



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.
- A partir de este sprint serán abordadas temáticas relacionadas con la lógica de negocio y funcionalidad básica del aplicativo.

## Recomendaciones

1. Esta entrega se realizará a través del autocalificador codegrade.
2. Verifica los nombres de los archivos de entrega y la extensión.
3. Ingresen al espacio en plataforma “Actividad: Sprint 2” y sigan las instrucciones.
4. Define la estructura MVC para adelantar el proceso de desarrollo del proyecto móvil sugerido en estos enunciados.

## 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 manejo de login de seguridad, validación de datos y test unitarios.

## 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-04	Título:	Diseño login del usuario
Descripción	COMO:	Usuario final	
	QUIERO:	Una interfaz amigable para poder introducir los datos requeridos por la aplicación	
	PARA:	Ingresar al sistema y poder realizar las acciones requeridas	
Criterios de aceptación	<div><div>1.</div><div>Maquetar la pantalla de Login mediante las etiquetas XML y componentes nativos de Android</div></div> <div><div>2.</div><div>El campo de usuario y contraseña deben de tener mínimo una longitud de 5 caracteres</div></div> <div><div>3.</div><div>Al presionar el botón ingresar el sistema debe validar que no estén vacíos</div></div> <div><div>4.</div><div>Si el sistema no cumple con la longitud requerida se debe mostrar un error informando al usuario que debe colocar un usuario y contraseña mayor a 5 caracteres</div></div> <div><div>5.</div><div>El botón ingresar debe validar que el usuario exista y que los datos ingresados correspondan igual a los que se tiene previamente guardados</div></div> <div><div>6.</div><div>El diseño y colores son libres y pueden utilizar el que deseen, pero respetando la estructura establecida en el diseño</div></div> <div><div>7.</div><div>Se debe mostrar un indicador de carga mientras el programa realiza la validación del usuario</div></div> <div><div>8.</div><div>Se deben emplear pruebas unitarias del validador de los campos y el botón ingresar</div></div>		

## Sprint backlog

### Historia de usuario HU-04

Como usuario requiero tener una interfaz amigable para poder introducir los datos requeridos por la aplicación

1. Maquetar la pantalla de **Login** mediante las etiquetas XML y componentes nativos de Android

- Para esta implementación se puede apoyar en la siguiente documentación oficial tener en cuenta que debe seguir los diseños ya planteados en el Wireframe del sprint anterior
- <https://developer.android.com/guide/topics/ui/declaring-layout?hl=es-419>
- <https://autoridadandroid.com/una-introduccion-a-xml-para-nuevos-desarrolladores-de-android-el-potente-lenguaje-de-marcado/>

- Los id de cada componente se deben de nombrar de la siguiente manera:

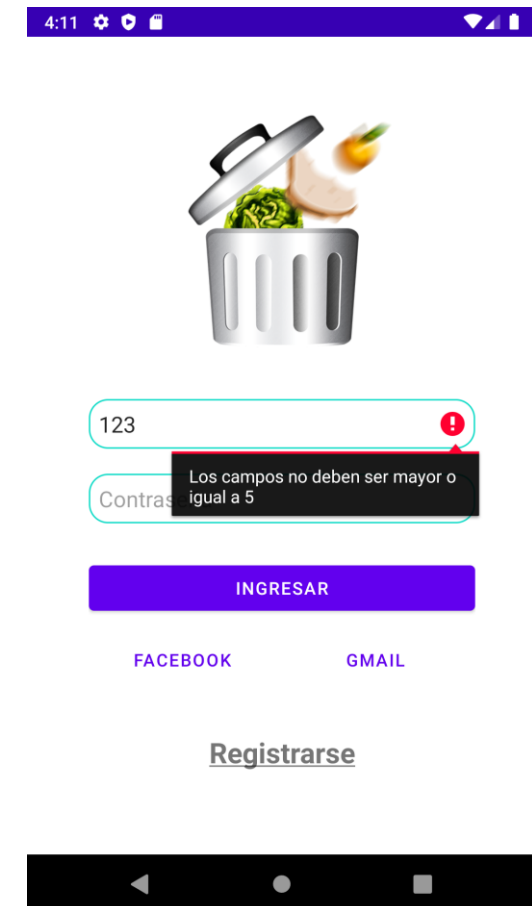
```
android:id="@+id/editUsuario"  
android:id="@+id/editPassword"  
android:id="@+id/btnIngresar"  
android:id="@+id/btnFacebook"  
android:id="@+id/btnGmail"  
android:id="@+id/txtRegistrarse"
```

## Sprint backlog

### Historia de usuario HU-04

Como usuario requiero tener una interfaz amigable para poder introducir los datos requeridos por la aplicación

2. El campo de usuario y contraseña deben de tener mínimo una longitud de 5 caracteres
  - Se deberá validar que **longitud**  $\leq 4$  y mostrar un mensaje como el siguiente



