



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

Diseño e implementación de una APP para juego  
colaborativo. Desarrollo del Front-End

Trabajo Fin de Grado

**Grado en Ingeniería Informática**

**Autor:** Guillermo Domingo Zaragoza

**Tutor:** Patricio Orlando Letelier Torres

Curso 2015/2016



# Resumen

---

Este Proyecto nace por una inquietud de emprendimiento de dos estudiantes que finalizábamos la carrera. Buscábamos un reto que nos resultara interesante, además de poder conseguir un beneficio por nuestro trabajo realizado.

Decidimos ubicarlo en el ámbito de las aplicaciones móviles, concretamente en el de los juegos. Creímos que este ámbito está en un continuo crecimiento de usuarios y que podríamos sacar adelante un buen proyecto.

Se ha desarrollado en el espacio emprendedor de la ETSINF, llamado START.Inf.

# Tabla de contenidos

---

## Contenido

1. Introducción.....	7
1.1 Objetivos del proyecto .....	8
1.2 Organización .....	8
2. Motivación .....	9
2.1 Juegos en Apps móviles .....	9
2.2 Juegos con multijugador.....	11
2.3 Juegos de cartas .....	11
3. Juego elegido: el Truc.....	12
3.1 Un poco de historia .....	12
3.2 Breve explicación del juego.....	12
4. Estado del arte .....	14
4.1 Otras Apps similares .....	14
4.1.1 Truco Free .....	14
4.1.2 Truco Argentino.....	16
4.1.3 Truco Argentino Free.....	17
4.2 ventaja sobre los competidores.....	18
5. Estudio de la idea.....	21
5.1 Lean Canvas.....	22
5.2 Análisis DAFO.....	23
5.3 Análisis económico .....	24
6. Proceso de desarrollo .....	26
6.1 Mapa de características .....	27
7. Tecnologías utilizadas.....	28
7.1 Android.....	28
7.1.1 Dispositivos móviles y sus características.....	28
7.1.2 ¿Qué es Android? .....	28
7.1.3 El problema de la fragmentación.....	29
7.1.4 Cómo es android .....	29
7.1.5 Kernel linux.....	30
7.1.6 Librerías Android.....	30
7.1.7 Runtime de Android .....	30

7.1.8 Entorno de aplicaciones .....	31
7.1.9 Nivel aplicativo .....	31
7.1.10 Inicialización del Sistema .....	31
7.1.11 Aplicaciones .....	32
7.1.12 Ciclo de vida .....	32
7.1.13 Los procesos en Android .....	34
7.1.14 Tipos de procesos .....	34
7.1.15 Intents .....	35
7.2 Android Studio.....	36
7.2.1 Proyectos y estructura de archivos .....	36
7.2.2 Android build system .....	38
7.2.3 Depuración y rendimiento .....	38
7.2.4 Android Virtual Device (AVD) manager .....	39
7.2.5 Definiendo layouts.....	39
7.2.6 Android debug monitor.....	40
7.3 Genymotion + virtualbox.....	41
7.4 GitHub.....	42
7.5 AdMob.....	43
7.5.1 ¿Por qué admob? .....	43
7.6 Developer Console .....	44
8 Parte Front-End .....	45
8.1 Introducción.....	45
8.2 Pantallas de la aplicación.....	46
8.2.1 Pantalla de cargando .....	46
8.2.2 Pantalla de log-in .....	47
8.2.3 Pantalla Menu principal.....	47
8.2.4 Pantalla de espera de los jugadores .....	49
8.2.5 Pantalla de uno contra uno .....	49
8.2.6 Pantalla de dos contra dos .....	52
8.2.7 Pantalla de partida finalizada .....	53
8.2.8 Pantalla para invitar .....	54
8.2.9 Pantalla de invitaciones .....	55
8.2.10 Pantalla de Ranking .....	56
8.2.11 Pantalla de logros.....	56
8.3 Diagrama de flujo .....	57
8.4 Librerías.....	58



8.4.1 Material dialogs .....	58
8.4.2 Floating action button toolbar .....	59
8.4.3 Circular progressbutton .....	59
8.4.4 FButton .....	60
8.4.5 DialogPlus .....	60
8.4.6 Waiting dots .....	61
8.4.7 Tour Guide .....	61
9. Publicación en play store.....	63
9.1 ¿Qué es play store? .....	63
9.2 Darse de alta como desarrollador .....	63
9.3 Cómo publicar una aplicación .....	64
9.4 Precio de la publicación .....	69
10. Resultados obtenidos.....	70
10.1 Descargas.....	70
10.2 Usuarios de la App .....	71
10.3 Opiniones de los jugadores en Play Store .....	72
11 Trabajo futuro.....	76
11.1 Tareas pendientes .....	76
11.2 Posibles mejoras .....	77
11.3 Expansión a otros países .....	77
12. Conclusiones .....	78
12.1 Resultados del proyecto.....	78
12.2 Aprendizaje .....	78
12.3 Opinión personal.....	79
13 Bibliografía.....	80
14. Índice de ilustraciones .....	82
15. Índice de Tablas .....	84

## 1. Introducción



**Ilustración 1.** Logo del juego.

Para este proyecto, se ha decidido implementar una aplicación para dispositivos Android que permita jugar al popular juego de cartas valenciano: El Truc.

Tal y como puede apreciarse en la **Ilustración 1**, el título del juego es: “Retruque!”.

El motivo de la elección es que se buscaba algún título que llamara la atención, que no fuera simple y que pudiera resultar atractivo para el jugador. En el Truc hay una acción llamada Retrucar que, conjugada en primera persona y traducida al valenciano, adquiere el nombre de “Retruque!”.

Hoy en día, la mayor parte de los dispositivos móviles usan el sistema operativo Android, por lo que decidimos elegirlo para llevar a cabo nuestra aplicación, ya que adaptarnos a este sistema nos iba a resultar mucho más sencillo que a cualquier otro.

Para desarrollar en Android se usa el lenguaje de programación Java, el cual nos han enseñado en gran parte durante el grado que hemos cursado y, por tanto, teníamos una gran base para empezar a trabajar.

Por último, cabe citar que este proyecto ha sido desarrollado junto a Carlos Alba Ponce, que se encargará, en mayor medida, de la parte del Back-End.

Así pues, en éste se profundizó más en el Front-End de la aplicación.

## 1.1 Objetivos del proyecto

Este proyecto era un reto muy interesante para nosotros, al que sabíamos que tendríamos que dedicar un gran esfuerzo y muchas horas de trabajo, pero que a la vez nos hacía mucha ilusión sacar adelante.

Los objetivos a cumplir se describen a continuación:

- El objetivo básico era tener la App publicada en Google Play, para que cualquier persona pudiera descargarla y divertirse jugando un par de partidas con el trabajo que habíamos realizado.
- El segundo objetivo es, obviamente, poder obtener una remuneración económica. Somos conscientes de que es difícil llegar a expandir una App sin ser conocidos ni haber hecho otros trabajos anteriores, pero siempre creímos que si hacíamos un juego divertido podría llegar a expandirse y triunfar, aunque fuera en la Comunidad Valenciana (dónde es más típico el juego).

## 1.2 Organización

Al ser dos personas, era muy importante planificar bien las tareas del proyecto y repartirlas equitativamente. Había que organizarse bien e ir al día con el trabajo del compañero, para ir avanzando paralelamente hacia los distintos objetivos del proyecto.

Con el objetivo de seguir esta organización, decidimos marcarnos un horario y trabajar juntos en un mismo sitio, de tal forma que pudiéramos comunicarnos y ayudarnos mutuamente de manera fácil y rápida.

Una cosa que nos fue muy útil es tener planificado el proyecto con bastante antelación. Es decir, cuando empezamos a desarrollar el juego, ya habíamos estructurado previamente y en acuerdo junto con nuestro profesor los que serían nuestros objetivos principales y las características que iba a tener nuestro juego.

Las tecnologías que hemos utilizado en el proyecto nos han facilitado mucho la necesidad de trabajar en paralelo, pero ese es un punto que trataremos en puntos posteriores de esta memoria.

## 2. Motivación

El Truc es un juego típico de Valencia y pensamos que había llegado el momento de digitalizarlo y hacer que cualquier persona pudiera divertirse jugando una partida desde cualquier lugar y en cualquier momento.

Montar una partida de Truc no es tan sencillo, ya que lo más típico es juntar a cuatro jugadores y si a eso se le añade el tener que quedar en algún sitio físico, hace más complicado el poder jugar cuando realmente te apetece. Con nuestra aplicación, este problema se soluciona de manera rápida, ya que es posible jugar con tus amigos o con jugadores aleatorios que estén conectados. Siempre va a haber alguien listo para disfrutar de una partida de Truc junto a ti.

Por otra parte, tras analizar el mercado de aplicaciones, nos dimos cuenta de que nadie había creado una aplicación como la que habíamos pensado: no existía una aplicación para jugar al Truc con las reglas de Valencia. Vimos una gran oportunidad de negocio al comprobar esto.

Sí que existían, sin embargo, otras aplicaciones para jugar con otras reglas, ya que también es un juego al que se juega mucho en Sudamérica y que es llamado Truco.

Además, toda la gente que escuchaba nuestra idea mostraba un gran interés y decía que le gustaría probar nuestra aplicación, por lo que esto nos dio una motivación extra: darnos a conocer entre la gente y hacer que se divirtieran con algo que habíamos creado nosotros mismos.

Como extra, y con el objetivo de seguir fomentando el origen del juego, decidimos traducirlo completamente al valenciano, por lo que si un jugador tiene configurado su dispositivo móvil en catalán (Android no tiene el idioma valenciano, sólo el catalán) podrá disfrutar del juego en su idioma original.

### 2.1 Juegos en Apps móviles

Una de las cosas que más triunfan hoy en día en las tecnologías móviles son las aplicaciones de juegos. ¿Quién no ha jugado nunca a un juego que tiene descargado en el móvil en algún momento de aburrimiento? Ya sea de camino al trabajo mientras vas en el metro o en un momento de relax en el sofá de casa. Siempre resulta divertido tener un par de juegos instalados en tu dispositivo para despejarse un poco en un momento determinado. Ahí es donde entra en escena el éxito de las aplicaciones de juegos móviles.

Generalmente, un usuario juega una partida de vez en cuando, durante un periodo de tiempo medianamente corto, por lo que existe un gran reto para el desarrollador: ganarse al jugador en un periodo de tiempo muy corto, puede que incluso sea necesario ganarlo en la primera partida. De lo contrario, si el jugador no se ve sorprendido puede decidir desinstalar la App y buscar otra que le divierta más.

Sea como sea, hay una cosa clara: las App de juegos móviles son muy populares en la actualidad y son una gran manera de poder ganar dinero y de darse a conocer entre los jugadores. Una gran App puede servir para que un jugador se baje tu próxima App, para ver si le gusta tanto como la anterior. Es importante desarrollar un producto de calidad.

Si miramos un par de años atrás, no debemos hacer mucho esfuerzo para recordar algunas Apps de juegos como Candy Crash, Andry Birds o Flappy Bird. Apps que consiguieron millones y millones de descargas y que provocan que los jugadores descarguen los nuevos juegos de los desarrolladores.

Es obvio que las anteriores son juegos mundialmente conocidos, y que es prácticamente imposible diseñar un juego que llegue a ese nivel de éxito pero, ¿por qué no? Todo es intentarlo.

