

**EXAMEN FINAL**

APELLIDOS Y NOMBRES	Abel Luciano Aragon Alvaro		
CURSO	Programación para Dispositivos Móviles	SEMESTRE	Sexto
DOCENTE	Josue Miguel Flores Parra	FECHA: 07-2024	
DURACIÓN	Entregar antes del 07 de Julio del 2024		

Instrucciones. Examen Final (Proyecto Final).**Descripción del Proyecto**

Los estudiantes deben desarrollar una aplicación móvil funcional en equipos de hasta tres integrantes. La aplicación debe resolver un problema real o satisfacer una necesidad específica, siendo innovadora y útil para sus usuarios potenciales.

Requerimientos Generales

1. **Plataforma:** La aplicación debe ser desarrollada para Android.
2. **Lenguaje y Framework:** La aplicación debe ser desarrollada en Kotlin utilizando Android Studio.
3. **Repositorio:** El proyecto debe estar alojado en un repositorio de GitHub, siendo público o privado según se indique. Todos los commits deben ser relevantes y descriptivos.
4. **Documentación:** La documentación debe estar bien estructurada y clara, incluyendo:
 - **README.md:** Instrucciones para la instalación y uso de la aplicación, descripción del proyecto y características principales.
 - **Documento Word.** Este debe contener portada donde incluya integrantes del equipo, título del proyecto, descripción del proyecto, características principales y capturas de pantallas. Además debe incluir información detallada sobre la arquitectura del proyecto, decisiones de diseño y cualquier otro aspecto relevante.

- **Comentarios en el Código:** El código debe estar adecuadamente comentado para facilitar su comprensión.

5. **Funcionalidades:**

- La aplicación debe incluir al menos tres funcionalidades principales.
- Debe contar con una interfaz de usuario intuitiva y amigable.
- Implementar la persistencia de datos (puede ser local o en la nube).
- En caso de no requerir una base de datos, utilizar SharedPreferences para guardar pequeñas cantidades de datos en el almacenamiento interno de Android.

Criterios de evaluación.

El examen final (Proyecto Final) se valorará de acuerdo a la rúbrica adjunta. Recuerde que esta rubrica solo es necesaria una respuesta por equipo. Por lo que deben llegar a un consenso de puntaje entre los miembros del equipo.

Entregables

1. **Repositorio de GitHub.** URL del repositorio donde se encuentra el código fuente y la documentación.
2. **Documento Word ó PDF.** Documentación del proyecto anteriormente descrita (Ver punto 4 de Requerimientos Generales).
3. **Presentación del Proyecto.** Video del proyecto de aproximadamente 5 a 10 minutos, donde los estudiantes muestren la aplicación y expliquen su funcionamiento, pueden mencionar decisiones de diseño y cualquier desafío enfrentado durante el desarrollo.
4. **Rúbrica.** Responder la rúbrica adjunta, una por equipo.

Rúbrica para Examen Final.

Contenido y demostración		Puntos	Checklist	Equipo	Profesor
1. Estructura del Proyecto.	<ul style="list-style-type: none"> • El proyecto tiene una estructura clara y organizada. • Los archivos están ubicados en carpetas adecuadas (por ejemplo, actividades en una carpeta, recursos en otra, etc.). • Se utilizan nombres de archivos y carpetas descriptivos y coherentes. 	3			
2. Buenas prácticas de programación	<ul style="list-style-type: none"> • Se siguen las convenciones de nomenclatura de Kotlin (camelCase para variables y funciones, UpperCamelCase para clases y LowerCamelCase para nombres de métodos). • Se evitan repeticiones de código innecesarias mediante la reutilización de funciones y clases. • Se utilizan tipos de datos adecuados para cada variable y función. • Se aplican principios de diseño de software como encapsulamiento, modularidad y cohesión. 	4			
3. Comentarios	<ul style="list-style-type: none"> • El código está debidamente comentado, explicando la lógica detrás de cada función, clases, métodos, variables y sección relevante. • Se incluyen comentarios en el código para aclarar decisiones de diseño o posibles mejoras. • Los comentarios están escritos de manera clara y concisa, utilizando gramática y ortografía correcta. 	3			
4. Github	<ul style="list-style-type: none"> • Se utiliza Github para controlar la versión del proyecto. • Se hacen commits frecuentes y con mensajes descriptivos. • Se incluye un archivo README.md que describe el proyecto y cómo ejecutarlo. Además Incluye un enlace al repositorio de GitHub en el archivo README.md y se mantiene actualizado. 	3			
5. Funcionalidad	<ul style="list-style-type: none"> • El proyecto cumple con los requisitos funcionales establecidos. • Se implementan las funcionalidades de manera correcta y eficiente. • La aplicación es fácil de usar y/o comprende una interfaz de usuario intuitiva. • Se manejan correctamente los casos de error y excepciones. 	4			
6. Documentación y Video Presentación.	<ul style="list-style-type: none"> • La documentación incluye portada, título, descripción, características principales y capturas de pantallas. • La documentación incluye información detallada sobre la arquitectura del proyecto, decisiones de diseño, etc. • Presenta un video de duración aproximada de 5 a 10 minutos, donde se muestra la aplicación y explica su funcionamiento. 	3			
Total		20			