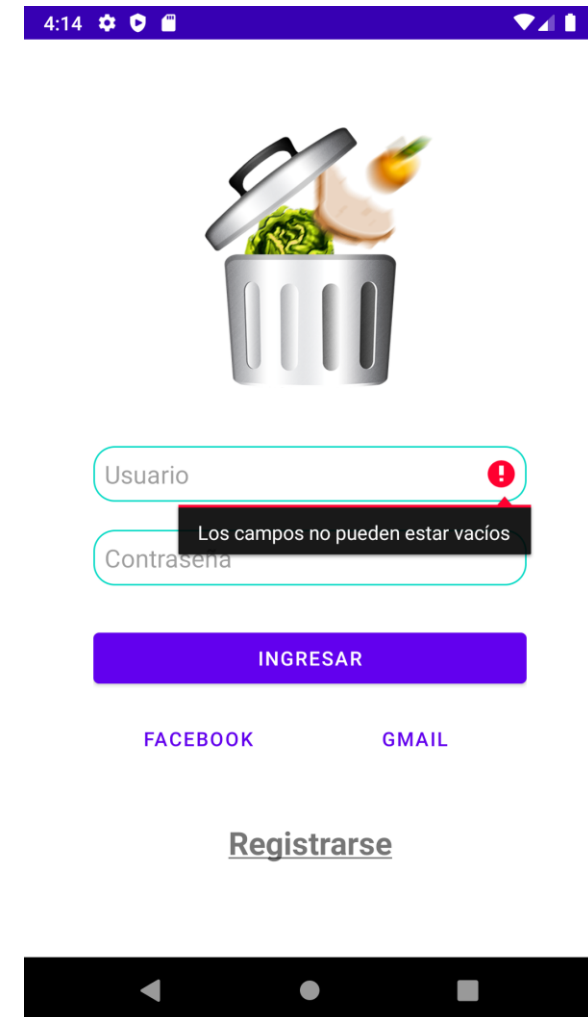
## Sprint backlog

### Historia de usuario HU-04

Como usuario requiero tener una interfaz amigable para poder introducir los datos requeridos por la aplicación

3. Al presionar el botón ingresar el sistema debe validar que no estén vacíos

- Se deberá validar que el usuario ingrese información en los campos de texto y mostrar un mensaje de error al momento de presionar el botón ingresar (**`campo.trim().isEmpty()`**)





## Sprint backlog

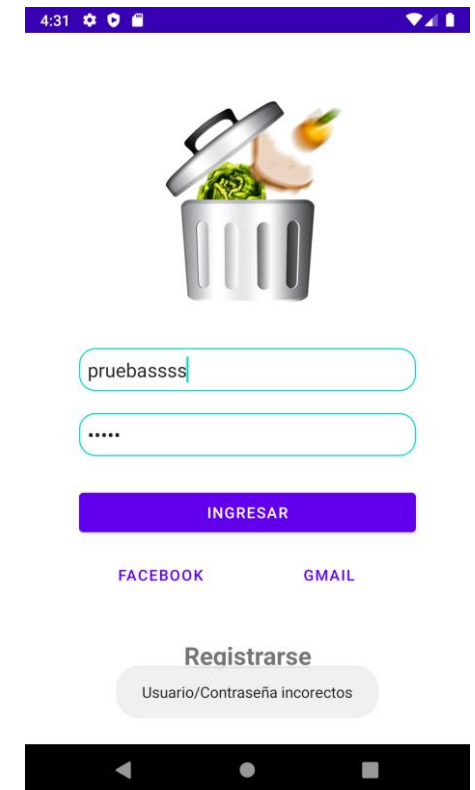
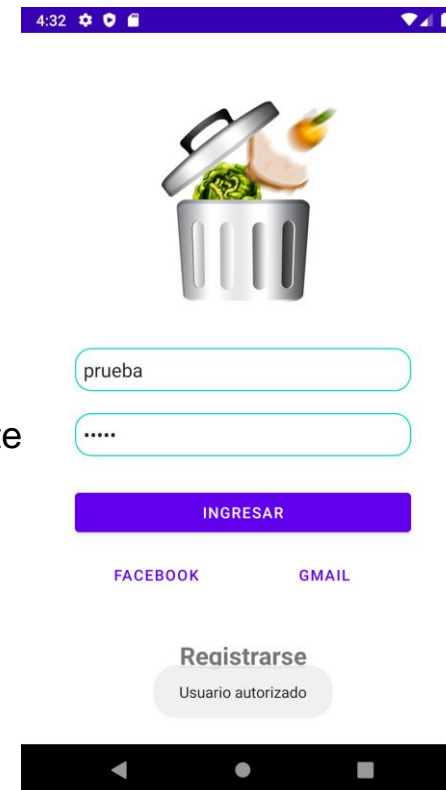
### Historia de usuario HU-04

Como usuario requiero tener una interfaz amigable para poder introducir los datos requeridos por la aplicación

1. El botón ingresar debe validar que el usuario exista y que los datos ingresados correspondan igual a los que se tiene previamente guardados

- Se deberá validar que los datos ingresados corresponda a los siguiente  
"prueba".equals(usuario)  
"12345".equals(password)

De lo contrario mostrar un mensaje por pantalla



## Entrega:

Suba a la plataforma los archivos del proyecto android con nombre ControladorLogin.java y LoginInterface.java.

Recuerda comentar las líneas de código asociadas al paquete y a las importaciones de otras clases.

Las funciones de validación deben retornar true si se cumplen todas las condiciones, de lo contrario deben retornar false.

```
//package com.upb.nodesperdicio.interfaz;  
  
public interface LoginInterface {  
  
    interface View {  
  
    }  
  
    interface Controlador {  
  
    }  
  
    interface Modelo {  
    }  
  
}
```

```
//package com.upb.nodesperdicio.controlador;  
  
//import com.upb.nodesperdicio.interfaz.LoginInterface;  
  
public class ControladorLogin implements LoginInterface.Controlador {  
  
    private final LoginInterface.View view;  
  
    public ControladorLogin(LoginInterface.View view) {  
        this.view = view;  
    }  
  
    @Override  
    public Boolean validarLogin(String editText, String indicador) {  
  
    }  
  
}
```



El futuro digital  
es de todos

MinTIC



## Material de apoyo:

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