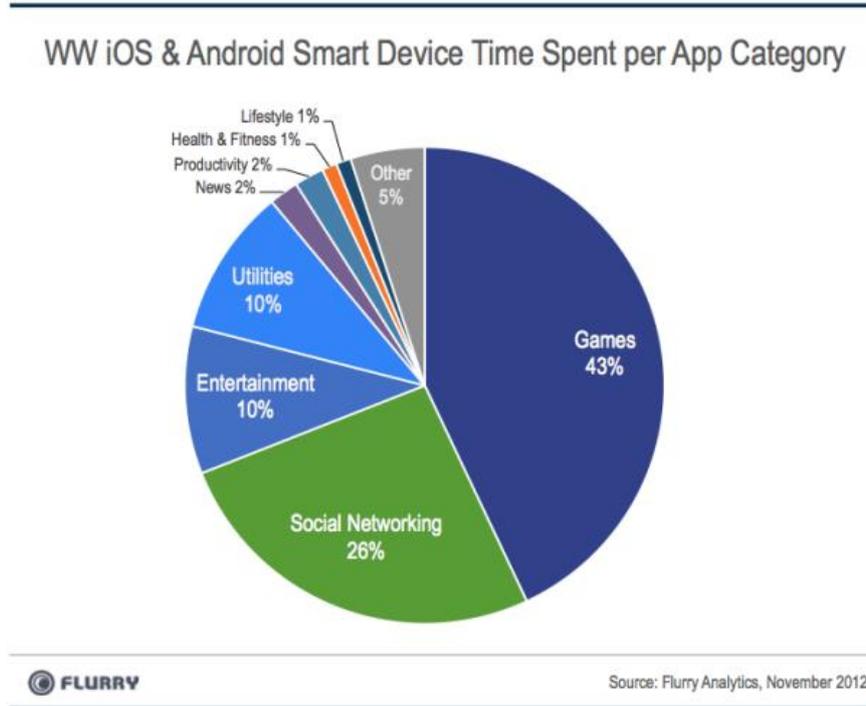


Ilustración 2. Gráfico de Apps más descargadas

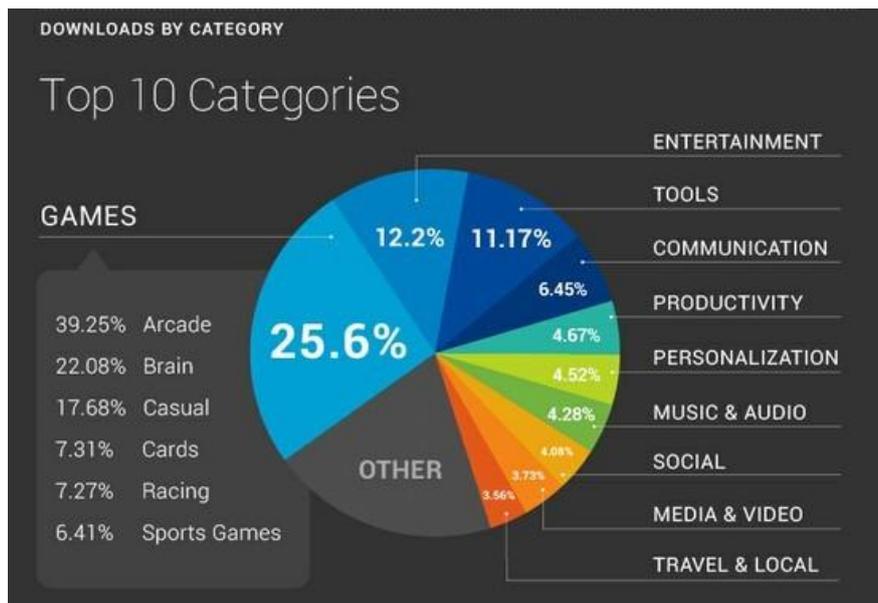


Ilustración 3. Gráfico de Apps más descargadas en Android

Como podemos ver en los dos anteriores gráficos, los juegos son las Apps más descargadas entre todos los tipos disponibles. En el primero, los datos son de todas las plataformas y, en el segundo, de dispositivos Android. Véase la **Ilustración 2 y 3**.

Además, de entre el 25,6% de descargas de juegos, el 7,31% son juegos de cartas, porcentaje mayor incluso que otras categorías importantes como música, redes sociales o comunicación.

## **2.2 Juegos con multijugador**

Expandiendo un poco más el apartado anterior, entramos en los juegos que incluyen un modo multijugador. Si un juego resulta divertido de por sí, ¿qué pasa si incluye un modo multijugador y te permite disfrutar de una partida con tus amigos o con quien quieras, estés donde estés?

La diversión del juego aumenta notablemente, y es mucho más fácil que algún jugador desafíe a sus conocidos y les incite a descargarlo. Por decirlo de alguna manera, es una forma de propiciar la expansión del juego y atraer a un número mayor de jugadores.

Gracias a las conexiones de datos de las que disponen prácticamente la totalidad de los dispositivos móviles de hoy en día, un usuario puede jugar una partida de un juego contra un amigo que viva en la otra parte del mundo con un par de acciones y en tan sólo unos segundos. ¿Y si además se le permite al jugador buscar partidas contra jugadores aleatorios que estén conectados al juego? Siempre habrá alguien dispuesto a echar una partida. Resulta muchísimo más divertido que jugar una partida contra la máquina.

Podemos concluir tras esta breve explicación que la inclusión de un modo multijugador en un juego para dispositivos móviles es totalmente positivo, pudiendo aumentar notablemente el número de usuarios que decidan probarlo, la expansión de la App y como no, la diversión.

## **2.3 Juegos de cartas**

Para concluir este apartado de la memoria hablaremos sobre una modalidad específica de los juegos móviles: los juegos de cartas.

Jugar a las cartas siempre ha sido muy popular para pasar los ratos muertos. Una baraja, una mesa y un par de amigos bastaban para tener un buen rato de diversión. Ahora, gracias a las nuevas tecnologías, podemos disfrutar de la mayoría de los juegos de cartas desde cualquier sitio, simplemente instalando una App en nuestro dispositivo móvil.

Nunca había sido tan fácil desafiar a tus amigos a una partida de Brisca, Cinquillo, etc.

Tras investigar detenidamente el mercado de aplicaciones nos dimos cuenta de que faltaba uno de los más populares, sobretodo en la Comunidad Valenciana: El Truc. Como citamos en puntos anteriores de la memoria, esta fue una de las causas que nos ayudó a decidirnos por este juego.

Con el fin de profundizar un poco más acerca de El Truc, veamos el próximo apartado, en el cual nos dedicaremos a conocer mejor nuestro juego de cartas elegido.

### 3. Juego elegido: el Truc

En este apartado de la memoria conoceremos un poco más nuestro juego elegido. Para meternos en contexto, hablaremos un poco sobre su historia, sus reglas (incluyendo las de nuestra App) y por último, profundizaremos un poco más en el motivo de nuestra elección.

#### 3.1 Un poco de historia

El juego del truco o truque, denominado “el truc” en valenciano, remonta sus orígenes al mundo árabe o musulmán, adquiriendo no obstante en la Comunidad Valenciana una enorme importancia, al estar arraigado en todas sus comarcas y poblaciones, hasta el punto de ser, posiblemente, el juego de cartas del que más uso hacen los valencianos. Tras cada celebración o comida entre amigos, en bares o en “casals” falleros, resulta casi obligado acabar con una buena partida de “truc”. Las habilidades, la forma y estrategias personales de juego se transmiten oralmente de padres a hijos, de abuelos a nietos o entre amigos. Se trata siempre de ser el mejor jugador con una estrategia de juego bien definida, en donde, además de la suerte propia de los juegos de azar y de una buena técnica en cada jugada, tiene especial importancia el uso de la mentira para generar confusión o llevar a engaño al contrario, obteniendo a cambio la correspondiente ventaja.

#### 3.2 Breve explicación del juego

Si bien existen diversas versiones del juego del “truc” en su versión valenciana, en esta aplicación hemos optado por el uso más extendido del mismo, en base a veintidós cartas, despreciando los doses, ochos, nueves, sotas, caballos y reyes de cada uno de los cuatro palos de la baraja española (Oros, copas, bastos y espadas), así como el as de oros y el de copas. La aplicación, en definitiva, se estructura en base al uso de las siguientes cartas:

Oros: Del tres al siete de oros.

Copas: Del tres a siete de copas.

Bastos: Del tres al siete de bastos y el as de bastos.

Espadas: Del tres al siete de espadas y el as de espadas.

Siendo las cartas de mayor valor en este juego, por su orden: el as de espadas, el as de bastos, el siete de espadas y el siete de oros (conocidas éstas dos cartas como “manillas”), siguiéndoles en cuanto a valor: los cuatro treses de cada uno de los palos y a continuación cada una de las cartas por su valor numérico.

Pero en el juego que examinamos no sólo se hace uso de las cartas, sino que son importantes además los gestos y la palabra, dado que cada una de las cartas importantes dispone de nombre y seña propia, siendo fundamental la comunicación entre jugadores y el traslado de las señas al compañero sin que puedan ser captadas por el rival. Bien para darle a la pareja una información de las cartas que se poseen, a efectos de efectuar una correcta estrategia de juego, o bien, para confundir al contrario, realizando señas de cartas que realmente no se poseen, induciendo con ello a confusión al mismo, a efectos de obtener, en uno y otro caso, la correspondiente ventaja en el juego. Hemos trasladado a la aplicación no sólo las cartas con sus reglas de juego, sino que además, hemos incorporado, los correspondiente iconos de señas identificativos de las cartas importantes del “truc”, tratando de hacer la versión virtual lo más próxima posible a la real, para que todo jugador avisado pueda obtener la correspondiente ventaja sobre su contrario. Incluso en la aplicación hemos introducido un icono que permite cazar y descubrir las señas del contrario, si bien es cierto que para ello hay que tener una especial habilidad y rapidez.

A su vez se han incorporado a la aplicación de las dos formas de jugarse los puntos (también denominados piedras) denominadas “envit” o envite, con sus correspondientes alternativas: “torne el envit” o devuelvo el envite, “envide fins igualar” o envido hasta igualar y la tan temida falta. Y el “truque” o truco, con sus alternativas de “retruque” o retruco, “quatre val” o vale cuatro, y “joc fora” o juego fuera. Y todo ello con la correspondiente valoración en puntos o piedras de cada una de las jugadas y su resultado para el ganador, cuya aplicación es automática al final de cada ronda hasta completar “la cama” o juego parcial, distribuida en dos fases, la de “a males” y “a bones”, esto es, a malas y a buenas, compuesta cada una de doce puntos o piedras. El jugador o jugadores que primero completan dos camas (una sólo en nuestra App) son los ganadores de la partida.

Cabe citar que el Truc también es muy popular en otros países, sobretodo en Sudamérica, donde tienen distintas reglas según el país (se juega con toda la baraja, hay otras modalidades de jugarse los puntos, etc.) y es conocido como Truco.



## 4. Estado del arte

En este punto 4, vamos a hacer un estudio de mercado de distintas Apps de juegos de cartas, así como de las características de nuestra aplicación, y seguidamente haremos una comparación para ver los puntos a favor y en contra que tiene nuestro juego respecto a los competidores.

Como ya se ha dicho anteriormente, no hay ninguna App que permita jugar al Truco como la nuestra, pero sí que existen otros juegos para jugar al Truco (la versión Sudamericana). Estas serán sobre las que haremos un pequeño análisis.

### 4.1 Otras Apps similares

#### 4.1.1 Truco Free

El primer juego que analizaremos se llama Truco Free. Un rival sin duda muy potente que cuenta con más de 5 millones de descargas. Desarrollado por Blyts, una empresa que se dedica a desarrollar Apps de juegos, y que tiene actualmente 15 juegos disponibles para descargas en App Store para iOS.



Ilustración 4. Truco Free.



Ilustración 5. Truco Free.



Ilustración 6. Truco Free.

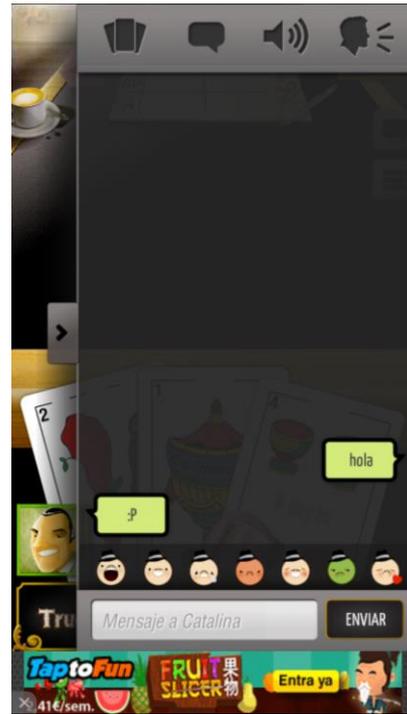


Ilustración 7. Truco Free.

Truco Free no especifica la forma de jugar del país que adquiere, simplemente dice que es la de América Latina. En el menú principal encontramos varias opciones:

- **Jugar.** Nos permite jugar una partida contra la máquina o buscar una partida online. Si queremos jugar online nos dice que debemos identificarnos y nos da tres opciones: mediante Facebook, mediante un email o ser un jugador Invitado. Cuando entramos en el modo multijugador nos deja buscar una partida rápida, meternos en una sala y ida, consultar las posiciones de los jugadores siguiendo un ranking (semanal o histórico), que se basa en puntos ganados o consultar nuestra lista de amigos.  
Nos deja jugar una partida online contra un sólo jugador, no por parejas.
- **Perfil.** Podemos gestionar nuestro perfil del juego. Podemos introducir nuestro apodo, nuestro sexo, si queremos flor o no (debe ser una regla de países sudamericanos), los puntos con los que se gana una partida, si activamos el chat o no y si queremos que el juego vaya rápido o lento. Además, podemos elegir entre varias voces, que se reproducen durante una partida cuando realizamos acciones.
- **Ayuda.** En este apartado nos encontramos con una explicación de las reglas del Truco y de cómo ganar puntos en el juego.
- **Anotador.** Si jugamos una partida física con jugadores reales, podemos entrar aquí e ir llevando la cuenta de puntos de forma sencilla. No tiene nada que ver con el juego en sí, es una funcionalidad aparte.

