

Fundamentos de la Programación

José Javier Villena
@JoseCodFacilito

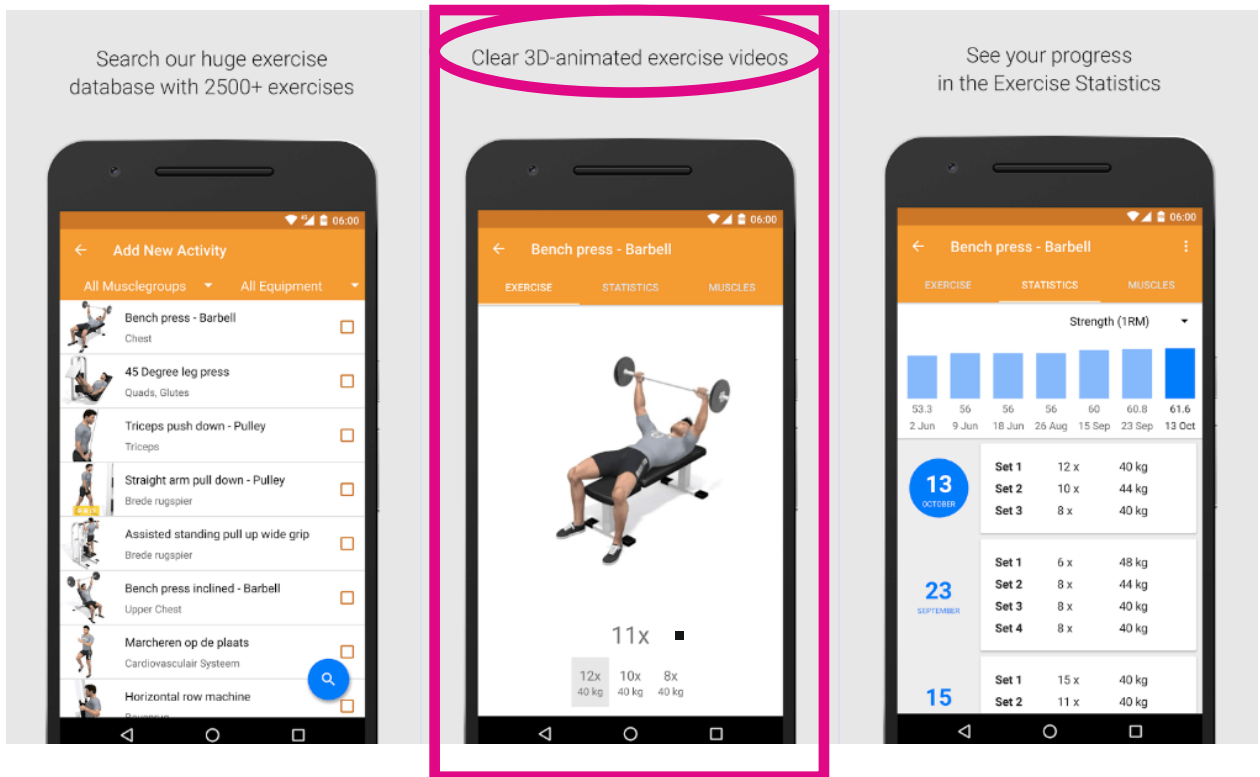
SECCIÓN 5

PROGRAMACIÓN

MODULAR

Ejercicios para funciones

La siguiente aplicación es usada en el mundo del fitness. Sirve para indicar al usuario ejercicios y tablas de musculación.



Como puedes ver en la segunda pantalla, el usuario puede recrear en video la simulación del ejercicio que debe realizar. Los ejercicios pueden tener varias series, cada una con distintos números de repeticiones y con diferentes segundos de descanso.

De modo que la app debe ser capaz de reproducir un video con el ejercicio muscular X veces hasta completar la serie (el video solo reproduce UNA vez el movimiento, si ese movimiento se debe hacer 12 veces, entonces el video se debe reproducir 12 veces) y reproducir otro gif X segundos de descanso en cada serie, excepto la última en la que el ejercicio queda completado. Para este caso hipotético asumiremos que los movimientos están recreados un video y la recreación del descanso en un gif.

Ej.

1ª serie: 12 repeticiones + 45 segundos de descanso

2ª serie: 10 repeticiones + 30 segundos de descanso

3ª serie: 8 repeticiones

ADVERTENCIA: en este ejercicio nos vamos a centrar ÚNICAMENTE en como dividir y modular el código de forma reutilizable y optimizable. No tanto en comandos más específicos de cada tecnología como los de reproducción de video, sino en la manera de crear funciones que nos ayuden a dividir el código para administrarlo mejor

Vamos a recrear el código que podría servir para generar ese servicio en la app. Lo haremos en pseudocódigo, así cada uno podrá trasladarlo al lenguaje que prefiera.

