

Proyecto final de curso

Android: Fundamentos de Programación

(septiembre-diciembre 2016)

Nombre de la aplicación: Educa Invaders

Autor: Asensio Llobat, Enrique

Qué hace la aplicación:

Versión para Android del juego clásico Space Invaders. La dinámica del juego consiste en un grupo de enemigos alienígenas que van disparando y desplazándose de lado a lado y descendiendo cada vez que llegan al lateral. Al mismo tiempo en la parte inferior se encuentra la nave que debe ir destruyendo a los enemigos.

El juego termina en el momento en que la nave es alcanzada por uno de los misiles enemigos o si estos alcanzan la parte inferior de la pantalla.

Licencia:

Autorizo la difusión del código fuente con fines educativos siempre que se haga referencia al autor bajo los términos generales de la licencia "[Academic Free License v.3.0](#)".

A destacar:

- El control de la nave se realiza con mediante la pantalla táctil. Además se puede hacer uso de los sensores del móvil para controlar el desplazamiento lateral de la nave. El desplazamiento es siempre a velocidad fija.
- Las puntuaciones se pueden almacenar en un fichero de la memoria interna o en una base de datos ubicada en un servidor externo. Para esto se hace uso de un *servicio Web*.
- A la hora de guardar la puntuación de la partida, puede almacenarse la localización geográfica donde se jugó dicha partida. Se obtiene la longitud y latitud, junto con el nombre del país y la zona geográfica obtenida de un servicio web de la página www.geonames.org.
- Se hace uso de la librería org.json para leer los datos obtenidos en formato JSON de la página www.geonames.org.
- Desde el menú de inicio se puede acceder a puntuaciones estadísticas de los países o regiones donde se ha jugado. Para esto se hace uso de FragmentTabHosts y vistas de tipo RecyclerView. De este modo se puede ver la puntuación acumulada por países o regiones y si se pulsa en cada uno de estos se mostrará un mensaje con las máximas puntuaciones de la zona seleccionada. Para esto se hace uso de un *Servicio Web* que realiza los cálculos necesarios, devolviendo la información necesaria.

- Si se selecciona la opción de almacenar información con localizaciones se solicita el permiso adecuado y se ejecuta un manejador que trata de obtener una localización con un nivel de precisión razonable. Una vez lo consigue para de pedir localizaciones. De este modo se ahorra batería, ya que no es importante ir realizando localizaciones continuamente.
- Se pasa información entre diferentes actividades para poder mostrar los datos correctos en la página de puntuaciones.
- Se hace uso del método AsyncTask para las solicitudes de información. De todas formas se ejecuta de forma síncrona para mostrar las puntuaciones.
- La vista inicial ha sido rediseñada para verse correctamente en vertical y horizontal.
- Idiomas: español e inglés.
- Se realizan distintas animaciones para visualizar los enemigos, así como los misiles y la destrucción de las defensas.

Motivaciones:

El objetivo de este proyecto era tratar de afianzar los conocimientos de la asignatura *Fundamentos de Programación de Android*. Para ello, a nivel de desarrollo, se ha tratado de realizar un código comprensible y documentado.

Se ha optado por el juego *Space Invaders* ya que la dinámica era bastante acorde a lo realizado en el curso.

Se ha tratado de introducir, en la medida de lo posible, alguna tarea que hiciese uso en la medida de lo posible de todo lo aprendido en el curso. Para ello se han realizado tareas como la conexión con distintos servicios web o la obtención de la localización del dispositivo y la solicitud y gestión de los permisos. Estas funcionalidades no son muy lógicas para este tipo de juegos, pero sí para afianzar los conocimientos del curso

Cómo lo hace:

Debemos diferenciar 3 flujos de trabajo en la aplicación. Estos son el empleado para controlar la dinámica del juego, el empleado para realizar el control de localizaciones y el empleado para controlar la gestión de las puntuaciones y estadísticas. A continuación se comenta cada uno de estos flujos de la aplicación.

1) Control de la dinámica del juego

- El peso principal del programa recae en la clase VistaJuego, un descendiente de View. Desde el hilo principal se representan los gráficos (enemigos, nave, defensas y misiles) y se atiende las entradas de usuario.
- Aquí se crea un hilo secundario para llamar al método actualizaFisica() responsable de desplazar los objetos y verificar las colisiones. Para conseguir un desplazamiento uniforme se tiene en cuenta el tiempo real del sistema.
- Además se crean diferentes clases que controlarán la estructura de los integrantes del juego. Estas son:
 - *DinamicaEnemigos*: Controla los movimientos y funcionalidades del conjunto de enemigos.
 - *Enemigo*: Controla la estructura de cada enemigo.(Extiende de Drawable)
 - *Nave*: Controla la estructura de la nave.(Extiende de Drawable)

- *Defensa*: Controla la estructura de las defensas

2) **Control de localizaciones**

- El peso de esta tarea se realiza en la clase **ManejadorLocalizaciones**. Esta clase extiende de **LocationListener**. De este modo pueden registrarse las distintas localizaciones y una vez se obtiene una con un nivel de precisión razonable se lanza un servicio web de www.geonames.org, obteniendo información sobre el país y región de la partida.
- Al obtenerse la información se paran los servicios de localización y se devuelve la información a la clase principal **MainActivity**. De este modo cuando vaya a guardarse la partida, la localización ya se habrá realizado, sin tener que demorarse en este apartado.

3) **Gestión de las puntuaciones y estadísticas**

- Para mostrar la información se hace uso de **FragmentTabHost** y **RecyclerView**, a nivel visual.
- Para obtener los datos de cada vista se hace uso de un servicio web que obtiene los distintos datos estadísticos, como son las puntuaciones por países o regiones, junto con el jugador de mayor puntuación en cada caso. Y por otro lado se hace uso del mismo servicio para obtener la información de las máximas puntuaciones en cada caso, siendo posible mostrarla al seleccionarse una región concreta.

Lista de actividades sin alcance en la versión actual (Lista de mejoras o TODO List):

- Controlar la dinámica de la partida, configurando distintas configuraciones enemigas o pantallas de jefe final.
- Controlar las vidas de la nave.
- Añadir una pantalla para establecer configuraciones de la estructura enemiga. Se ha creado una estructura de posiciones de los enemigos en la matriz de juego, pero queda fuera de alcance en esta versión.
- Realizar un servicio que se ejecute puntualmente para actualizar las puntuaciones remotas desde los datos de puntuaciones locales.
- Realizar un servicio que actualice las zonas geográficas en función de la localización almacenada en la base de datos. Esto servirá para los casos en que falle el servicio de www.geonames.org al no usarse una licencia comercial.

Captura de pantallas:

