



Carrera:

Ingeniería de software

Tema:

Proyecto Final – Daily Planner

Integrantes:

Byron Serrano – Joao Dueñas

Nivel:

Segundo semestre

Asignatura:

Aplicaciones Móviles

Docente:

Ing. Eugenio Morocho

Periodo:

2022-02

Link del repositorio de Github:

https://github.com/joaoduenas17/Daily_Planner_Finalized.git

Dentro del Github se encuentra el archivo APK ejecutable.

Capitulo	Descripción
Prologo	Esta documentación se encuentra en una versión temprana, por lo cual en la entrega del Proyecto final se entregará la documentación actualizada corrigiendo así ciertos errores que esta pueda contener.
Introducción	La aplicación Daily Planner es una sencilla pero robusta To Do List, que en español se traduciría como Lista de que hacer, la cual se basa en la creación de una lista que contenga todas las actividades o tareas que realicemos durante el día.
Glosario	To Do List: Lista de que hacer, la cuál ayuda organizar nuestro tiempo. Inputs: Datos que se ingresan a la aplicación. Outputs: Datos que se reciben de la aplicación. RecyclerView: Conjunto de vistas las cuales podemos reciclar para añadir cuantas queramos. Checkbox: Botón que sirve para poder darle una selección a algo.
Requerimientos de Usuario	Requerimientos Funcionales: Ingreso de nuevas tareas – Tachar las tareas – Editar los campos de las tareas Requerimientos no Funcionales: Compatibilidad con la mayoría de dispositivos Android, aproximadamente un 75% de los dispositivos

Estructura específica de un Requerimiento para Daily Planner:	
Función:	Ingreso de nuevas tareas
Descripción:	Se podrá ingresar nuevas tareas a la aplicación, usando un botón el cuál nos abrirá una pequeña ventana en donde ingresaremos los datos de la tarea y después se añadirá a la aplicación.
Inputs:	Darle click al botón añadir -> Ingresar datos de la tarea -> Dar click al botón guardar.
Outputs:	Tarea añadida a la vista de la aplicación.
Destination:	RecyclerView, txtTexto, ImageView, FloatingActionButton, CardView.
Action:	Se actualizará el ViewModel del RecyclerView para poder añadir un nuevo item a la lista, el cuál será añadido y pintado por el adapter y el ViewHolder.
Requerimientos:	Se necesitaría saber la tarea a realizar
Precondición:	El nombre de la tarea y una descripción de esta.
Postcondición:	Ninguno.
Side Effects:	Ninguno.

Estructura específica de un Requerimiento para Daily Planner:	
Función:	Tachar las tareas
Descripción:	Al darle click al signo del checkbox se tachará la tarea y se dará a entender que esa tarea ya fue realizada.
Inputs:	Dar click al signo del checkbox.
Outputs:	Se marcará el texto en la tarea.
Destination:	RecyclerView, txtTexto, ImageView, CardView.
Action:	Se le añadirá una función que tienen los textos para poder ser tachados al momento de darle click al ImageView del checkbox el cuál cambiará su imagen después del evento.
Requerimientos:	Haber creado una tarea.
Precondición:	Realización de la tarea con su nombre.
Postcondición:	Ninguno.
Side Effects:	Cambio en el ImageView del checkbox y en el tachado del texto.

Estructura específica de un Requerimiento para Daily Planner:	
Función:	Editar los campos de las tareas
Descripción:	Al presionar sobre la tarea se abrirá un Fragment el cuál nos señalará que podremos editar el nombre y la descripción de la tarea y actualizarla.
Inputs:	Presionar la tarea. Ingresar nuevo nombre. Ingresar nueva descripción. Presionar el botón actualizar
Outputs:	Se cambiará los campos de la tarea, el nombre y la descripción.
Destination:	RecyclerView, txtTexto, ImageView, CardView.
Action:	Se abrirá el Fragment pero con el texto del título y del botón cambiados, después se ejecutará la función <code>updateList()</code> el cual obtendrá los nuevos datos que se ingresaron y los cambiará por los antiguos datos.
Requerimientos:	Haber creado una tarea previamente.
Precondición:	Realización de la tarea con su nombre y descripción.
Postcondición:	Ninguno.
Side Effects:	Cambio dentro de los parámetros de nombre y descripción.