Como vemos en las capturas, el juego nos muestra publicidad continuamente en la parte inferior de la pantalla mientras estamos jugando, de esta forma gana dinero.

También vemos que tiene un sistema de chat para comunicarte con el otro jugador mientras juegas una partida. Esto puede ser interesante para comunicarte con tu pareja de juego si las incluyera.

Cuando hemos entrado varias veces al juego, nos aparece una ventana que nos dice si nos gustaría valorar el juego en App Store, esto es interesante para ganar puntos a favor.

La interfaz gráfica es bastante buena y hay buenas animaciones cuando eliges una carta.

Este es un juego bastante completo y lo hemos usado para sacar ideas para nuestro propio juego.

#### 4.1.2 Truco Argentino

Seguimos analizando la competencia directa con el Truco Argentino, aunque este no alcanza el nivel del anterior. Cuenta con más de 100,000 descargas y ha sido desarrollado por un hombre llamado Ignacio Raffa, que cuenta con 4 Apps más subidas.



Ilustración 8. Truco Argentino.



Ilustración 9. Truco Argentino.

El Truco Argentino nos presenta una interfaz gráfica muy pobre en la que incluso se tapa su propio logo con la publicidad.

En el menú principal tenemos 3 botones en rojo que son para jugar: Jugar a 15 y Jugar a 30 para jugar una partida contra la máquina a 15 o 30 puntos y jugar online para jugar una partida en línea contra otro jugador. Además, encontramos los botones Ayuda, que nos muestra como manejar el juego (ni siquiera las instrucciones del truc) y Acerca De que nos muestra quien ha desarrollado el juego.

Si elegimos jugar online entramos en el menú de la última captura, donde nos asigna un número de usuario y nuestras estadísticas. Si elegimos jugar ahora nos empareja con otro usuario y empezamos una partida online. Si elegimos jugar con amigo nos pide que introduzcamos su identificador para poder conectar con él.

La publicidad sigue apareciendo por todo el juego, tapándonos parte de las cartas incluso.

En definitiva, un juego bastante pobre si lo comparamos con el anterior.

#### 4.1.3 Truco Argentino Free

Por último, veremos el Truco Argentino Free, el menos trabajado de los tres. Desarrollado por Jaime García Ghirelli y sin datos de descargas. Disponible en App Store.



Ilustración 10. Truco Argentino Free. Ilustración 11. Truco Argentino Free.

El Truco Argentino Free es un juego para un único jugador en el que juegas contra la máquina una partida de Truc con las reglas de Argentina. Como podemos ver en las capturas, la interfaz gráfica es muy deficiente y todos los botones están activos para ser pulsados siempre, de tal forma que nunca sabes cuándo puedes pulsarlos o cuando no tiene sentido pulsarlos durante la partida.

Además, las instrucciones del juego se encuentran en la parte inferior de la pantalla, bajando con el dedo, pudiendo ocultar en la pantalla la partida y pararte a verlas.

No tiene modo online ni ranking, únicamente puedes jugar una partida contra la máquina.

## 4.2 ventaja sobre los competidores

De estos tres competidores que hemos analizado, es obvio que el más potente es el de punto 4.1.1, el Truco Free. Es un juego muy completo del que hemos sacado ideas.

A continuación vamos a hacer una comparación de nuestra aplicación y la de estos competidores. Analizaremos si tenemos sus características (tick verde o cruz roja) y la ventaja que tenemos sobre ellos. Pero cabe destacar que tenemos una ventaja exclusiva sobre todos que nos coloca por encima: nuestro juego es el único que tiene modo multijugador online para cuatro jugadores.

### Retruque! vs Truco Free

Menú principal ✓	Modo multijugador en línea para 2 jugadores ✓	Ranking ✓
Amigos en el juego ✓	Contador de puntos en partida ✓	Personajes ✗
Jugar contra la máquina ✗	Señas en modo multijugador ✓	Ayuda ✗
Gráficos HD ✓	Animaciones en la partida ✓	Sonidos ✓
Anotador fuera de partida ✗	Partida rápida contra jugador aleatorio ✓	Publicidad ✓

Tabla 1. Retruque! vs Truco Free.

Como podemos ver arriba, “Retruque!” tiene la mayoría de características que posee nuestro competidor más potente, por lo que podemos deducir con este primer análisis que, como mínimo, estamos muy cerca de su nivel.

### Retruque! vs Truco Argentino

Menú principal ✓	Sonidos en la App ✓	Marcador de puntos ✓
Acciones del juego (trucar, etc.) ✓	Ayuda ✗	Publicidad ✓
Jugar con amigos ✓		

Tabla 2. Retruque! vs Truco Argentino

En comparación con el Truco Argentino, podemos observar que tenemos prácticamente la totalidad de sus estadísticas, exceptuando la Ayuda que también tiene Truco Free. Nuestra App también es superior a esta.

### **Retruque! vs Truco Argentino Free**

Acciones del juego (trucar, etc.) ✓	Sonidos en el juego ✓	Reglas del Truc ✗
Marcador de puntos ✓		

**Tabla 3. Retruque! vs Truco Argentino Free**

Por último, comparamos con el Truco Argentino Free. No debemos de mirar mucho para darnos cuenta de que es el producto menos trabajado de los tres competidores y que no tiene muchas características. Nuestro juego vuelve a tener prácticamente la totalidad de estas características.

Tras analizar estos tres competidores, vamos a ver que en nuestro juego tenemos unas características que van a hacer que los superemos:

- **Multijugador online para 4 jugadores.** Esta, sin duda, es nuestra característica más relevante, ya que no hemos encontrado ningún juego que la tenga. Nos coloca directamente por encima de cualquier competidor al ser una característica única.
- **Sistema de logros.** No hay nada mejor para enganchar al jugador que desafiarle a que consiga ciertos objetivos del juego. Nuestra aplicación contiene 15 logros (unos más fáciles de conseguir que otros) para que el jugador ponga a prueba su capacidad e intente demostrar que es el mejor.
- **Tour inicial.** Cuando entras por primera vez al juego, éste muestra un pequeño tour inicial indicando al jugador qué acciones puede realizar desde el menú principal y le invita a que envíe una invitación a un amigo, por ejemplo.
- **Búsqueda de jugadores cercanos.** ¿No sabes con quién jugar y te apetece jugar con alguien de tu barrio? Retruque! Tiene un buscador de jugadores cercanos para que puedas desafiar a tus vecinos.
- **Perfil real de usuario.** Como hemos visto en las características de Truco Free, éste tiene un personaje ficticio que te representa en la partida. Nuestra aplicación usa tu perfil real de Google+ y muestra tu nombre y tu foto en la partida. Estos datos son únicamente mostrados cuando juegas con jugadores conocidos, para los demás jugadores aleatorios tu perfil queda oculto bajo el nombre de Jugador XXXX y una foto genérica.
- **Publicidad Interstitial.** Es la publicidad por la que más se paga. No queríamos molestar al jugador con los típicos banners apareciendo en mitad de una partida, ya que pensamos que podría resultar incluso molesto. Nuestra estrategia de publicidad es la llamada Interstitial, que consiste en una publicidad que se muestra al usuario a pantalla completa durante unos pocos segundos (5 en nuestro caso) y que puede ser cerrada en cuanto haya transcurrido este tiempo. En nuestro juego, se muestra una vez se ha acabado la partida, en la pantalla de resultados, por lo que el jugador no se verá molestado por publicidad durante el transcurso de la partida.

- **Frases en la partida.** ¿Quieres picar un poco a los demás jugadores? Tenemos más de 15 divertidas frases para añadir algo de picante a la partida. Puedes decir las cuando sea tu turno y todos los demás jugadores verán un bocadillo saliendo de tu icono de jugador junto con tu frase. ¿Qué te contestarán los otros jugadores?
- **Menú desplegable de acciones para el jugador.** Es molesto tener siempre los botones de acciones a la vista del jugador y ocupando espacio en la pantalla. Retruque! Tiene un menú desplegable con las acciones disponibles que el jugador podrá elegir si abrirlo o no conforme a sus necesidades en la partida.
- **Posibilidad de tapar la carta.** Una de las posibilidades del Truc es la de tapar tu carta durante tu turno. Nuestro juego es el único que la tiene.
- **Diálogos con material design para preguntas al jugador.** Cuando un jugador envidia o truca, los demás jugadores verán un moderno diálogo preguntando sobre qué acción desean realizar. Además, se muestra en la parte superior de la pantalla para que el jugador pueda ver sus cartas y, de esta forma, tomar mejor su decisión. En los competidores son pequeños botones sin mayor relevancia.
- **Tiempo para los turnos visible.** Cada jugador tiene 40 segundos en su turno que puede usar para decidir que carta tirar o realizar alguna de las acciones que le estén permitidas. Debajo de su icono de perfil aparecerá una barra de tiempo que permitirá a él mismo y a los demás jugadores tener la información del tiempo restante que le queda a ese jugador. En caso de que el jugador que tiene el turno no haya tirado ninguna carta durante su tiempo, el juego tirará una automáticamente por él para que la partida pueda seguir avanzando.
- **Partidas mixtas.** Es posible jugar partidas mixtas con amigos y con jugadores aleatorios. Por ejemplo, se puede invitar a un amigo y buscar a otros dos jugadores aleatorios para así jugar una partida de cuatro jugadores. Retruque! Es el único juego que permite esta acción.
- **Fondos de partida dinámicos.** Jugar una partida siempre con el mismo fondo puede llegar a resultar monótono. Nuestra aplicación elige un fondo aleatorio de entre varios disponibles para cada partida, aportando así un toque distinto y novedoso.

Éstas son las características principales que nos diferencian sobre nuestros competidores. Creemos que ha sido positivo analizarlos a fondo para así tomar ideas y ver cómo era posible superarlos.

Este análisis nos ha hecho ver que todos disponen de un apartado de Ayuda que permite que jugadores que no saben jugar al Truc puedan ver las reglas y, de ese modo, poder aprender mientras disfrutan del juego. Lo tendremos en cuenta para añadirlo en próximas versiones.

Sin lugar a dudas, nuevas ideas y características incluidas en Retruque! nos sitúan por encima de nuestros competidores de forma notable. Destacando, como ya se ha citado anteriormente, el exclusivo modo de 4 jugadores en línea que posee nuestro juego. Si a eso le añadimos que somos el único juego que contiene las reglas del Truc valenciano (todos los productos usan las reglas de la versión Sudamericana), podemos esperar un gran éxito del juego en nuestra zona geográfica

## 5. Estudio de la idea



Ilustración 12. Estudio de la idea.

Llegamos al punto 5 de la memoria, en el que vamos a hacer un estudio de la idea de proyecto que tuvimos, siguiendo una serie de análisis. Empezaremos por el Lean Canvas, seguido del DAFO y analizaremos la proyección económica de la App para finalizar.

Para concretar un poco más sobre los tres análisis, vamos a definirlos brevemente:

- **Lean Canvas:** Lean Canvas es una herramienta rápida y eficaz para crear y comunicar modelos de negocio para startups. Su creador, Ash Maurya, te propone un método híbrido basado en Canvas Business Model y la metodología Lean, de la adaptación surge una herramienta específica para que las startups puedan diseñar modelos de negocio de la forma más eficaz. Lean Canvas ayuda a descomponer tu modelo de negocio en nueve partes que luego serán sistemáticamente sometidas a pruebas en orden de menor a mayor riesgo.
- **Análisis DAFO:** Es una metodología de estudio de la situación de una empresa o un proyecto, analizando sus características internas (Debilidades y Fortalezas) y su situación externa (Amenazas y Oportunidades) en una matriz cuadrada. Es una herramienta para conocer la situación real en que se encuentra una organización, empresa o proyecto, y planear una estrategia de futuro. El objetivo del análisis DAFO es determinar las ventajas competitivas de la empresa bajo análisis y la estrategia genérica a emplear por la misma que más le convenga en función de sus características propias y de las del mercado en que se mueve.
- **Proyección económica:** Un análisis de dos puntos: ingresos y tiempo. Su principal objetivo es estimar en un periodo de tiempo determinado, cuáles pueden ser los ingresos del proyecto y si puede salir rentable, así como prevenir el crecimiento o decrecimiento de los ingresos que pueda aparecer en un momento del periodo de vida.

## 5.1 Lean Canvas

A continuación, veremos la tabla del Lean Canvas.

