

1.- Descripción de la tarea

Caso práctico



elaboración propia ([CC BY-NC](#))

Ada, **jefa de BK Programación**, ha decidido que su equipo debe mejorar en el uso de **contenidos multimedia** para que las aplicaciones móviles sean **más atractivas y competitivas en un mercado saturado**. Sabe que una app visualmente impactante y funcional es clave para captar la atención de los usuarios.

Para ello, ha propuesto un ejercicio práctico en el que **Juan y María**, programadores junior, desarrollen una **guía de inicio para una aplicación Android**. Ada, consciente de que ambos son **fanáticos de los videojuegos**, ha elegido como proyecto una guía para una pequeña app de **Spyro The Dragon** que ya tiene desarrollada. Este ejercicio les permitirá **explorar herramientas multimedia**, mejorando la experiencia de usuario y asegurando que la app destaque frente a la competencia.

Con este enfoque, Ada busca que su equipo comprenda la **importancia de crear aplicaciones atractivas y funcionales**, maximizando su impacto en los usuarios y el mercado.

Actualización de mejora a una App de Spyro The Dragon

En esta actividad actualizaremos una app inspirada en el universo de **Spyro the Dragon** para hacerla más atractiva y fácil de usar. Para lograrlo, desarrollaremos dos elementos clave: una [Guía de inicio interactiva](#) y un [Easter Egg](#).

La **guía de inicio interactiva** presentará las principales funcionalidades de la app de manera visual y dinámica, utilizando animaciones, sonidos y elementos gráficos para enriquecer la experiencia del usuario.

Por su parte, el **easter egg** añadirá un toque de sorpresa y entretenimiento, creando momentos divertidos y memorables para los usuarios.

-Introducción

Para empezar, **deberéis descargar el código de la app de Spyro** en vuestro IDE **Android Studio**. A continuación, se detallan los pasos a seguir:

- **Ejecuta Android Studio.**
- En el menú superior, selecciona **"File > New > Project From Version Control..."**.
- Se abrirá un diálogo donde debes seleccionar **"Git"** en **"Version Control"**. Luego, introducir este enlace: ["https://github.com/lbarmar/SpyroTheDragon.git"](https://github.com/lbarmar/SpyroTheDragon.git) en el

apartado de **"URL"** y, en el apartado **"Directory"**, seleccionar la carpeta donde se guardará el proyecto.

- Pulsa en **"Clone"**

Por último:

- **Identificación:** Modifica el nombre del "About" para que se muestre el tuyo.
- **Prueba de la App:** ejecuta la app para comprobar que todo ha ido bien y explora todas su navegación en el emulador y la estructuración de su código.

-Apartado A. Guía de inicio interactiva - Diseños XML

La guía estará compuesta por las siguientes pantallas, que las tendréis que crear mediante layouts XML:

- **Pantalla 1:** Bienvenida a la guía interactiva e introducción a la aplicación:

Diseña una pantalla inicial con un fondo personalizado utilizando la clase Drawable, inspirado en el universo de Spyro the Dragon (por ejemplo, con tonos mágicos y elementos como cristales, tesoros o llamas).

Incluye un botón de "Comenzar" que inicie la guía, con una transición animada al pasar a la siguiente pantalla.

A continuación se incluye un [mockup](#) como ejemplo ilustrativo, que sirve de inspiración para que cada alumno lo adapte según su propio estilo y preferencias.



- **Pantalla 2:** Explicación de la pestaña Personajes (primer tab).

Implementa un bocado informativo que explique la sección: *"Aquí podrás explorar a todos los personajes del mundo de Spyro."*. Este bocado deberá contener alguna animación.

- **Pantalla 3:** Explicación de la pestaña Mundos (segundo tab).

Muestra un bocado informativo que señale los mundos disponibles y explique la sección. Este bocado deberá contener alguna animación. (Puedes hacer que el usuario pulse la pestaña para mostrar su contenido o que la guía lo haga automático)

- **Pantalla 4:** Explicación de la pestaña Coleccionables (tercer tab).

Implementa un bocado informativo que señale los coleccionables y explique la sección. Este bocado deberá contener alguna animación. (Puedes hacer que el usuario pulse la pestaña para mostrar su contenido o que la guía lo haga automático)

- **Pantalla 5:** Explicación del icono de información (icono mostrado en la parte derecha de la action bar)

Implementa un bocado informativo que señale el icono y explique la sección. Este bocado deberá contener alguna animación. (Puedes hacer que el usuario pulse el icono para mostrar su contenido o que la guía lo haga automático)

- **Pantalla 6:** Resumen final de la guía y cierre. Aquí se enumeran los pasos completados por el usuario y se muestra el botón para comenzar a usar la aplicación

Las pantallas 2, 3, 4 y 5 deben superponerse a la aplicación para guiar al usuario y resaltar las secciones que se explicarán. A continuación, se muestran dos ejemplos visuales para ilustrar cómo debería verse.



