos y la información con el resto del equipo.</li> <li>- Aplica estrategias, herramientas y recursos que facilitan los procesos de trabajo colaborativo en línea.</li> </ul>			
	Hacer el seguimiento del trabajo del equipo e introducir los cambios necesarios para alcanzar los objetivos.	<ul style="list-style-type: none"> <li>- Participa en la definición, el análisis del trabajo y la identificación de mejoras.</li> <li>- Facilita el proceso de adaptación del equipo a situaciones cambiantes implicándose en los cambios acordados.</li> </ul>			

Búsqueda, gestión y uso de la información		0%	30%	70%	100%
		No alcanza ningún indicador	Alcanza sólo algunos indicadores	Alcanza la mayor parte de los indicadores	Alcanza todos los indicadores
	Identificar cuáles son las necesidades de información.	<ul style="list-style-type: none"> <li>- Es capaz de definir un objetivo o necesidad de información, datos o recursos de manera clara.</li> <li>- Seleccionar adecuadamente fuentes de información relacionadas con el ámbito de estudio y / o aplicación.</li> </ul>			
	Aplicar estrategias para la búsqueda de la información necesaria.	<ul style="list-style-type: none"> <li>- Busca en bases de datos, repositorios, buscadores ... datos, información y recursos de manera efectiva.</li> <li>- Evalúa de manera crítica la adecuación y calidad de las fuentes consultadas e identifica las que son relevantes para su búsqueda.</li> <li>- Utiliza las comunidades en redes profesionales y foros temáticos del ámbito de estudio de manera efectiva.</li> </ul>			

## Pla de Qualificació en Tecnologia Mòbil

	Interpretar, analizar y relacionar esta información para saber aplicarla al problema o necesidad.	<ul style="list-style-type: none"> <li>- Es capaz de dar solución a la situación, caso o problema iniciales, adaptando la información encontrada.</li> <li>- Organiza y construye nuevo conocimiento de manera autónoma, integrando información, datos o recursos.</li> </ul>
--	---	---

Resolución de problemas		0%	30%	70%	100%
		No alcanza ningún indicador	Alcanza sólo algunos indicadores	Alcanza la mayor parte de los indicadores	Alcanza todos los indicadores
	Definir y establecer una comprensión / visión compartida del problema a resolver.	<ul style="list-style-type: none"> <li>- Discute el problema: hace preguntas y responde a las preguntas de los demás.</li> <li>- Se comunica durante el período de seguimiento y la resolución del trabajo en equipo.</li> </ul>			
	Emprender acciones adecuadas para la resolución del problema.	<ul style="list-style-type: none"> <li>- Describe y discute tareas y participa en su asignación.</li> <li>- Realiza los planes junto con los demás y realiza las acciones del rol asignado.</li> <li>- Hace el seguimiento de las acciones realizadas.</li> </ul>			