<p><b>2</b> <b>Problema</b></p> <ul style="list-style-type: none"> <li>- En la actualidad no existe un juego para poder jugar al truc online para Android. Al menos de la versión española, sí que existen versiones de países sudamericanos con distintas reglas de juego.</li> <li>- Hay muchos aficionados al truc a los que le gustaría jugar una partida en un rato libre y no pueden, ya que hace falta reunir varios jugadores.</li> <li>- A muchos jugadores o incluso parejas de jugadores les atrae la posibilidad de demostrar que son mejores que otros al truc.</li> </ul>	<p><b>4</b> <b>Solución</b></p> <ul style="list-style-type: none"> <li>- Modo online para jugar desde cualquier parte.</li> <li>- Posibilidad de invitar a amigos y mandarles notificaciones para poder jugar una partida online.</li> <li>- Un sistema de estadísticas o de clasificación según partidas ganadas o perdidas podría crear un ranking de mejores jugadores del juego.</li> </ul>	<p><b>3</b> <b>Proposición única de valor</b></p> <p>Ofrecemos la posibilidad de que los jugadores del truc online disfruten de una partida de truc con sus amigos o con otros jugadores desconocidos en el momento que quieran, con la comodidad de no tener que reunirse físicamente con los demás jugadores y desde el sitio que quieran, simplemente con una conexión a internet. También crearemos un ranking para poder comparar al jugador con sus amigos y demostrar quien es el mejor jugando al truc.</p>	<p><b>9</b> <b>Ventaja injusta</b></p> <p>Existen versiones del truc de países sudamericanos, no podríamos copiarlos el mismo juego y cambiando las reglas. Necesitamos intentar innovar en cuanto a funcionalidades para competir.</p>	<p><b>1</b> <b>Segmentos de Clientes</b></p> <p><b>Clientes objetivo:</b> Aficionados a jugar al truc. Españoles y especialmente valencianos. No hay una edad determinada, ya que los juegos de cartas tienen un rango de edad de los jugadores muy amplio (podríamos estar hablando desde los 14 hasta los 60 años aproximadamente).</p>
	<p><b>8</b> <b>Métricas clave</b></p> <p>Algunas posibles métricas:</p> <ul style="list-style-type: none"> <li>- Número de partidas semanales.</li> <li>- Número de descargas mensuales.</li> <li>- Número de jugadores semanales.</li> </ul>		<p><b>5</b> <b>Canales</b></p> <p>Nuestro canal de distribución para los clientes será la tienda de aplicaciones de Android: Google Play.</p>	<p><b>Early adopters:</b> Tenemos a nuestra disposición a un grupo de amigos a los que les gusta jugar al truc y que podrían ser nuestros primeros jugadores que prueben las distintas versiones que vayamos diseñando.</p>

## 7 Estructura de costes

Como costes fijos tendríamos el coste de subir la aplicación a Google Play y la tarifa mensual de la contratación de un servidor donde poder alojar las partidas online.

## 6 Ingresos

Nuestra fuente de ingresos será la publicidad que pondremos en el juego. También cabe la posibilidad de ofrecer funcionalidades especiales a usuarios Premium o de ofrecer el mismo juego sin publicidad pero pagando una cantidad por la descarga inicial.

Además, en el juego podemos incluir monedas que apostarás para poder jugar partidas. Esas monedas o algunas mejoras como el diseño de las cartas podrían comprarse con dinero real.

Tabla 4. Lean Canvas.

## 5.2 Análisis DAFO

Aquí tenemos la primera parte del análisis DAFO, que está dividida en cuatro puntos.

Fortalezas		Peso	Debilidades		Peso
Originalidad de Productos/Servicios	8		Marketing Efectivo	5	
Precio Competitivo	10		Base de Clientes	5	
Innovación de Productos/Servicios	10		Alianzas de Negocios	3	
Ubicación Geográfica	9		Presencia Online	3	
Liderazgo en el Mercado	7		Reputación	3	
Calidad de Productos/Servicios	6		Relación con los Proveedores	2	
Cuota del Mercado	6		Costos	2	
Plantas y Equipamiento	4		Estado Financiero	2	
Posicionamiento en el Mercado	4		Cadena de Distribución	1	
Personal Experimentado	4				
Propiedad de Patentes/Tecnologías	2				
<b>Total</b>	<b>70</b>		<b>Total</b>	<b>26</b>	
Oportunidades		Peso	Amenazas		Peso
Mayor utilización de Internet por los Consumidores	5		Adquisición de Empresa Proveedora de Tecnología	3	
<b>Total</b>	<b>5</b>		<b>Total</b>	<b>3</b>	

Tabla 5. Análisis DAFO.

Como vemos, la mayoría de puntos asignados a nuestro proyecto son para las fortalezas, ya que como hemos comentado anteriormente, es un producto totalmente novedoso, sin competidores directos y con una gran esperanza de éxito.

Por otra parte, también tenemos debilidades, como no. Nuestras principales debilidades son que no somos desarrolladores conocidos ni hemos llevado otros proyectos por lo que puede costar el hecho de expandirnos y dar a conocer nuestra aplicación. Tampoco tenemos una base de clientes determinada más allá de la gente cercana con la que nos podamos comunicar. Además, cualquier coste del proyecto ha salido directamente de nuestro bolsillo, por muy pequeño que sea.

En cuanto a las oportunidades y amenazas, no tenemos ningún punto destacado que comentar.

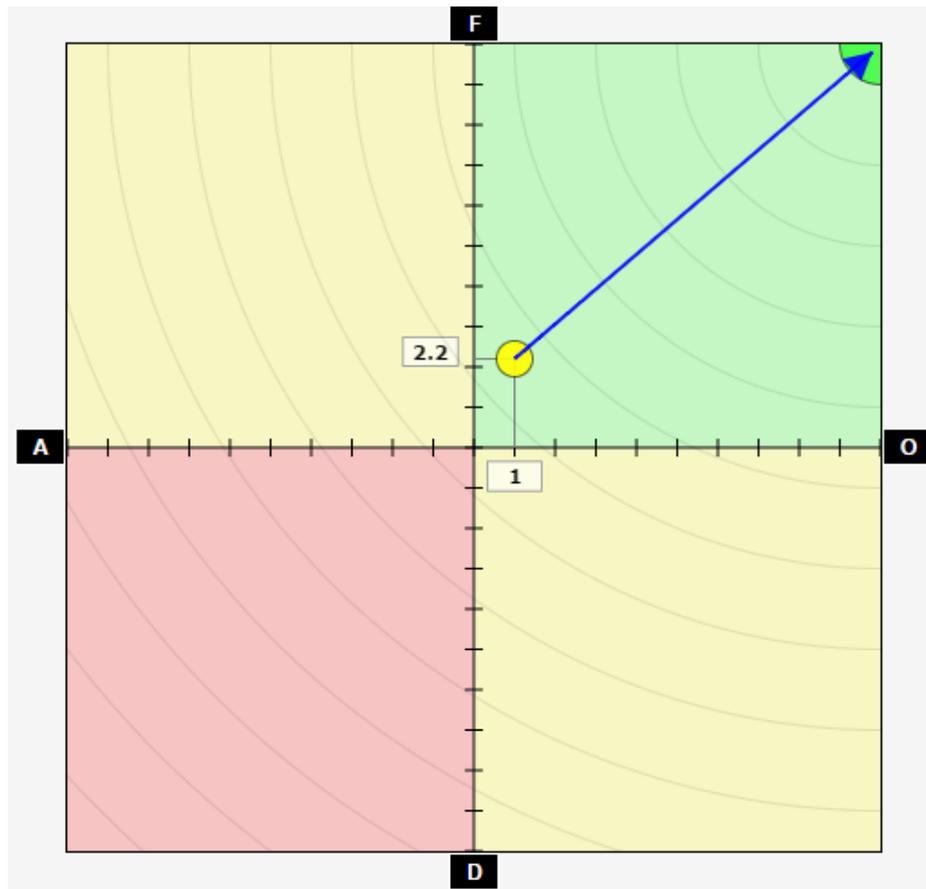


Ilustración 13. Grafo generado por el Análisis DAFO.

En el grafo que hemos generado (**Ilustración 13**), podemos observar como nuestro proyecto se ubica completamente en el cuarto de las fortalezas, por todo lo que hemos comentado anteriormente.

Por ello, de este análisis podemos deducir que, aparentemente, nuestro proyecto tendrá éxito.

### 5.3 Análisis económico

Para acabar con el punto 5, efectuaremos un análisis económico de nuestro producto.

Comenzaremos con el análisis económico del primer año de vida de la aplicación. En este primer periodo nos encontramos con una aplicación nueva, que no obtendrá un gran número de descargas y que tendrá una inversión inicial.

Los únicos beneficios que obtendremos serán de la publicidad y de las descargas que obtenga nuestra aplicación, por lo que será esencial la expansión de esta para recuperar gastos y obtener algo de beneficio.

Cabe destacar que tanto los gastos como los beneficios serán repartidos entre dos personas, al ser un proyecto conjunto.

<b>Retruque!</b>		<b>Primer año de vida de la App</b>	
<b>Costos Fijos</b>			
Descripcion	Detalle	Gasto Mensual	Gasto Anual
Alta como desarrollador Google	Darse de alta como desarrollador de Google Play	0,00	25,00
Diseñador gráfico	Servicio de diseñador gráfico para iconos de la App	0,00	50,00
Publicidad en línea	Adwords (aproximado 0,05 por clic)	20,00	240,00
<b>TOTAL</b>		20,00	315,00
<b>Ingresos</b>			
Descripcion	Detalle	Ingreso Mensual	Ingreso Anual
Descargas	0,02 por cada descarga (100 mensuales)	20,00	240,00
Publicidad	(100000 impresiones, 5% CTR, 0,12/click)	80,00	960,00
<b>TOTAL</b>		80,00	1200,00
<b>TOTAL Ingreso - Gastos</b>		60,00	885,00

**Tabla 6. Costos e ingresos, del primer año de vida de la App.**

En cuanto a los años de vida posteriores del juego, prevemos un aumento notable de los beneficios. En este periodo, la App debe de estar expandida sobre todo por la Comunidad Valenciana, habremos añadido mejoras y tendrá un gran número de usuarios y partidas diarias.

Nuestros ingresos seguirán siendo por publicidad y por descargas en su mayor parte pero también existirá contenido Premium en el juego, del que podemos sacar un beneficio mensual.

Por otra parte, nuestra aplicación no necesita ninguna otra inversión aparte de la publicidad que podamos contratar para darla a conocer, por lo que los gastos no variarán demasiado.

Como se ve en la **tabla 7**, los ingresos anuales aumentarán notablemente.

<b>Retruque!</b>		<b>Años posteriores</b>	
<b>Costos Fijos</b>			
Descripcion	Detalle	Gasto Mensual	Gasto Anual
Publicidad en línea	Adwords (aproximado 0,05 por clic)	40,00	480,00
<b>TOTAL</b>		40,00	480,00
<b>Ingresos</b>			
Descripcion	Detalle	Ingreso Mensual	Ingreso Anual
Descargas	0,02 por cada descarga (500 mensuales)	10,00	120,00
Elementos premium	Elementos premium de los que disponga el juego	50,00	600,00
Publicidad	(100000 impresiones, 5% CTR, 0,12/click)	250,00	3000,00
<b>TOTAL</b>		310,00	3720,00
<b>TOTAL Ingreso - Gastos</b>		270,00	3240,00

**Tabla 7. Costos e ingresos en los años posteriores.**

Es posible que el ritmo de descargas varíe o disminuya con el tiempo, ya que está acotado geográficamente y en algún momento (cuando se haya expandido en gran medida), las descargas comenzarán a reducirse. Sin embargo, se seguirá ganando dinero por los demás elementos.

## 6. Proceso de desarrollo



Ilustración 14. Proceso de desarrollo.

En esta sexta parte de la memoria vamos a describir el proceso de desarrollo de “Retruque!”.

Nuestro proyecto nos ha llevado, aproximadamente, 8 meses desde principio hasta el final. Suena a mucho tiempo, pero vamos a explicar un poco la dedicación y tareas realizadas.

La idea nos surgió al quedar varias veces con nuestro profesor, buscando el proyecto adecuado que nos gustaría hacer. Buscábamos un producto novedoso, que nos gustara hacer y que fuera emprendedor. Tras barajar varias alternativas, decidimos hacer un Truc para jugar online.

Durante el primer mes y el segundo de reuniones con nuestro profesor, concretamos muy bien todas las características que debería tener nuestro juego, así como los distintos análisis tanto de nuestro producto como de los juegos que pudieran hacernos competencia.

Más tarde, estuvimos un par de meses apartados del proyecto, ya que debíamos acabar con las últimas asignaturas que nos quedaban y no teníamos tiempo que dedicarle al proyecto.

Los últimos 3 meses, como se ha dicho antes, los dedicamos completamente al desarrollo del juego, ya que el trabajo previo que habíamos hecho junto con el profesor nos permitió tener las ideas y los objetivos muy claros desde el primer momento. Sabíamos lo que queríamos pero no sabíamos cómo hacerlo, ya que nunca habíamos hecho un juego online y no teníamos mucha idea de cómo empezar a desarrollar.

