



**Materia:** Programacion Movil

**Nombre** Reporte de Proyecto final

**Informe:**

**Alumno(s):** Patrick Alexander Baeza Zamudio ,Francisco Rociles Magaña,  
Juan Carlos Nava Aguilera, Ana Laura Felipe Juarez

**Fecha:** 02/10/24

## Objetivo

Avance del Proyecto Final

## Temas del plan de estudios

TEMA 2:

2.1

## Material

Androi Estudio, Githgb

## Marco Teórico

El desarrollo de aplicaciones móviles es una disciplina en constante evolución debido al creciente uso de dispositivos móviles en todo el mundo. La creación de aplicaciones para smartphones, como Android, requiere una combinación de conocimientos sobre diseño de interfaz, arquitectura de software y usabilidad. Este marco teórico cubre los aspectos más relevantes para el desarrollo de una aplicación móvil enfocada en la gestión de notas y tareas multimedia con recordatorios.

## Desarrollo

Las aplicaciones de gestión de notas y tareas son herramientas esenciales para organizar y planificar actividades diarias. Estas aplicaciones permiten al usuario registrar información de manera sencilla, ordenarla, buscarla y recibir recordatorios para garantizar que las tareas se completen a tiempo. Entre las principales funcionalidades que debe incluir una aplicación de este tipo se encuentran:

Creación de notas: Se refiere al proceso de registrar texto, multimedia y otras formas de información estructurada.

Creación de tareas: Las tareas generalmente incluyen una fecha límite y la opción de recibir recordatorios para cumplirlas a tiempo.

Adjuntos multimedia: La capacidad de adjuntar imágenes, videos y audios mejora la versatilidad de la aplicación y permite al usuario registrar más tipos de información.

## Resultado

Funcionales

ID	Nombre	Descripción	Tipo
RF1	Registro de notas	Permitir al usuario registrar notas con título, descripción y archivos multimedia adjuntos (fotos, videos, audios)	Funcional
RF2	Registro de tareas	Permitir al usuario registrar tareas con fecha, hora y recordatorios.	Funcional
RF3	Adjuntar archivos multimedia	Permitir adjuntar archivos multimedia con descripción (fotos, videos, audios).	Funcional



RF4	Recordatorios programables	Permitir que el usuario programe múltiples recordatorios por tarea.	Funcional
RF5	Notificaciones	Enviar notificaciones cuando se cumple un recordatorio o la fecha de la tarea.	Funcional
RF6	Marcar tareas como cumplidas	Permitir marcar una tarea como cumplida o posponerla.	Funcional
RF7	Multidispositivo	La aplicación debe ser compatible con smartphones y tablets.	Funcional
RF8	Edición de notas y tareas	Permitir al usuario editar o eliminar notas y tareas previamente registradas.	Funcional
RF9	Búsqueda de notas y tareas	Permitir buscar notas y tareas por título y descripción.	Funcional
RF10	Ordenar notas y tareas	Las notas se ordenan por fecha de registro y las tareas por fecha de cumplimiento.	Funcional
RF11	Miniaturas de archivos adjuntos	Mostrar una miniatura representativa de cada archivo adjunto.	Funcional

No funcionales

ID	Nombre	Descripción	Tipo
RNF1	Compatibilidad con Android Jelly Bean	La aplicación debe ser compatible con dispositivos Android a partir de la versión Jelly Bean (API 16).	No Funcional
RNF2	Soporte multilenguaje	La aplicación debe soportar inglés y español.	No Funcional
RNF3	Fluidez y usabilidad	La aplicación debe ser fluida, minimizando el número de movimientos necesarios para registrar o visualizar información.	No Funcional
RNF4	Interfaz adaptable	La UI debe adaptarse correctamente a smartphones y tablets	No Funcional
RNF5	Uso de Material Design	La aplicación debe utilizar Material Design, soportando	No Funcional



dark mode, personalización  
de formas y tipografía

Código

Mercado  
10-8-2024  
Escuela  
10-9-2024  
Familiar  
10-10-2024

```
import ...  
  
@Composable  
fun MainScreen(notesViewmodel: Notes_ViewModel) {  
    val state = notesViewmodel.state  
    if(state.isLoading) {  
        Box(modifier = Modifier.fillMaxSize(), contentAlignment = Alignment.Center) {  
            CircularProgressIndicator()  
        }  
    }  
    LazyColumn(modifier = Modifier.fillMaxWidth()){  
        //items(Notes_ViewModel.notes){  
        items(state.Notes){  
            Column(modifier = Modifier.fillMaxWidth()){  
                Text(text = it.TitleNote)  
                Text(text = it.NoteDate)  
                Divider()  
            }  
        }  
    }  
}
```



```
import net.noble.notes_view_try.ViewModel_Screen_Notes.Notes_ViewModel
import net.noble.notes_view_try.ui.theme.Notes_View_TryTheme

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        val Notes_ViewModel by viewModels<Notes_ViewModel>()
        setContent {
            Notes_View_TryTheme {

                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    MainScreen(Notes_ViewModel)
                }
            }
        }
    }
}

@Composable
fun Greeting(name: String, modifier: Modifier = Modifier) {
    Text(
        text = "Hello $name!",
        modifier = modifier
    )
}

@Preview(showBackground = true)
@Composable
fun GreetingPreview() {
    Notes_View_TryTheme {
        Greeting("Android")
    }
}
```

## Evaluación

## Conclusiones

El desarrollo de una aplicación para gestionar notas y tareas multimedia con recordatorios en Android implica la integración de diversas tecnologías y prácticas de diseño. Utilizando Android Studio, Jetpack Compose y Material Design, es posible crear una aplicación moderna, funcional y estéticamente agradable, cumpliendo con los requerimientos del cliente en términos de usabilidad, rendimiento y compatibilidad. Además, la correcta gestión de notificaciones, almacenamiento y soporte multilinguaje garantizará una experiencia de usuario fluida y accesible.

## Bibliografía