Pista: Para que las pantallas puedan mostrarse encima de la App, acuerdate de que los XML NO ocupen toda la pantalla y utiliza transparencias.

-Apartado B: Guía de inicio interactiva - Navegación

- **Avance en la guía:** Añade un botón (o elemento similar) en cada pantalla de la guía que permita avanzar a la siguiente pantalla.
- **Omitir la guía:** Incluye un botón (o elemento similar) en cada pantalla para que el usuario pueda omitir o saltar la guía si lo desea.
- **Bloqueo de interacción:** Asegúrate de que no se pueda interactuar con la app mientras la guía está activa, para mantener el enfoque del usuario en las explicaciones.
- **Mostrado único:** La guía debe mostrarse solo la primera vez que se abre la app. Como sugerencia, puedes usar SharedPreferences para registrar si el usuario ya completó la guía, evitando que vuelva a mostrarse.

-Apartado C: Guía de inicio interactiva - Animaciones

- **Animaciones para los bocadillos:** Implementa animaciones que hagan más dinámicos y atractivos los bocadillos en las pantallas 2, 3, 4 y 5. Por ejemplo, puedes usar efectos de aparición, deslizamiento o ampliación al mostrarlos.

- **Transiciones entre pantallas:** Añade transiciones visuales al cambiar entre pantallas de la guía. Escoge el tipo de transición que mejor se adapte al diseño, como desvanecimientos, desplazamientos laterales o escalas.

-Apartado D: Guía de inicio interactiva - Sonidos

- **Sonidos temáticos:** Incluye efectos de sonido relacionados con la temática de Spyro the Dragon que se reproduzcan en momentos clave de la guía. Por ejemplo, puedes usar sonidos al avanzar de pantalla, interactuar con los bocadillos o completar la guía.

-Apartado E: Easter Egg

Se crearán 2 en la aplicación.

- **Easter Egg con vídeo:**
Ubicado en la pestaña de **coleccionables**, se activará cuando el usuario pulse **cuatro veces consecutivas** sobre las Gemas. Al activarse, se reproducirá un **video temático** en pantalla completa. Al finalizar el video, el usuario será redirigido automáticamente a la pestaña de coleccionables.
- **Easter Egg con animación:**
Ubicado en la pestaña de **personajes**, se activará al realizar una **pulsación prolongada** sobre el personaje Spyro. Al activarse, se mostrará una **animación creada con Canvas** que simulará una llama de fuego saliendo de la boca de Spyro.

-Apartado F. Creación y Subida de un Archivo README.md para el Proyecto en GitHub

Para documentar tu proyecto y proporcionar una visión clara de su desarrollo y funcionalidades, es necesario crear un archivo **README.md** en el repositorio de GitHub del proyecto. Este archivo servirá para explicar el propósito de la aplicación, las tecnologías utilizadas y las conclusiones del desarrollo.

1. **Sube el Proyecto a GitHub:**
 - **Eliminar el repositorio remoto actual:** Ve a VCS > Git > Remotes y elimina la URL del repositorio remoto (porque será el GitHub del profesorado).
 - **Crear un nuevo repositorio en GitHub:** Crea el repositorio y copia su URL.
 - **Vincular al nuevo repositorio:** En VCS > Git > Remotes, agrega la nueva URL como remoto.
 - **Subir el proyecto:** Realiza un Commit y usa Git > Push para subir los archivos al nuevo repositorio.
2. **Crear el archivo README.md directamente desde GitHub:**
 - Una vez que hayas subido los ficheros del proyecto, verás un botón llamado **"Add a README"** justo debajo de los ficheros de tu repositorio. Haz clic en ese botón.
3. **Contenido del README.md:** Asegúrate de incluir los siguientes apartados y la información correspondiente en el archivo **README.md**:
 - **Introducción:** Explica brevemente el propósito de la aplicación y qué hace.

- **Características principales:** Describe las funcionalidades clave de la aplicación, como la autenticación, la Pokédex, la lista de Pokémon capturados y los ajustes.
- **Tecnologías utilizadas:** Enumera las principales tecnologías y librerías empleadas en el proyecto (como Firebase, Retrofit, RecyclerView, etc.).
- **Instrucciones de uso:** Proporciona pasos para clonar el repositorio e instalar cualquier dependencia necesaria para ejecutar la aplicación.
- **Conclusiones del desarrollador:** Reflexiona sobre el proceso de desarrollo y cualquier desafío o aprendizaje obtenido durante el proyecto.
- **Capturas de pantalla** (opcional): Si lo deseas, agrega imágenes que muestren la interfaz o funcionalidades de la app.