Empezamos un trabajo de pruebas e investigación que nos iba a llevar aproximadamente 2 semanas, en las que probamos distintas tecnologías como el login con Facebook o tener nuestro propio servidor para las partidas, pero encontramos Google Play Game Services y vimos que nos facilitaría muchísimo el trabajo, de ahí nuestra elección final.

Una vez tuvimos claras las tecnologías que íbamos a usar y cómo tenía que ser nuestra aplicación, pudimos comenzar finalmente a programar en Android Studio. Este proceso dura desde esta etapa hasta la publicación de la App en Play Store y dura, como ya se ha dicho anteriormente, unos 3 meses.

Decidimos empezar logeando a los jugadores con Google+ y con el modo para 2 jugadores, que nos costó un poco, ya que tuvimos que aprender cómo funcionaba Google Play Game Services y cómo podríamos comunicar a los jugadores.

Una vez entendimos el funcionamiento de esta tecnología, fuimos poco a poco avanzando hasta conseguir terminar el modo de 2 y abordar el modo de 4, que añadía bastante dificultad a la programación.

Sin embargo, nos costó bastante menos de lo que creíamos, debido a que la base que teníamos del modo de 2 jugadores nos iba a resultar muy útil.

Cuando conseguimos acabar el modo de 4, incluimos también un sistema de logros para desafiar a los jugadores y un ranking de victorias, así como una mejora notable de la interfaz gráfica con animaciones y sonidos que resultarían muy atractivas para el usuario final.

## 6.1 Mapa de características



Ilustración 15. Mapa de características.

En la **ilustración 15** observamos el mapa de las características que tenía que tener nuestra aplicación. Como se ha dicho en el apartado anterior, fueron consensuadas junto con el profesor y las estructuramos en distintas etapas o Sprints, separándolas de mayor a menor importancia.

Cabe destacar que todas las características de la primera línea (las que entendíamos que eran las esenciales para subir el juego a Play Store) han sido incluidas en su totalidad en la primera versión de la App, pero también hemos incluido la mayor parte de la segunda fila, que en un principio iban a entrar en la segunda versión.

El motivo de esta inclusión de características fue que nos vimos capaces de desarrollarlas para la primera versión y quisimos hacer un juego mucho mejor del que teníamos pensado en su versión inicial. Pudimos incluir los efectos de sonido, el sistema de logros, las señas para 4 jugadores y una remodelación de la interfaz gráfica de la aplicación, haciéndola mucho más visual y divertida para el jugador.

Por tanto, nos dejamos las características restantes y la tercera fila para una próxima versión, que iremos desarrollando con el paso del tiempo.

## 7. Tecnologías utilizadas

En nuestro proyecto hemos utilizado numerosas tecnologías. Hacer una aplicación para Android no es una tarea sencilla y necesita muchos elementos para realizarse. A continuación, pasamos a describirlas.

### 7.1 Android

#### 7.1.1 Dispositivos móviles y sus características

Un dispositivo móvil es un aparato de pequeño tamaño, con capacidades de proceso limitadas con respecto a otros dispositivos. Además nos puede proporcionar conexión a una red y poseen memoria limitada, diseñados específicamente para una función, pudiendo llevar a cabo otras funciones más generales y variadas.

Entre sus características más destacables se encuentran las siguientes:

- Reducido tamaño, que junto con su peso reducido facilitan su movilidad.
- Equipamiento específico tanto de procesamiento, como de almacenamiento y alimentación, con la finalidad de mejorar la autonomía.
- Proporcionan servicios de conectividad inalámbrica.
- Almacenamiento local de datos, así como sincronización con otras plataformas.
- Interacción con el usuario adaptada.

Algunas de las tecnologías que posibilitan el desarrollo de este tipo de dispositivos son:

HSDPA, ARM, 3G/3.5G, Memorias Flash, Servlets, WIFI, Li-ion, OLED, Multitouch, teclados virtuales, Pantallas capacitivas, Bluetooth, Pantallas resistivas, Cloud computing, AMOLED, microSD HC, SoC, GPU, Web services, GPS/AGPS

El verdadero éxito de los dispositivos móviles se encuentra en su capacidad para ofrecer múltiples funciones y servicios a sus usuarios.

#### 7.1.2 ¿Qué es Android?

Android es una plataforma software diseñada con el objetivo de abstraer el hardware y facilitar el desarrollo de aplicaciones para dispositivos con recursos limitados, generalmente dispositivos móviles.

- Se trata de una plataforma “open source” y se distribuye con licencia Apache 2.0.
- Define toda una pila software, incluyendo el sistema operativo a utilizar, middleware, soporte de ejecución para las aplicaciones y un conjunto mínimo de apps básicas.
- Se apoya en el uso de una máquina virtual propia, basada en el kernel de Linux.
- Proporciona un conjunto completo de APIs y herramientas de desarrollo, compilación, depuración y emulación.

Inicialmente, Android fue una propuesta del *Open Handset Alliance*, una organización dedicada al desarrollo de software libre, aunque su expansión ha sido propiciada por el uso masivo de los servicios propietarios de Google, afectando de esta forma a la naturaleza “open source” de la plataforma.



Ilustración 16. Diferentes versiones de Android.

### 7.1.3 El problema de la fragmentación

La fragmentación en Android hace referencia a la variedad de versiones y especificaciones técnicas de hardware (forma, resolución de pantalla, tamaño, prestaciones, etc.), en definitiva, hay una enorme variedad de entornos Android.

- Esta variedad de entornos dificulta que los programadores adapten sus aplicaciones al universo de dispositivos disponibles en el mercado. En ocasiones supone un problema a la hora de recibir actualizaciones por parte de los fabricantes.
- Esta variedad ha permitido que la plataforma penetre con fuerza en multitud de mercados, adaptándose tanto al poder adquisitivo como a las necesidades de los compradores existentes en cada uno.

### 7.1.4 Cómo es android

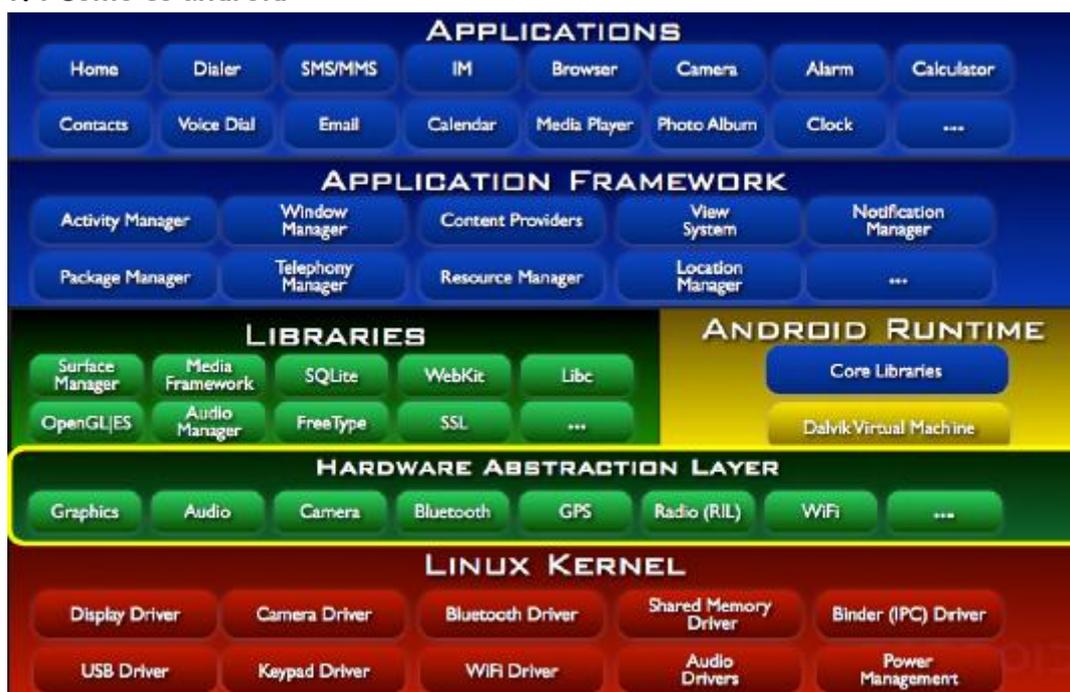


Ilustración 17. Cómo es Android.

A continuación estudiaremos cada uno de estos módulos con mayor detenimiento.

### 7.1.5 Kernel linux

Android está basado en un kernel probado de Linux, dotando de una gran fiabilidad al sistema. Es la única capa que depende del hardware del dispositivo y es responsabilidad del fabricante. Android se aprovecha del kernel para la mejora de la seguridad, la gestión de memoria y procesos así como la gestión de red y alimentación.

Android utiliza un modelo de controladores, que proporciona soporte a los nuevos accesorios a través de controladores. De esta forma desacopla el hardware del software a través de una capa de abstracción HAL (Hardware Abstraction Layer).

### 7.1.6 Librerías Android

Las librerías más importantes son las siguientes:

- **Surface Manager:** Gestión de acceso a pantalla
- **Media Framework:** Reproducción de imágenes, audio y video.
- **SQLite:** Potente y ligero motor para bases de datos relacionales.
- **WebKit:** Navegador web usado por el navegador de Android y los WebView.
- **SGL:** Gráficos 2D.
- **OpenGL:** Gráficos 3D.
- **FreeType:** Renderización vectorial y fuentes en bitmap.
- **SSL:** Capa de comunicación segura por sockets.
- **Libc:** Reimplementación optimizada de las librerías en C.

### 7.1.7 Runtime de Android

Android hace uso de dos máquinas virtuales muy optimizadas, si tenemos en cuenta las condiciones que imponen los dispositivos móviles en la actualidad. Estas máquinas virtuales ejecutan bytecode de tipo Dalvik y son Dalvic Virtul Machine (DVM) y Android Runtime (ART).

- **Máquina Virtual Dalvik:**
  - Uso del kernel para la ejecución de apps (1app = 1 proceso = 1DVM), lo que implica la ejecución concurrente de instancias de la DVM.
  - Utiliza los servicios ofrecidos por el OS de aislamiento de procesos, gestión de memoria y soporte multihilo.
  - No es una máquina virtual de Java (JVM) → ejecuta aplicaciones en formato propio Dalvik Executable (DEX) y no soporta clases AWT o Swing.
  - Aplicación de compilación de tipo JIT (just-in-time) → conversión de ejecutables DEX a código máquina cuando dicho código se necesita y solo entonces.
- **Android Runtime (ART):**
  - Aplicación de compilación de tipo Ahead-of-time, que convierte el ejecutable DEX a código máquina cuando se instalan las aplicaciones para una mejora de las prestaciones.
  - Mejora la recolección de basura Dalvik.
  - Introduce mejoras de depuración como informes más detallados en caso de excepción y crash de las aplicaciones.
  - Opcional en KitKat, pero obligatorio en Lollipop.

### 7.1.8 Entorno de aplicaciones

El entorno de aplicaciones nos proporciona una plataforma de desarrollo que permite la reutilización de componentes, en la que cada aplicación publica sus capacidades y se utilizan unas a otras en base a ellas.

A través de este framework, el desarrollador puede acceder al hardware de los dispositivos, localizarse, gestionar el ciclo de vida de las aplicaciones etc.

Las aplicaciones básicas (core applications) del sistema también utilizan las APIs ofrecidas por el framework.

### 7.1.9 Nivel aplicativo

En esta capa se ubican las aplicaciones preinstaladas por los fabricantes y operadores y se instalan las que deseen los usuarios. Las aplicaciones bien se desarrollan en Java utilizando el SDK (Software Development Kit) de Android o en código nativo C/C++, lo que requiere del uso del NDK (Native Development Kit) de Android.

Entre las aplicaciones incluidas por defecto destaca la aplicación de Google Play, que ofrece al resto de apps servicios esenciales para, por ejemplo, el trabajo con los mapas o la facturación a clientes.

