

Desafío - Elementos de la interfaz, navegación e interacción

En este desafío validaremos nuestros conocimientos de fragmentos, navegación, enlazamiento de vistas usando view binding y utilización de listeners. Para lograrlo, necesitarás aplicar lo aprendido en clases y la guía de ejercicios “Elementos de la interfaz de navegación e interacción III”.

Lee todo el documento antes de comenzar el desarrollo **individual**, para asegurarte de tener el máximo de puntaje y enfocar bien los esfuerzos.

Descripción

Aplicando los conceptos y herramientas aprendidas hasta ahora, debes construir un Producto Mínimo Viable (MVP por sus siglas en inglés) para una app que valide la fortaleza de una contraseña.

La app consta de 2 pantallas:

1. Pantalla principal donde el usuario ingresa la contraseña:
Tiene un botón que está inicialmente desactivado y que se activa cuando la contraseña ingresada cumple con las siguientes condiciones:
 - a. Tiene más de 5 caracteres.
 - b. Tiene al menos una letra mayúscula.Cuándo se cumplen AMBAS condiciones, el botón se activa y al ser presionado, navega hacia la pantalla secundaria que muestra el resultado esperado.

<div>Validador de contraseñas</div> <div>Escribe una contraseña</div> <div>VALIDAR CONTRASEÑA</div> <div>La contraseña debe tener al menos 5 caracteres y una letra mayúscula</div>	<div>Validador de contraseñas</div> <div>Escribe una contraseña *****</div> <div>VALIDAR CONTRASEÑA</div> <div>La contraseña debe tener al menos 5 caracteres y una letra mayúscula</div>
---	---

Pantalla de ejemplo que cumple con los requisitos mínimos.

2. Pantalla de resultado donde se le muestra un mensaje al usuario, indicando que cumple con ambos requisitos y agrega consejos para mejorar la contraseña.

← Resultado

Recuerda que para que la contraseña sea segura debe cumplir con estas características:

> Al menos 5 caracteres

> Al menos 1 letra mayúscula

Otras características que hacen la contraseña más segura

> Al menos 1 carácter especial (como . , % \$)

> Al menos 1 número

Pantalla de ejemplo que cumple con los requisitos mínimos.

Requerimientos

Respetando los requerimientos mínimos detallados en la descripción, deberás cumplir con los siguientes requerimientos:

1. Una actividad que contenga un NavHostFragment para mostrar los fragmentos.
(1 Punto)
2. La pantalla principal está definida en un fragmento donde el usuario ingresa la contraseña. El diseño de la pantalla es libre, pero debe incluir un campo de texto para ingresar la contraseña, un botón deshabilitado que se habilita cuando se cumplen los criterios y un texto indicando las características que debe tener la contraseña.
(1 Punto)
3. La pantalla de resultado está definida en un fragmento donde se le muestra al usuario los criterios usados para validar su contraseña, y algunos consejos para mejorar su seguridad.
(1 Punto)
4. Utilizar ViewBinding para enlazar y utilizar las vistas.
(1 Punto)
5. Validación de la contraseña aplicando los 2 criterios:
 - a. Tiene más de 5 caracteres.
 - b. Tiene al menos una letra mayúscula.
(1 Punto)
6. Utilizar un ChangeListener para escuchar los cambios en la contraseña y activar el botón cuando se cumplen las condiciones, pero cuando las condiciones no se cumplen, el botón debe estar deshabilitado.
(1 Punto)
7. Utilizar onClickListener para reaccionar al click en el botón.
(1 Punto)
8. Utilizar Navigation para navegar entre fragmentos, donde debe estar definido el NavHostFragment y el grafo de navegación.
(1 Punto)
9. Navegar hacia la pantalla de resultado.
(1 Punto)
10. Extraer los textos al archivo de recursos de textos (strings.xml).

(1 Punto)



¡Mucho éxito!

Consideraciones y recomendaciones

- El proyecto debe ser comprimido en un archivo de formato .zip y subido a la plataforma. Tip: Antes de comprimir, elimina la carpeta build de la raíz del proyecto para disminuir el tamaño del archivo comprimido.
- La plantilla de proyecto BasicActivity viene con 2 fragmentos y con otras características configuradas y puede ser un buen punto de partida. Si quieres realizar la configuración tú mismo, puedes partir desde la plantilla de proyecto EmptyActivity.
- La clase [String](#) de Java tiene los métodos [toUpperCase\(\)](#) para convertir el texto a mayúsculas y [toLowerCase\(\)](#) para convertir el texto a minúsculas.