
Laboratorio 2: Worker threads, Internet connection,
elementos de UI, dialogs y View Binding
CURSO: SERVICIOS Y APLICACIONES PARA IOT [1TEL05]

SEMESTRE: 2025-2

Observaciones:

- El laboratorio se realizará de manera individual.
- El entregable será el enlace de GitHub: LAB2_[Código PUCP]
- La fecha de entrega máxima es el domingo 21/09 a las 10 am.

Recomendación: “Leer toda la evaluación antes de comenzar”

Ejercicio 1: Realizar el menú principal. (6 Ptos)

Se debe crear la vista principal de forma similar a la mostrada en el ejemplo, en la cual se debe mostrar 3 items como se observa, “Cantidad” donde deberá escribir la cantidad de imágenes a mostrar, “Texto” es donde se elegirá Si o No y en “Escribir texto” podrá ingresar cualquier palabra como crea conveniente, este item solo se activará si en “Texto” se eligió “Si”. Se tendrá también dos botones, uno que será para “Comprobar la conexión a internet” el cual al presionarlo deberá mostrar un ‘Success Toast’ si se tiene conexión a Internet o un ‘Error Toast’ si no se tiene internet, y el otro botón será “Comenzar” el cual se habilitará luego de comprobarse la conexión a internet.



The mockup shows a mobile app interface with a red header containing the text "¡Bienvenido a TeleCat!" and a cat emoji. Below the header, it says "La app diseñada para generar memes de gatos y divertirse :)". The main content area is yellow and contains three input fields: "Cantidad:" with a text input "Escribe aqui", "Texto:" with a dropdown menu showing "Elegir", and "Escribir texto:" with a text input "Escribe aqui". At the bottom, there are two buttons: a red button with a warning icon and the text "Comprobar Conexión", and a grey button with a play icon and the text "Comenzar".

Nota 1: Debe realizar las validaciones respectivas (Si se elige “Si” en “Texto” debe escribir algo en “Escribir texto” y la cantidad no debe estar vacía), si no se cumplen debe mostrar un toast.

Nota 2: Para los siguientes Ejercicios la conectividad por métodos GET debe realizarse previa validación de que existe la conectividad a Internet.

Ejercicio2: TeleCat (8 Ptos)

Redirigido del menú principal, se podrá visualizar una vista como la mostrada con las siguientes características:

- Se tendrá un contador que irá de forma descendente y tendrá un tiempo definido de acuerdo a la cantidad de imágenes que se mostró, cada imagen debe mostrarse 4s (Ejm: Si elijo 3 imágenes el contador debería disminuir desde 12s) Si se termina el tiempo se habilitará el botón “siguiente” para pasar a la siguiente vista (Vista de Finalización)
- El contador no debe parar así dentro de la aplicación se navegue entre vistas, se rote la pantalla y se cierre la pantalla.
- Hacer uso de hilos para su implementación, caso contrario se debe explicar su no uso.
- Se mostrará las imágenes como se muestra.

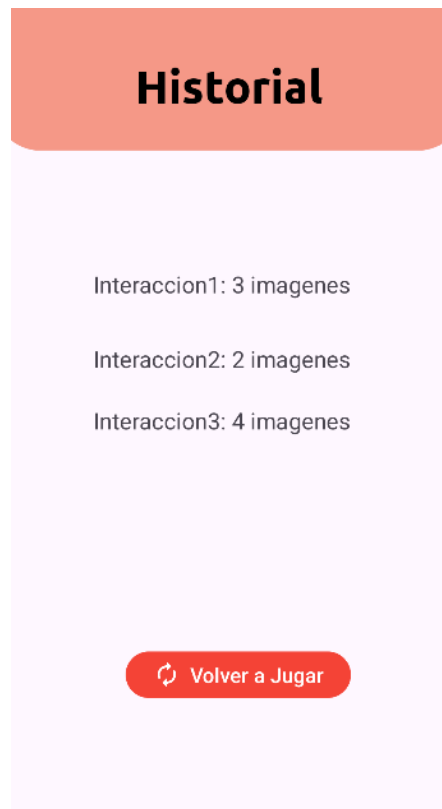


Obtención de las imágenes de la app: Se utilizará el método GET desde una API para obtener las preguntas, la API a utilizar será <https://cataas.com> y permite la obtención de las imágenes, se debe revisar la documentación de esta [acá](#) se incluye como incluir texto en las imágenes proporcionadas por la API.

Ejercicio 3: Pantalla de finalización (4 Ptos)

Redirigido del botón “siguiente” luego de visualizar todas las imágenes:

- Se mostrará un historial de cada interacción
- Se tendrá un botón “Volver a Jugar” que mostrará un dialog con la pregunta: ¿Esta seguro que desea volver a Jugar? Y elegir entre “Si” y “No”, si elige “Si” se redirigirá al menú principal y reiniciará el juego, si elige “No” se quedará en la misma pantalla.



Ejercicio 4: (2 Pto)

Se tomará en cuenta el diseño y la originalidad de la app, botones de navegación, menús o cualquier funcionalidad que mejore la interacción con la aplicación.

Además el uso de ayuda LLM deberá ser indicado en el código de la APP