1- Necesitamos una función a la que llamar o invocar para que reproduzca el video con el movimiento muscular a realizar. A esa función le diremos exactamente qué ejercicio reproducir, ya que esa función será la misma tanto para abdominales, pecho, pierna, etc.

**funcion reproducirEjercicio (nombreEjercicio)
comando_para_reproducir_archivo (nombreEjercicio)**

Esta función reproducirá el ejercicio solo una vez. ¿Cómo haremos para recrear la serie completa?

2- Crearemos otra función en la que administraremos cada serie, indicándole el número de veces que se tiene que realizar el movimiento y el número de segundos de descanso que corresponde a tal serie.

Sin embargo, puesto que los segundos a descansar no serán necesario siempre, sino tan sólo tras la primera y segunda serie, tal parámetro será opcional, y su valor por defecto será 0, para que no se reproduzca ningún gif de descanso tras la última serie

Desde esta función se llamará tantas veces como sea necesaria a la función creada en el paso anterior

**funcion serie (nombreEjercicio, numMovimientos, opcional segDescanso = 0)
for i=1 to numeroMovimientos
reproducirEjercicios (nombreEjercicio)**

3- Ahora necesitamos otra función a la que llamar para reproducir el gif del personaje descansando tras cada serie

¿Creamos otra función nueva? ¿O podríamos reutilizar la que reproduce los movimientos del ejercicio para también reproducir los descansos? Al fin y al cabo...se trata de archivos en ambos casos. De modo que podemos hacerlo en una misma función. A esto se le llama reutilizar código para optimizar

primero hicimos esta función:

```
funcion reproducirEjercicio (nombreEjercicio)  
    comando_para_reproducir_archivo (nombreEjercicio)
```

ahora vamos a cambiarle el nombre para reutilizar también en la reproducción de descansos

```
funcion reproducirArchivo (nombreArchivo, optional Segundos = 0)  
    if segundos > 0  
        comando_para_reproducir_archivo (nombreArchivo, Segundos)  
    else  
        comando_para_reproducir_archivo (nombreArchivo)
```

De modo que esta función ahora puede reproducir los videos y también los descansos tantos segundos como le hayamos indicado.

Así que la función serie quedará adaptada a estos cambios de la siguiente manera

```
funcion serie (nombreEjercicio, numMovimientos, opcional segDescanso = 0)  
    for i=1 to numeroMovimientos  
        reproducirArchivo (nombreEjercicio)  
  
    if segDescanso > 0  
        reproducirArchivo ("descanso.gif", segDescanso)
```

4- Bien, ya tenemos la manera de reproducir cada serie el número de veces que se precise y los descansos tantos segundos como requiera cada serie. Ahora falta la manera de poder administrar el ejercicio completo indicando cuántas series tendrá y el número de repeticiones y descanso que tiene cada uno de ellas

Para ellos nos crearemos una función desde donde invoquemos cada serie indicando el ejercicio, las repeticiones y los descansos

```
funcion ejercicio (nombreEjercicio, rep1, des1, rep2, des2, rep3)  
    serie (nombreEjercicio, rep1, des1)  
    serie (nombreEjercicio, rep2, des2)  
    serie (nombreEjercicio, rep3)
```

NOTA: tales datos podrían estar establecidos de antemano o dejarle al usuario que los personalice, pero recuerda que el objetivo de este ejercicio es simplemente aprender a dividir el código en funciones

De modo que el código total sería el siguiente:

```
funcion ejercicio (nombreEjercicio, rep1, des1, rep2, des2, rep3)  
    serie (nombreEjercicio, rep1, des1)  
    serie (nombreEjercicio, rep2, des2)  
    serie (nombreEjercicio, rep3)
```

```
funcion serie (nombreEjercicio, numMovimientos, opcional segDescanso = 0)  
    for i=1 to numeroMovimientos  
        reproducirArchivo (nombreEjercicio)
```

```
    if segDescanso > 0  
        reproducirArchivo ("descanso.gif", segDescanso)
```

```
funcion reproducirArchivo (nombreArchivo, optional Segundos = 0)  
    if segundos > 0  
        comando_para_reproducir_archivo (nombreArchivo, Segundos)  
    else  
        comando_para_reproducir_archivo (nombreArchivo)
```

5- Para hacerlo funcionar sólo haría falta invocar a la función ejercicio.

Ej:

```
ejercicio ("abdominales", 30, 45, 20, 30, 10)  
ejercicio ("pecho", 15, 60, 12, 30, 10)
```

Conclusiones:

Como has podido ver, dividir los diferentes objetivos a cumplir en funciones específicas que se centren en dichas tareas nos ayuda a separar cada parte del programa y atacar cada objetivo de forma concreta

Después solo hay que combinar tales funciones hasta crear el panorama completo. Pero para ello ha sido necesario determinar las funciones y sus parámetros de forma que permitan ser usadas de diferentes maneras y así reutilizar esas piezas de código.



Fundamentos de Programación - José Javier Villena