4. Formato Markdown:

- Usa formato [Markdown](#) para estructurar el archivo y darle una apariencia más legible (por ejemplo, usa títulos con #, listas con - o *, y texto en negrita con **).

5. Guardar el archivo README.md:

- Después de completar el archivo, haz clic en "**Commit changes...**" para guardarlo en tu repositorio.

3.- Información de interés

Recursos necesarios y recomendaciones

Para realizar esta tarea es necesario utilizar y consultar:

- Los contenidos y ejemplos de la unidad.
- Ordenador personal.
- Instalar Git.
- Crear una cuenta en el repositorio GitHub.

Antes de ponerse a implementar la aplicación solicitada, es muy recomendable:

- Leer los contenidos de la unidad e ir anotando en un cuaderno las ideas y conceptos más relevantes de cada capítulo.
- Cada vez que tengas dudas o no sepas como seguir, vuelve a tu cuaderno de notas y busca la información que necesitas. Si no la tienes reflejada, actualiza tus notas con los contenidos de la unidad. Todo lo que recopiles en ese cuaderno te será de gran ayuda para estudiar los exámenes.

TUTORIAL GIT Y GITHUB

Paso 1: Crear una cuenta en GitHub

1. Ve a la página principal de **GitHub**: github.com.
2. Haz clic en **Sign up** (Registrarse) en la esquina superior derecha.

3. Rellena los campos con tu información:
 - **Nombre de usuario:** Escoge un nombre de usuario único.
 - **Correo electrónico:** Ingresa tu dirección de correo.
 - **Contraseña:** Crea una contraseña segura.
4. Sigue los pasos para verificar tu correo y terminar el registro.

Paso 2: Instalar Git en tu computadora

Antes de usar GitHub con Android Studio, necesitas tener **Git** instalado en tu máquina.

- **Para Windows:**
 1. Ve a git-scm.com y descarga el instalador.
 2. Ejecuta el instalador y sigue las instrucciones. En la mayoría de los casos, puedes dejar las opciones predeterminadas.

Paso 3: Configurar Git en Android Studio

Ahora vamos a configurar Git en **Android Studio**.

1. Abre **Android Studio** y carga el proyecto que quieres subir a GitHub.
2. En el menú superior, ve a **File > Settings** (o **Android Studio > Preferences** en macOS).
3. En la ventana que aparece, busca **Version Control > Git**.
4. En el campo **Path to Git executable**, debería estar configurado automáticamente, pero si no lo está, navega al archivo ejecutable de Git que instalaste (por ejemplo, en Windows, suele estar en C:/Program Files/Git/bin/git.exe).
5. Haz clic en **Test** para asegurarte de que Git está configurado correctamente.

Paso 4: Crear un repositorio en GitHub

Ahora necesitas crear un repositorio en GitHub para almacenar tu proyecto.

1. Ve a **GitHub** y haz clic en el botón + en la esquina superior derecha y selecciona **New repository**.
2. Completa los campos:
 - **Repository name:** Elige un nombre para tu repositorio (puede ser el nombre del proyecto).
 - **Description:** (Opcional) Añade una breve descripción de tu proyecto.
 - **Public:** Selecciona esta opción si quieres que otras personas puedan ver tu repositorio.
 - **Initialize this repository with a README:** **No selecciones esta opción** (ya que lo inicializaremos desde Android Studio).
3. Haz clic en **Create repository**.

Paso 5: Conectar Android Studio con GitHub

Para que Android Studio pueda comunicarse con GitHub, sigue estos pasos:

1. En Android Studio, ve a **File > Settings > Version Control > GitHub**.
2. Haz clic en el botón **Add account**.
3. En la ventana emergente, selecciona **Log in via GitHub**.
4. Se abrirá una ventana del navegador. Inicia sesión con tu cuenta de GitHub.
5. Autoriza la aplicación para que Android Studio pueda acceder a tu cuenta de GitHub.
6. Una vez autenticado, tu cuenta de GitHub debería aparecer listada en la ventana de configuración de GitHub.

Paso 6: Iniciar el control de versiones (Git) en el proyecto

1. En Android Studio, abre el proyecto que deseas subir a GitHub.
2. Ve al menú superior y selecciona **VCS > Enable Version Control Integration**.
3. En la ventana emergente, selecciona **Git** y haz clic en **OK**.
4. Ahora Android Studio ha convertido tu proyecto en un repositorio de Git local.