### 7.1.10 Inicialización del Sistema

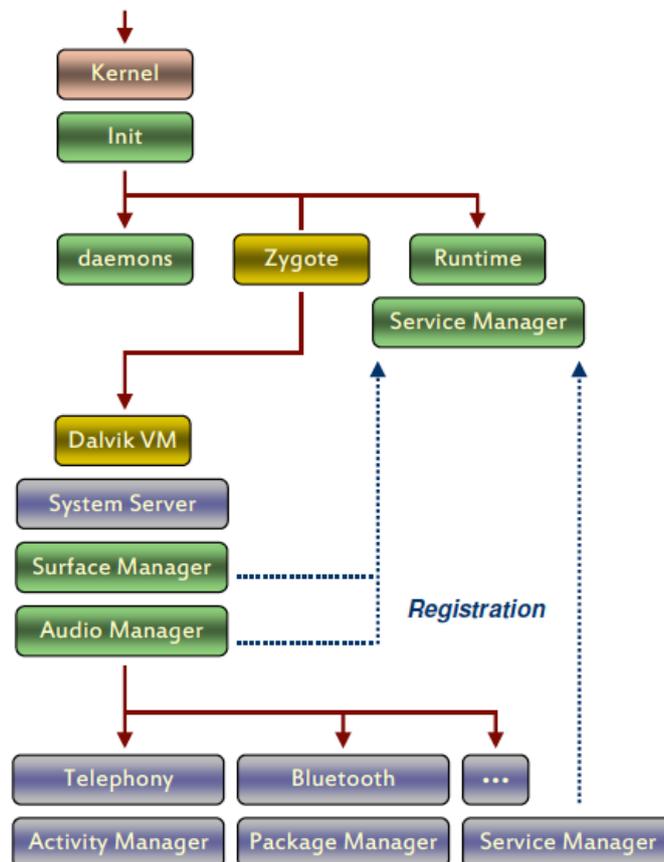


Ilustración 18. Inicialización del sistema.

1. El cargador arranca el kernel y lanza a ejecución el Init.
2. Se lanzan los procesos demonios que gestionan los recursos hardware (USB, radio...).
3. Se crea el proceso Zygote, que no es más que una instancia de la máquina virtual.
4. El proceso Runtime solicita a Zygote que lance una nueva instancia de la máquina virtual para ejecutar el servidor del sistema,
5. Los primeros procesos que se ejecutan gestionan la pantalla y el audio.
6. A continuación se lanzan el resto de procesos.

#### 7.1.11 Aplicaciones

Centrándonos ya en el desarrollo de aplicaciones en Android debemos destacar los siguientes componentes:

- **Activity:** Componente que define la interfaz de usuarios, típicamente asociado a una pantalla de la aplicación.
- **Broadcast Receiver:** Componente que responde a notificaciones o cambios de estado. Su activación puede conllevar la creación de un proceso.
- **Service:** Componente sin interfaz que realiza una tarea en segundo plano.
- **Content Provider:** Componente que permite a las aplicaciones compartir información.

En este contexto, el desarrollo en Android es centrado en Activities o Actividades.

El concepto de Actividad (Activity) viene a estar relacionado con cada una de las pantallas de nuestra aplicación, de esta forma su función se asocia a la interfaz de usuario, es decir, cada actividad se asociará a una pantalla de la aplicación y definirá su funcionalidad. Toda actividad tiene una vista asociada, generalmente un Layout, y que será lo que se le mostrará al usuario. Además toda Actividad ha de ser una subclase de la clase Activity.

Una aplicación está formada por varias actividades independientes que colaboran con el objetivo común de satisfacer las necesidades del usuario.

#### 7.1.12 Ciclo de vida

Cada aplicación Android se ejecuta en su propio proceso Linux y posee su propia instancia de la máquina virtual, de esta forma vamos a tener el siguiente procedimiento.

Cuando una petición necesita de la intervención de un componente de la aplicación, Android se asegura de que el proceso asociado a la aplicación este en ejecución, arrancándolo en caso de necesidad. También verifica que la instancia adecuada del componente este disponible y en caso de no estarlo, crea una nueva.

El proceso Linux que encapsula toda la aplicación se crea cuando el código de la misma debe ejecutarse, es decir, en el momento de arrancar la aplicación. Así pues permanece en ejecución hasta que, o bien no se necesita más, o bien el sistema reclama los recursos ocupados para otras aplicaciones. Como vamos a ver ahora, los usuarios no controlan el ciclo de vida de las aplicaciones, es el propio sistema el que lo hace.

El ciclo de vida de los componentes tiene los siguientes estados:

- El inicio. Android instancia los componentes en respuesta a un Intent. Éste se corresponde con la intención de arrancar la aplicación.
- El fin. Android destruye esta instancia de la aplicación.
- Entretanto, su vida transita entre los estados activo e inactivo, o en el caso de las Actividades, el de visible e invisible para el usuario.

El ciclo de vida completo va desde onCreate, método que inicializa la Actividad principal por primera vez, hasta onDestroy, donde se destruye tanto la aplicación como su proceso. Las Actividades son visibles entre onStart y onStop, pero tan solo entre onResume y onPause son interactivas y se ejecutan en primer plano permitiendo la interacción con el usuario.

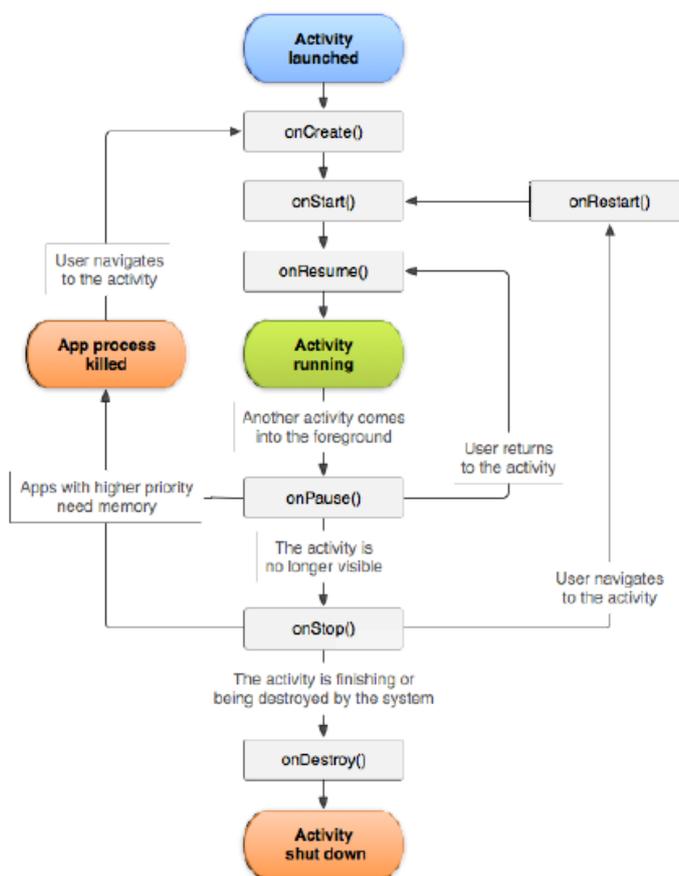


Ilustración 19. Ciclo de vida de una Actividad.

Un control correcto del ciclo de vida nos va a permitir desarrollar una aplicación fiable y estable, y por ello se recomienda sobrescribir todos los métodos de transición entre actividades. Entre ellos los más importantes, el onCreate para inicializar la Actividad de la forma que deseamos, así como el método onPause cuando se necesario asegurar que los cambios en los datos se almacenan correctamente y se gestiona correctamente el fin de la interacción con el usuario.

### 7.1.13 Los procesos en Android

En Android, la destrucción del proceso de una aplicación no la decide el usuario sino el sistema, de modo que su proceso continuará en memoria incluso después de que el usuario cierre la aplicación que se está ejecutando.

Una app sólo es eliminada cuando el sistema necesita memoria para otra. En casos muy extremos en los que el sistema necesite memoria de forma urgente (ante una llamada telefónica por ejemplo), el sistema podrá eliminar de memoria incluso apps en ejecución, es decir, con componentes activos. Por este es conveniente realizar un buen seguimiento del ciclo de vida de nuestra aplicación, con el fin de no perder datos de cierta relevancia para el usuario.

Para determinar que procesos eliminar, Android los ordenará jerárquicamente según su importancia relativa.

### 7.1.14 Tipos de procesos

#### 1. Proceso en primer plano (Foreground process)

- Contiene a la Actividad en ejecución con la cual el usuario interactúa, la última sobre la que se activó el método onResume.
- Actualmente, sólo puede haber un proceso de este tipo en todo el sistema y nunca debería ser eliminado.

#### 2. Proceso visible (Visible process)

- Contiene una actividad visible en pantalla pero que no está en ejecución, por tanto se ha invocado su método onPause.
- Se considera importante porque suele contener una de las últimas aplicaciones que el usuario utilizó, y quizás quiera volver a relanzarla.
- No se suele eliminar, a menos que sea necesario para mantener en ejecución al proceso en primer plano.

#### 3. Proceso en segundo plano (Background process)

- Alberga actividades que o son visibles, en las que se ha invocado su método onStop.
- Su eliminación no tienen un impacto directo en la experiencia que el usuario tiene de la aplicación.
- Cuando el usuario la vuelva a lanzar y navegue hacia alguno de los componentes del proceso, éste se puede instanciar nuevamente, aunque su estado de perderá y deberá restaurarse.
- Normalmente hay muchos procesos de este tipo, por lo que el sistema intenta asegurar que el que contenga la última actividad vista por el usuario sea el último en ser eliminado.

#### 4. Proceso vacío (Empty process)

- No aloja ningún componente activo.
- Estos procesos se mantienen en memoria a modo de “caché” para mejorar los tiempos de activación de las aplicaciones cuando estas vuelven a ser utilizadas y relanzadas por los usuarios.

### 7.1.15 Intents

Hasta ahora hemos visto que las Actividades son las pantallas de nuestra aplicación, como crearlas y destruirlas, y su ciclo de vida entre otros, pero ¿Cómo navegamos entre ellas?

Para ello se introduce el concepto de Intent. Un Intent es un mensaje asíncrono que facilita el intercambio de información entre distintos componentes de la plataforma Android, no solo entre Actividades. Usaremos los Intents cada vez que queramos:

- Lanzar una Actividad, mediante `startActivity` y `startActivityForResult`.
- Lanzar un servicio, mediante `startService`.
- Remitir una notificación, uso de `sendBroadcast`.
- Comunicarnos con un servicio con `bindService`.

Los intents permiten indicar acciones genéricas, por ejemplo, si nuestra aplicación es una agenda y queremos llamar a un contacto, el Intent lanzará la aplicación registrada por defecto a tal propósito, en este caso la aplicación del teléfono. Si se diese el caso en el que quisiésemos mandar un correo y tuviésemos más de una aplicación que nos pudiera satisfacer esta petición (Gmail e Inbox), se abriría un desplegable en el cual el usuario indicaría que aplicación arrancar.

Existen dos tipos de Intents, el Intent explícito y el Intent implícito.

- **Intent explícito:** Se indica exactamente el componente a lanzar, en este caso siempre haremos referencia a una Actividad de nuestra aplicación.
- **Intent implícito:** Son solicitudes de tareas abstractas, en las que el destinatario del Intent se determina en tiempo de ejecución y como hemos visto antes, puede que se requiera de la intervención del usuario si el sistema encuentra más de una Actividad que satisfaga la petición. Por ejemplo, “quiero tomar una foto” o “quiero compartir algo” (Facebook, Twitter, Whatsapp).



## 7.2 Android Studio

Android Studio es el IDE oficial para el desarrollo de aplicaciones en Android. Aunque se puede desarrollar en otras plataformas como Eclipse, nosotros escogimos el IDE por excelencia de Google. Android Studio está basado en IntelliJ IDEA. De las capacidades más interesantes que esperamos de IntelliJ, Android Studio nos ofrece:

- Gradle-based build system: Sistema para la compilación y construcción de dependencias llamado Gradle.
- Otros sistemas de construcción como Maven, además de muchas opciones para la generación de APK.
- Plantillas de código que ayudan a desarrollar características frecuentes de aplicaciones.
- Editor de layouts muy avanzado, con soporte drag and drop para la edición de los temas de las aplicaciones. También soporte XML del mismo.
- Herramientas para capturar el rendimiento, la usabilidad, la compatibilidad de versiones y otros problemas.
- Capacidad de firmas para las aplicaciones.
- Soporte para la plataforma Google Cloud, facilitando la integración con Google Cloud Messaging y App Engine.

A continuación veremos una introducción a algunas características clave de Android Studio.

### 7.2.1 Proyectos y estructura de archivos

Por defecto, Android Studio muestra los archivos del proyecto en la vista del proyecto. Esta vista nos proporciona un acceso rápido a los archivos fuente más importantes además de mostrarnos la estructura de nuestro proyecto. También nos ayuda a trabajar con Gradle.

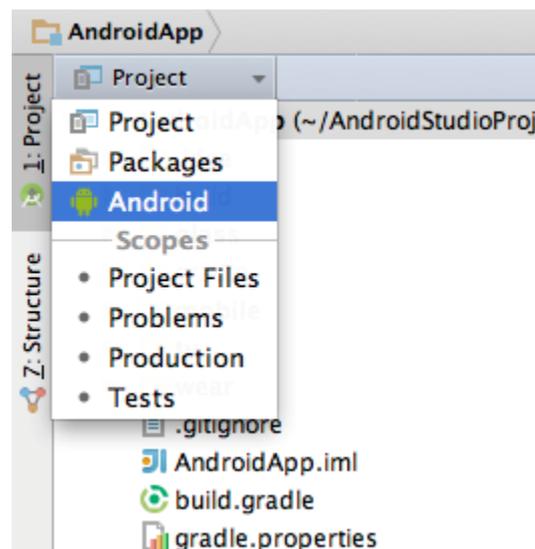


Ilustración 20. Vista del proyecto.

