



El futuro digital
es de todos

MinTIC

Sprints: Desarrollo Aplicaciones Móviles.



Universidad
Pontificia
Bolivariana

Vigilada MinEducación

Misión
TIC 2022



Generalidades

- Recuerda reunirte con tu equipo de trabajo para determinar los acuerdos en la ejecución de las actividades.
- Ten presente avanzar en el valor agregado de tu proyecto. Si resuelven las historias de usuario en poco tiempo, podrán concentrar sus esfuerzos en los componentes diferenciales.
- Recuerda que estamos trabajando sobre funcionalidades básicas relacionadas con la lógica de negocio.
- Debes implementar la segunda versión del prototipo del proyecto en un aplicativo móvil y generar el repositorio compartido con tu equipo de trabajo.

Recomendaciones

1. Incluye las vistas, funcionalidades básicas y elementos de valor agregado en la segunda versión del proyecto.
2. Esta entrega se realizará a través del autocalificador codegrade.
3. Verifica los nombres de los archivos de entrega y la extensión.
4. Ingresen al espacio en plataforma “Actividad: Sprint 4” y sigan las instrucciones.

Introducción/Información:

Los seres humanos influyen cada vez mas en el clima y la temperatura de la tierra, según FAO, un 7% de las emisiones totales se produce mediante el desperdicio de alimentos que terminan directamente en la basura. Esto agrega enormes cantidades de gases de efecto invernadero a lo que incurre naturalmente en la atmosfera, aumentando el calentamiento global.

Objetivos:

- Resolver las historias de usuario propuestas como requisitos del proyecto de aplicaciones móviles asignado durante el ciclo.
- Repasar el concepto eventos, lógica de negocio y actualización de datos.

Contexto:

En los últimos años, han surgido varias plataformas que persiguen acabar con el desperdicio alimentario, tanto para aprovechar mejor los recursos como para intentar reducir el impacto de la comida en el cambio climático.

Historias de usuario:

Identificador Historia#:	HU-07	Título:	Diseño Inicio/Home de la aplicación
Descripción	COMO:	Usuario final	
	QUIERO:	Como usuario requiero ver todos los alimentos publicados de todos los usuarios	
	PARA:	Ver si alguno me interesa para solicitar su donación	
Criterios de aceptación	<div><div>1.</div><div>En la parte superior derecha se requiere un botón o un menú donde el usuario pueda cerrar la sesión de la aplicación</div></div> <div><div>2.</div><div>En la parte inferior se requiere un menú con los siguientes ítems (Home/Inicio, Perfil, publicaciones) estos deben navegar entre pantallas y mantenerse</div></div> <div><div>3.</div><div>Se deberá mostrar todas las publicaciones realizadas de todos los usuarios mediante una lista con las siguientes descripción: imagen del alimento, fecha de caducidad y nombre del alimento</div></div> <div><div>4.</div><div>La información requerida en la lista de publicación se deberá consultar mediante un servicio donde me regrese todos los alimentos disponibles</div></div> <div><div>5.</div><div>Al presionar uno ítem de la lista deberá mostrar un detalle del mismo</div></div> <div><div>6.</div><div>En la pantalla detalle se deberá mostrar: la imagen del alimento una descripción del mismo, nombre y la fecha de vencimiento</div></div>		

Sprint backlog

Historia de usuario HU-07

1. En la parte superior derecha se requiere un botón o un menú donde el usuario pueda cerrar la sesión de la aplicación

Para la implementación de esta funcionalidad puede utilizar:

```
@Override  
public boolean onCreateOptionsMenu(Menu menu) {...}  
  
@Override  
public boolean onOptionsItemSelected(MenuItem item) {...}
```

Sprint backlog

Historia de usuario HU-07

1. En la parte inferior se requiere un menú con los siguientes ítems (Home/Inicio, Perfil, publicaciones) estos deben navegar entre pantallas y mantenerse

Para realizar esta implementación deberá usar fragmentos e implementar en la activity principal colocar estas líneas de código:

```
AppBarConfiguration appBarConfiguration = new AppBarConfiguration.Builder(  
    R.id.navigation_home, R.id.navigation_perfil, R.id.navigation_publicaciones  
).build();  
NavController navController = Navigation.findNavController( activity: this, R.id.nav_host_fragment_activity_main_inicio);  
NavigationUI.setupActionBarWithNavController( activity: this, navController, appBarConfiguration);  
NavigationUI.setupWithNavController(binding.navView, navController);
```