Paso 7: Hacer un commit del proyecto

1. Ve a **VCS > Commit** o usa el atajo de teclado Ctrl + K (o Cmd + K en macOS).
2. Selecciona los archivos que deseas incluir en el commit (deberían ser todos los archivos del proyecto).
3. Añade un mensaje descriptivo en el campo **Commit Message** (por ejemplo, "Initial commit").
4. Haz clic en **Commit** (puede que se te pida resolver advertencias o errores, si es el caso, sigue las instrucciones).
5. Después de hacer el commit, selecciona **VCS > Git > Push** para subir estos cambios a GitHub.

Paso 8: Subir el proyecto a GitHub

1. En el menú superior, ve a **VCS > Git > Push**.
2. En la ventana emergente, selecciona **Define remote**.
3. Ahora debes añadir el repositorio remoto:
 - Ve a tu repositorio en GitHub y copia la URL (algo como <https://github.com/tu-usuario/nombre-repositorio.git>).
 - Vuelve a Android Studio y pega esa URL en el campo **Remote** y asigna un nombre como **origin**.
4. Haz clic en **Push**.
5. Si todo ha salido bien, verás un mensaje de confirmación de que el proyecto ha sido subido con éxito a GitHub.

Paso 9: Compartir el enlace de tu repositorio

1. Ve a **GitHub** y navega al repositorio que acabas de subir.
2. Copia la URL del repositorio (Ejemplo -> <https://github.com/tu-usuario/nombre-repositorio>).
3. Comparte este enlace con tu profesorado via Moodle. El profesorado podrá acceder al enlace y ver el contenido del proyecto si lo has configurado como **público**.
5. Agregar entrega

7. Estado de la entrega

- 8.
9. **Resultados de aprendizaje o criterios de evaluación trabajados:**
 - 3.a) Se han analizado entornos de desarrollo multimedia. : -
 - 3.b) Se han reconocido las clases que permiten la captura, procesamiento y almacenamiento de datos multimedia. : -
 - 3.c) Se han utilizado clases para la conversión de datos multimedia de un formato a otro. : -
 - 3.d) Se han utilizado clases para procesar datos multimedia. : -
 - 3.e) Se han utilizado clases para el control de eventos, tipos de media y excepciones, entre otros. : -
 - 3.f) Se han utilizado clases para la creación y control de animaciones. : -
 - 3.g) Se han utilizado clases para reproducir contenidos multimedia. : -
 - 3.h) Se han depurado y documentado los programas desarrollados.: -
- 10.

Número del intento	Este es el intento 1.
Estado de la entrega	Todavía no se han realizado envíos
Estado de la calificación	Sin calificar
Tiempo restante	La Tarea está retrasada por: 17 días 9 horas
Fecha límite	martes, 4 de marzo de 2025, 23:55
Última modificación	-

Comentarios de la entrega	Comentarios (0)
---------------------------	---------------------------------

11. Criterios de calificación

Rúbrica		
(RA3.a, RA3.h) Diseños XML y documentación README.md	<p>** No conseguido: ** Implementación incorrecta o incompleta, con errores graves que afectan la funcionalidad.</p> <p><i>0 puntos</i></p>	<p>**Casi conseguido:** Implementación correcta o con errores menores.</p> <p><i>0.8 puntos</i></p>
(RA3.b, RA3.e) Navegación y control de interacción	<p>** No conseguido: ** Implementación incorrecta o incompleta, con errores graves que afectan la funcionalidad.</p> <p><i>0 puntos</i></p>	<p>**Casi conseguido:** Implementación correcta o con errores menores.</p> <p><i>0.8 puntos</i></p>
(RA3.c) Animaciones y efectos de sonido	<p>** No conseguido: ** Implementación incorrecta o incompleta, con errores graves que afectan la funcionalidad.</p> <p><i>0 puntos</i></p>	<p>**Casi conseguido:** Implementación correcta o con errores menores.</p> <p><i>0.8 puntos</i></p>
(RA3.d, RA3.g) Easter Egg Vídeo	<p>** No conseguido: ** Implementación incorrecta o incompleta, con errores graves que afectan la funcionalidad.</p> <p><i>0 puntos</i></p>	<p>**Casi conseguido:** Implementación correcta o con errores menores.</p> <p><i>0.8 puntos</i></p>
(RA3.f) Easter Egg Animación	<p>** No conseguido: ** Implementación incorrecta o incompleta, con errores graves que afectan la funcionalidad.</p> <p><i>0 puntos</i></p>	<p>**Casi conseguido:** Implementación correcta o con errores menores.</p> <p><i>0.8 puntos</i></p>