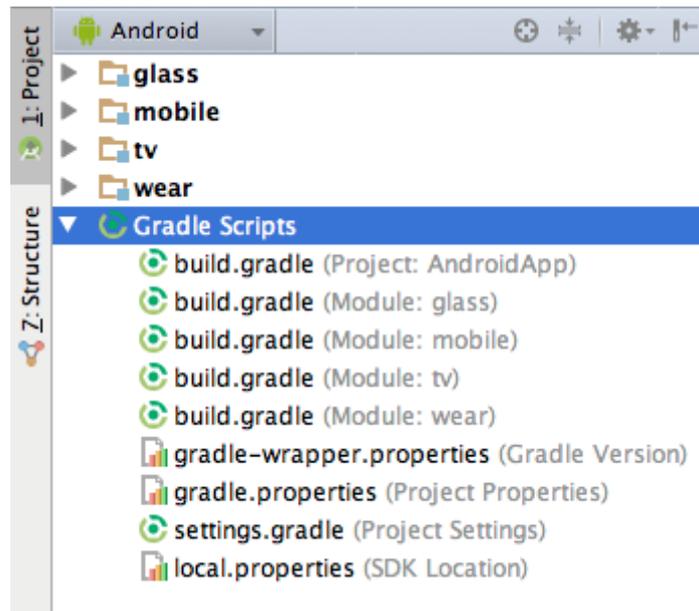


Ilustración 21. Archivos Gradle del proyecto.

La vista del proyecto contiene todos los archivos en el nivel más alto de la jerarquía, bajo los scripts de Gradle. Cada módulo del proyecto es una carpeta (glass, mobile, tv y wear en la **ilustración 21**) y dentro de cada módulo contiene los siguientes cuatro elementos:

- **java/** - Código fuente Java para cada módulo.
- **manifests/** - Manifest para cada módulo.
- **res/** - Recursos del módulo.
- **Gradle Scripts/** - Ficheros Gradle y propiedades.

**Ejemplo:** La vista del proyecto en Android Studio agrupa todas las instancias de una imagen para diferentes densidades de pantalla, bajo el mismo directorio en res/ejemplo.png.

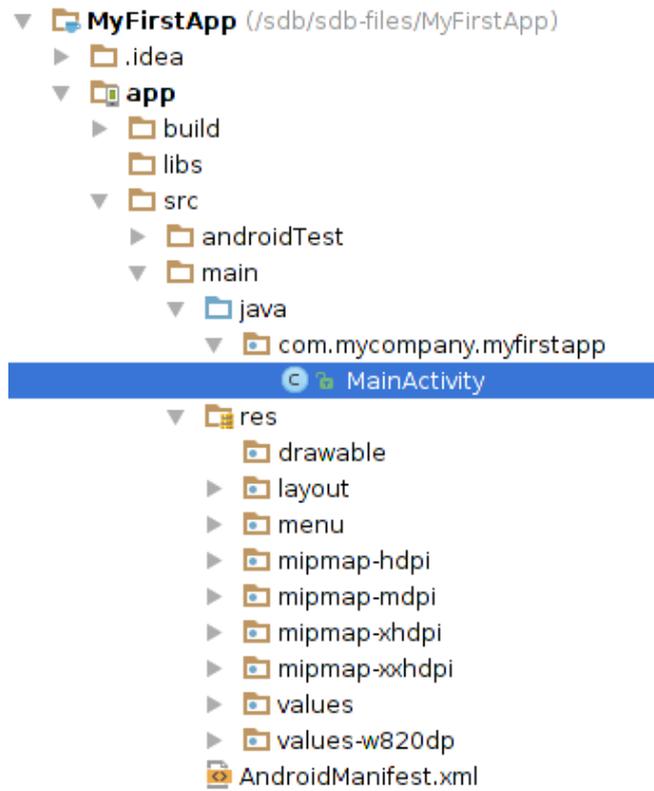


Ilustración 22. Estructura de un proyecto.

### 7.2.2 Android build system

El sistema de construcción de Android es un kit de herramientas que nos permite compilar, probar, ejecutar y empaquetar las aplicaciones. Este sistema sustituye al sistema Ant que hace uso el ADT Eclipse.

Con este sistema podemos:

- Personalizar, configurar y extender los procesos de construcción.
- Crear múltiples APKs para una aplicación, con características diferentes usando los mismos módulos y el mismo proyecto. Gracias a esto podemos reutilizar y reusar código.

### 7.2.3 Depuración y rendimiento

Android Studio nos proporciona un gran número de mejoras para asistirnos en la depuración y ayudar a mejorar el rendimiento de nuestro código, incluyendo un asistente para la emulación de dispositivos móviles, depuración línea a línea y herramientas de análisis del rendimiento.

## 7.2.4 Android Virtual Device (AVD) manager



El AVD de Android Studio pone a nuestro servicio una máquina virtual que simula un dispositivo Android. El AVD manager nos viene con toda la gama Nexus de Google, incluyendo las tablets pero además podremos configurar nuestro dispositivo como nosotros queramos especificando sus propiedades. Podremos elegir cuanta memoria RAM tendrá, establecer las dimensiones del dispositivo y su resolución y muchas más configuraciones.

El único requerimiento está en el procesador de nuestro ordenador. Deberá ser capaz de soportar la aceleración hardware de Intel, puesto que Android Studio instala el Intel® x86 Hardware Accelerated Execution Manager (HAXM). Sin él no podremos ejecutar los emuladores.

Por último, haremos uso del SDK de Android para descargar e instalar las diferentes APIs de Google así como todas las versiones de Adroid que necesitemos.

## 7.2.5 Definiendo layouts

Android Studio nos permite trabajar con los layout de nuestra aplicación en modo diseño como vamos a ver, o en modo texto, editando directamente el XML del layout.

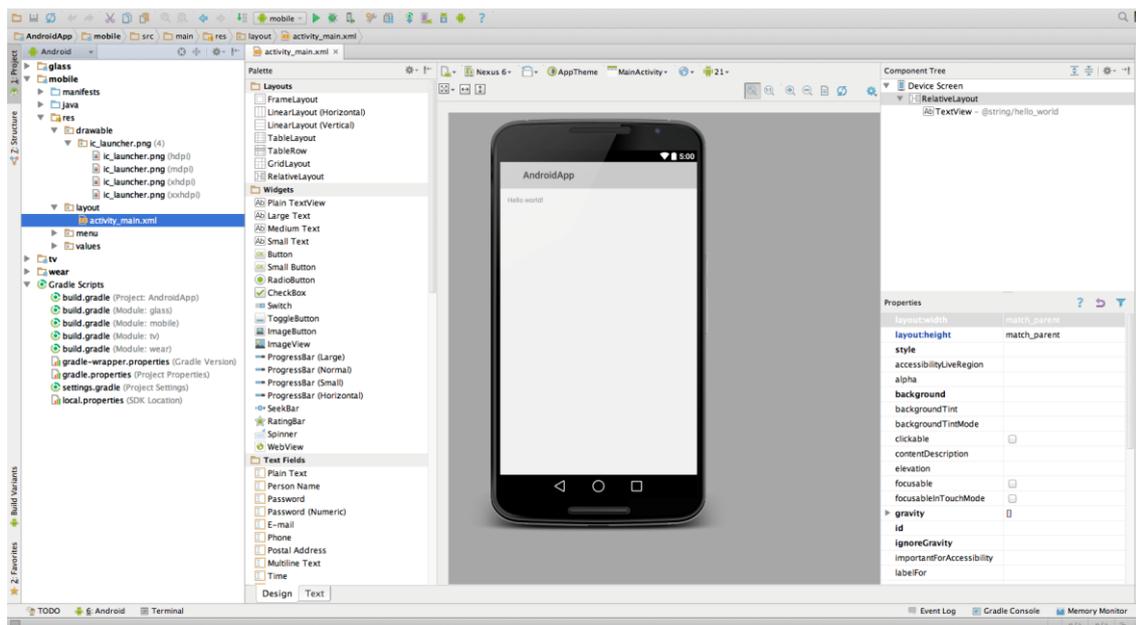


Ilustración 23. Modo diseño.

El modo diseño de Android Studio nos permite realizar cambios en la interfaz arrastrando los componentes de la lista a la misma pantalla. Además Android Studio nos provee de una ventana “preview”, que nos permite visualizar como quedaría nuestro layout en diferentes tamaños, desidad de pixeles incluso en diferentes versiones de Android.



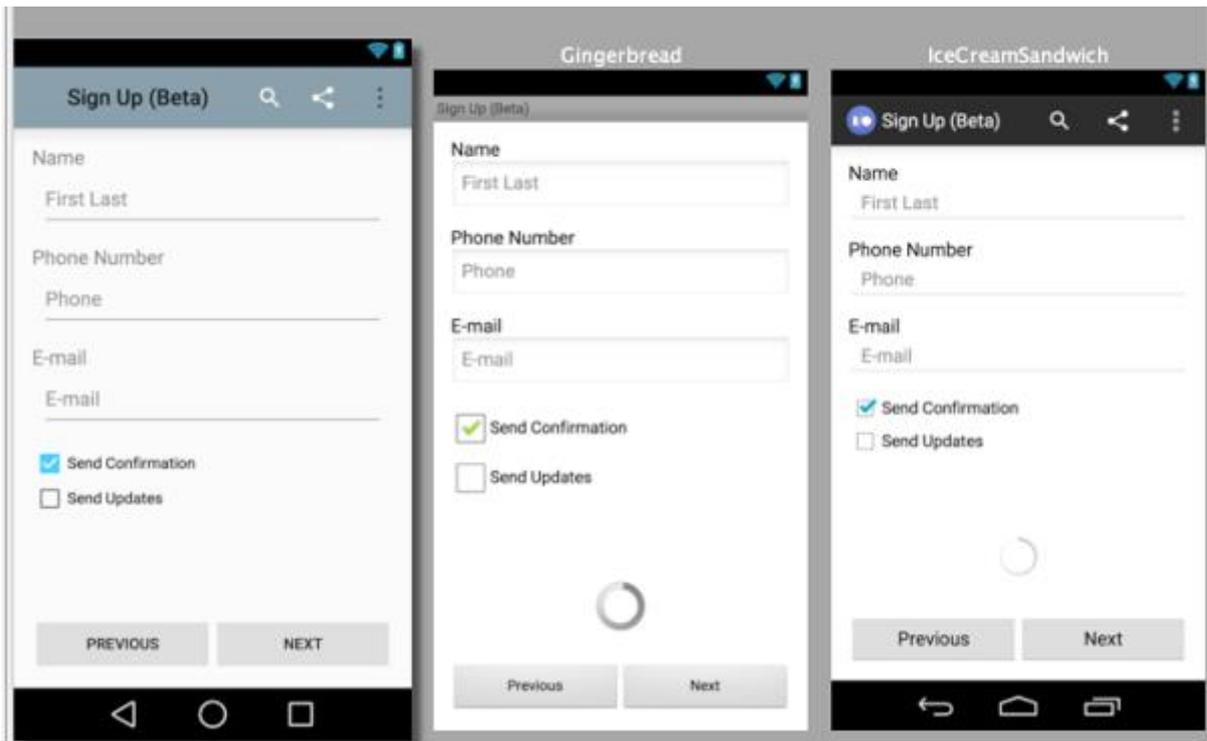


Ilustración 24. Multi-API version rendering.

### 7.2.6 Android debug monitor



Si queremos depurar nuestra aplicación Android Studio nos provee del Android Debug Monitor.

Tendremos a nuestra disposición una serie de herramientas para controlar los comportamientos de nuestros emuladores, así como ejecutar nuestra aplicación en modo debug, haciendo uso de breakpoints permitiéndonos hacer trazas de comportamiento.

Podremos mostrar mensajes “Log” en el logcat, una “consola” donde también podremos ver las excepciones que da nuestra aplicación.

### 7.3 Genymotion + virtualbox

Genymotion es un emulador de Android que aprovecha la arquitectura x86 para ejecutar de forma fluida y rápida distintos dispositivos Android. Olvidando la lentitud del emulador nativo de Android (AVD que hemos visto anteriormente) podemos ejecutar todo tipo de aplicaciones y juegos en las plataformas Windows, Mac o Linux.