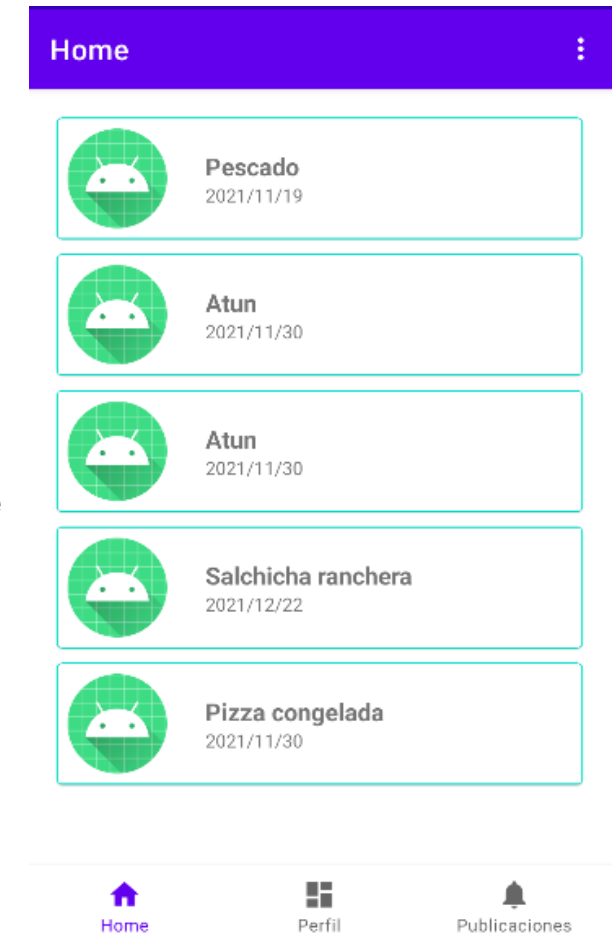
Sprint backlog

Historia de usuario HU-07

1. Se deberá mostrar todas las publicaciones realizadas de todos los usuarios mediante una lista con las siguientes descripción: imagen del alimento, fecha de caducidad y nombre del alimento

Para realizar esta implementación podrá utilizar el componente RecyclerView para mostrar una lista como esta:

la información requerida en la lista de publicación se deberá consultar mediante un servicio donde me regrese todos los alimentos disponibles



Sprint backlog

Historia de usuario HU-07

1. En la pantalla detalle se deberá mostrar: la imagen del alimento una descripción del mismo, nombre y la fecha de vencimiento

Deberá implementar una nueva activity y maquetar la siguiente información podrá pasar los parámetros o información con la siguiente sentencia:

```
intent.putExtra("llave", "valor");  
intent.putExtra("llave", "valor");
```

NO Desperdicio



Descripción: Resulta que esas fechas, comúnmente conocidas como fechas de caducidad o caducidad, no tienen nada que ver con la seguridad alimentaria. En cambio, indican las fechas en las que el fabricante estima que los alimentos habrán perdido calidad

Nombre: En latados

Fecha de vencimiento: 2022-03-28

LO QUIERO

Entrega:

Se debe entregar los siguientes archivos con el formato y nombre específicos

1. **PublicacionesDto.java**
2. **ControladorInicio.java**
3. **InicioInterface.java**

Entrega:

Se debe entregar los siguientes archivos con el formato y nombre específicos

1. ControladorInicio.java

```
public class ControladorInicio implements InicioInterface.Controlador {  
    private final InicioInterface.View view;  
  
    public ControladorInicio(InicioInterface.View view) {  
        this.view = view;  
    }  
  
    @Override  
    public void recuperarLista() {  
  
    }  
  
    @Override  
    public void salirApp() {  
  
    }  
}
```

Entrega:

Se debe entregar los siguientes archivos con el formato y nombre específicos

1. InicioInterface.java

```
public interface InicioInterface {  
    interface View {  
    }  
    interface Controlador {  
    }  
    interface Modelo {  
    }  
}
```



Entrega:

Se debe entregar los siguientes archivos con el formato y nombre específicos

1. PublicacionesDto.java

```
public class PublicacionesDTO {
    private String id;
    private String comentario;
    private String fecha;
    private String nombre_alimento;
    private String tipo_alimento;

    public PublicacionesDTO() {}

    public PublicacionesDTO(String id, String comentario, String fecha, String nombre_alimento, String
    tipo_alimento) {}

    }

    public String getId() {
    }

    public void setId(String id) {
    }

    public String getComentario() {
    }

    public void setComentario(String comentario) {
    }

    public String getFecha() {
    }

    public void setFecha(String fecha) {
    }

    public String getNombre_alimento() {
    }

    public void setNombre_alimento(String nombre_alimento) {
    }

    public String getTipo_alimento() {
    }

    public void setTipo_alimento(String tipo_alimento) {
    }
}
```

Material de apoyo:

- <https://developer.android.com/training/testing/ui-testing/espresso-testing>
- <https://www.vogella.com/tutorials/Mockito/article.html>