

Examen de Desarrollo de Servicios de Comunicación en Red

Curso 2023-2024. Convocatoria ordinaria. Grupo 1

Avisos previos

Tiempo máximo: 2 horas

Debes usar el ordenador del laboratorio y puedes acceder a internet para resolver tus dudas mientras no haya interacción humana directa en el proceso. Es decir, puedes consultar cualquier fuente publicada (Stack Overflow, Foros, Blogs, búsquedas en Google, etc), pero no podrás hacer uso de chats, mensajería, ni herramientas de publicación con otras personas, sea o no en tiempo real.

Para presentar correctamente el examen, tienes que enviar el código que vas a realizar, sigue los pasos que hay en el apartado "Cómo entregar" que hay al final de este documento. Recuerda entregar el código, aunque no acabes toda la funcionalidad ya que cada paso cuenta. Tras subir el fichero podrás añadir los comentarios que quieras en la web.

Enunciado

Te han pedido que crees una nueva aplicación que muestre el listado final de las llegadas de una carrera. Para ello se pedirá al servidor los datos y los mostraremos en un listado.

La petición es de tipo GET a la dirección <https://api.battleship.tatai.es/v2/exam-2023-1>

Si la petición es correcta, el servidor retornará un código HTTP 200 y un mensaje en formato JSON cuyo contenido podría ser (se muestra en distintas líneas y con espacios por claridad):

```
{ runners: [
  { runner: { name: "Paco", "lastname": "Gómez" }, gender: "M", number: 1, time: 104 },
  { runner: { name: "Ana", "lastname": "Pérez" }, gender: "F", number: 2, time: 100 },
  { runner: { name: "Pedro", "lastname": "García" }, gender: "M", number: 3, time: 340 }
] }
```

Es decir, cada elemento en `runners` tiene cuatro propiedades, `runner` objeto con el nombre de la persona, `gender` con el sexo del corredor (string, M para hombre, F para mujer), `number` (entero con el número de dorsal), `time` con el tiempo en segundos.

Una vez tenemos los datos, vamos a proceder a mostrarlos en un `ListView` y para ello tienes que crear un adaptador propio (clase que extiende de `BaseAdapter`).

Cada fila tendrá 5 celdas:

1. Puesto ocupado en la general. Ranking único para todos los corredores. Mostrar centrado

2. Puesto ocupado en ranking de categoría (por sexo)
3. Nombre completo mostrar alineado a la izquierda
4. Número de dorsal, mostrar centrado
5. Tiempo empleado en la carrera, mostrado separando minutos y segundos, por ejemplo 109 segundos se mostrará como 1 m 49 s. Mostrar alineado a la derecha

Debe estar ordenado de forma descendente según la el tiempo empleado, es decir, primero el más rápido:

1	1	Ana Pérez	2	1 m 40 s
2	1	Paco Gómez	1	1 m 44 s
3	2	Pedro García	3	5 m 40 s

Al pulsar en cualquier fila, se abrirá una pantalla en la que se muestre la información desglosada y correctamente organizado:

Puesto en la general: 3

Puesto en la categoría: 2

Nombre completo: Pedro García

Dorsal: 3

Tiempo: 5 m 40 s

A tener en cuenta

Crea una nueva aplicación con estos datos (el valor de XX te lo indicará el profesor):

- Name: Examen 2023 XX
- Package name: es.unavarra.tlm.examen_23_XX

Deberás usar las técnicas y librerías mostradas en clase.

Recuerda configurar los permisos adecuados en tu aplicación.

Todos los textos deben estar como `string` en `values`.

Debes usar una librería para generar y procesar JSON. No debes generarlos ni leerlos a mano.

Puntuación

- Petición al servidor: 2 puntos
- Adaptador y visualización de información en listado: 5 puntos
- Pantalla de detalle: 3 puntos

Cómo entregar

Junto a este documento, en la web de la asignatura, hay un script llamado “preparar”. Descárgalo en tu ordenador, ve a la carpeta y ejecútalo:

```
$ ./preparar
```

Si el fichero no se ejecutase, prueba antes a ejecutar este comando:

```
$ chmod +x ./preparar
```

Sigue los pasos que te indica y al final generará un fichero comprimido que **debes subir usando la web de la asignatura**. Tras subir el fichero puedes añadir comentarios.