Uno de los principales usos de Genymotion es facilitar el desarrollo de aplicaciones Android. Se integran perfectamente con el Android Studio, Eclipse y todas las herramientas de depuración que hemos visto. En el caso de Android Studio, tenemos un plugin de Genymotion que nos permite lanzar en apenas un par de segundos las aplicaciones desde el IDE para testearlas. Esto nos ahorra mucho tiempo.

Genymotion está basado en el uso de máquinas virtuales x86 optimizadas para correr sobre Virtualbox. En nuestra opinión la diferencia entre Genymotion y el AVD es notable.

Genymotion, como AVD, nos permite crear máquinas virtuales con diferentes características, resoluciones de pantalla y configuraciones Android (2.3, 4.1, 4.2, 4.3, 4.4 y 5), además viene por defecto con una serie de dispositivos, Nexus 4, Nexus 5, Nexus 7, Galaxy Nexus, HTC One, Moto X, Galaxy S4, Xperia Z, entre otros.

En cuanto a lo que nos ofrece Genymotion, se encuentra el uso de nuestra conexión a internet, la simulación ubicaciones GPS con el widgets que facilita la búsqueda y el posicionamiento sobre un mapa, simulación de la cámara, estado de la batería, rotación del dispositivo, etc.

El problema más significativo de Genymotion es que no incluye por defecto las aplicaciones de Google, por motivos de licencia. Actualmente se está trabajando para conseguir un acuerdo con Google. Esto limita bastante su funcionalidad, puesto que no podremos trabajar ni con Google Maps, ni con Gmail, y tampoco con los Google Play Services.



## 7.4 GitHub

GitHub es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.



Ilustración 25. Git.

GitHub aloja tu repositorio de código y te brinda herramientas muy útiles para el trabajo en equipo, dentro de un proyecto.

Además de eso, puedes contribuir a mejorar el software de los demás. Para poder alcanzar esta meta, GitHub provee de funcionalidades para hacer un fork y solicitar pulls.

Realizar un fork es simplemente clonar un repositorio ajeno (genera una copia en tu cuenta), para eliminar algún bug o modificar cosas de él. Una vez realizadas tus modificaciones puedes enviar un pull al dueño del proyecto. Éste podrá analizar los cambios que has realizado fácilmente, y si considera interesante tu contribución, adjuntarlo con el repositorio original.

Es un controlador de versiones muy potente y muy útil para los equipos de trabajo. Además Android Studio también nos facilita la sincronización de nuestros proyectos con GitHub. Podremos bajarnos nuestro repositorio con el comando “update”, que actualizará nuestro proyecto local con el código del repositorio en la nube y en caso de existir problemas (dos usuarios han cambiado las mismas líneas de código), nos dará la posibilidad de hacer un “merge” y fusionar ambos códigos como queramos. Una vez hemos realizado modificaciones, podremos subir al repositorio de GitHub nuestros cambios mediante los comandos “commit” y “push”.

GitHub también nos da la posibilidad de crear “branches” (ramas), de esta forma podremos alojar diferentes módulos de nuestro proyecto en diferentes ramas, facilitando así tanto el desarrollo como el mantenimiento y el testing de cada módulo por separado.

Podremos subir a GitHub tantos proyectos como queramos, eso sí, serán públicos y visibles para todos los usuarios. Si queremos tener proyectos privados, deberemos adquirir una cuenta premium.

## 7.5 AdMob



**Ilustración 26. Google Admob.**

Puesto que hemos desarrollado este proyecto dentro del contexto emprendedor, queremos sacarle un rendimiento económico. En principio nuestra aplicación seguirá un modelo Free, el cual si tiene la suficiente adopción evolucionará hacia un modelo Freemium, en el cual los usuarios pagarán por ciertas funcionalidades. Con el modelo Free la descarga es totalmente gratuita, de esta forma utilizaremos AdMob para sacarle rentabilidad a nuestro proyecto.

AdMob es la plataforma publicitaria más usada y reconocida por los programadores de aplicaciones. Más de 650.000 aplicaciones usan AdMob que ha pagado más de un millón de dólares.

AdMob reúne la tecnología más avanzada en una sola plataforma para que puedas obtener datos sobre tus usuarios, generar más compras en la aplicación y aumentar tus ingresos publicitarios. Con estas herramientas que nos proporciona AdMob nos olvidamos de utilizar recursos de programación costosos o aplicaciones externas para monitorizar y ver que gusta y que disgusta a nuestros usuarios.

### 7.5.1 ¿Por qué admob?

Google lleva más de una década ayudando a millones de desarrolladores a mejorar sus negocios digitales. Entre las características más importantes:

- **Compatibilidad con varias plataformas:** AdMob puede usarse con las principales plataformas, Android y iOS y, además, admite los motores de juego más importantes, como Unity y Cocos2d.
- **Actualizaciones automáticas en Google Play:** La integración de AdMob con los servicios de Google Play incorpora mejoras automáticas para las aplicaciones de Android sin cambios adicionales en el SDK.
- **Recibe pagos rápidamente:** “Recibe pagos en tu moneda local rápidamente y de manera fiable. En AdMob no aplicamos tarifas de transferencia”, es lo que nos dicen desde la página de AdMob aunque todavía no lo hemos comprobado.
- **Fácil y gratuito:** El SDK puede instalarse muy rápidamente y totalmente gratuito.



En la búsqueda de entender a nuestros usuarios, AdMob nos proporciona:

- **Analytics para aplicaciones:** Nos permite analizar el rendimiento de las aplicaciones desde AdMob con Google Analytics. Nos permite conocer en tiempo real, desde dónde se descarga la aplicación, y que funciones usan los usuarios con más frecuencia.
- **Informes de visualización del flujo de tráfico:** Con Analytics, comprobamos cuántos usuarios navegan por la aplicación con los informes gráficos de flujo de tráfico. Podemos ver qué ruta siguen los usuarios para realizar una compra y en qué momento salen de la aplicación, entre otros datos.

## 7.6 Developer Console

La consola de desarrollo de Google nos permite manejar todo lo relativo a operaciones y herramientas para la publicación de nuestra aplicación. Desde ella podremos subir aplicaciones a Play Store, configurar los precios y la distribución desde cualquier navegador.

Para ello primero deberemos registrarnos como desarrolladores de Google, y una vez nos manden el correo de verificación, podremos acceder a la Google Play Developer Console. Para registrarnos como desarrolladores deberemos abonar una cantidad anual de 25 dólares.

En la ventana All Applications encontraremos una lista con nuestras aplicaciones, tanto si están como si no están subidas a Google Play. Haciendo click sobre una aplicación podremos ver detalles del producto.

Desde la consola podremos desarrollar también la página principal que veremos en Play Store. Esta será la página que vean los usuarios en su dispositivo móvil y desde donde podrán obtener la información de la aplicación y descargarla.

Subir capturas de pantalla, videos o alguna decoración especial puede resultar atractivo para los usuarios. Es conveniente dar una buena descripción, comentar los cambios de las nuevas versiones y mantener informado al usuario de todo. Todos estos datos pueden ser actualizados en cualquier momento.

## 8 Parte Front-End

### 8.1 Introducción

Aquí llegamos a la parte dividida de la memoria, en la que mi compañero Carlos hará la parte del Back-End y dónde yo me dedicaré al Front-End de la aplicación.

Antes de entrar más profundamente en la materia, veamos que es el Front-End y el Back-End de una aplicación. Según Wikipedia:

*En diseño de software el front-end es la parte del software que interactúa con el o los usuarios y el back-end es la parte que procesa la entrada desde el front-end. La separación del sistema en front-ends y back-ends es un tipo de abstracción que ayuda a mantener las diferentes partes del sistema separadas. La idea general es que el front-end sea el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas, y los transforma ajustándolos a las especificaciones que demanda el back-end para poder procesarlos, devolviendo generalmente una respuesta que el front-end recibe y expone al usuario de una forma entendible para este. La conexión del front-end y el back-end es un tipo de interfaz.*

Por tanto, entendemos que el Front-End es la parte que interactúa con el o los usuarios. Una vez que tenemos los términos claros, podemos avanzar en este punto de la memoria.

En ésta parte vamos a ver todos los componentes de la aplicación que son visibles para el jugador. Elementos de la interfaz de usuario, interacciones tanto con Google+ como con los propios componentes de la aplicación, movimientos de las cartas, y diálogos entre muchos otros.

Comenzaremos viendo la navegación entre las diferentes pantallas de la aplicación, y describiendo la funcionalidad de cada una de ellas, así como la forma en la que nos moveremos de una a otra. Explicaremos también cómo se comportan las pantallas ante un cambio en el ciclo de vida de la app.

Posteriormente haremos un análisis de los objetos contenedores o widgets que se usan en cada una de las ventanas, y comentaremos también por qué se ha decidido usar ese diseño. En este apartado, tendremos muy en cuenta la usabilidad y explicaremos las diferentes librerías de las que hemos hecho uso. Asimismo veremos cómo interactuaremos con estos componentes, por ejemplo el efecto “drag & drop” de las cartas.



## 8.2 Pantallas de la aplicación

La aplicación cuenta con un total de 12 pantallas y son las siguientes:

- Pantalla de cargando.
- Pantalla log-in.
- Menú principal.
- Sala de espera de jugadores.
- Pantalla de juego uno contra uno.
- Pantalla de juego dos contra dos.
- Pantalla de partida finalizada (Ganada).
- Pantalla de partida finalizada (Perdida).
- Pantalla para invitar a amigos.
- Pantalla para ver invitaciones.
- Ranking de jugadores.
- Logros del jugador.

### 8.2.1 Pantalla de cargando

La pantalla de cargando está compuesta por tres elementos:

1. Un ImageView con una carta como fondo.
2. Un TextView “Cargando...”.
3. Otro TextView que mostrará una frase.

El ImageView es el componente que utiliza Android para representar imágenes y el TextView es su análogo si hablamos de textos.



Ilustración 28. Pantalla de cargando 1.



Ilustración 27. Pantalla de cargando 2.

Como podemos observar en las **ilustraciones 28 y 29**, tanto el ImageView que representa una carta, como el TextView “Cargando” están animados.

En cuanto a la carta, hemos utilizado la clase Animation de Android para hacerla girar sobre sí misma, de esta forma por cada giro de 360 grados, la carta que muestra será elegida de entre los ases de la baraja española, repitiendo la misma secuencia. Por otro lado, los puntos suspensivos tras el TextView “Cargando” suben y bajan. Logramos este efecto gracias a una librería que veremos en el siguiente apartado.

Por último, tenemos un TextView donde, de forma totalmente aleatoria, se cargará una frase informativa. El principal motivo por el que se decidió incorporar éste elemento es su utilización en muchos de los juegos más famosos del mercado de aplicaciones de hoy en día. Proporciona descripciones breves y concisas tanto de funcionalidades del juego, como de diferentes tipos de ayudas. Así se consigue que el usuario conozca con mayor profundidad la aplicación.

### 8.2.2 Pantalla de log-in

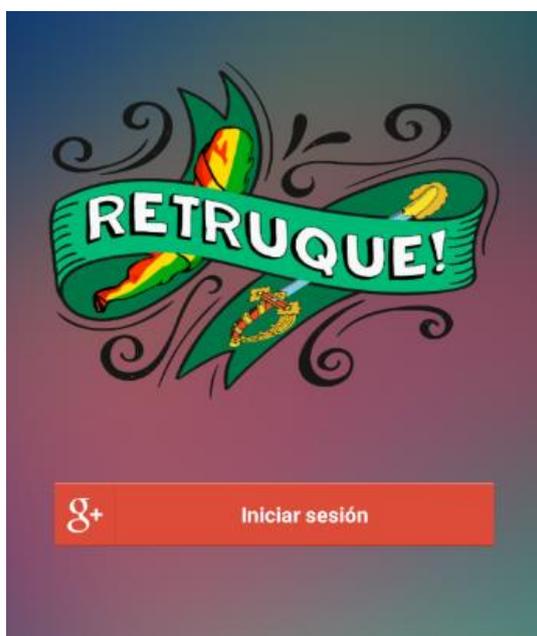


Ilustración 29. Pantalla de Log-in.

La pantalla de Log-in nos proporciona acceso a todas las funcionalidades de la aplicación.

Para iniciar sesión se eligió la red social Google+ como punto común de entrada, así se controlan los usuarios, mediando sus perfiles de Google. Sin una cuenta de Google es imposible acceder al juego. Por este motivo gracias a la API de Google, en caso de no tener un perfil de Google creado, el usuario podrá crearlo justo antes de iniciar sesión.

La pantalla está compuesta por un ImageView con el logo de la app, y justo debajo el botón que permite iniciar sesión. Este último es un componente de la librería de Google, llamado SignInButton.

Tan solo se mostrará esta pantalla en dos ocasiones. La primera será cuando el usuario arranque el juego e inicie sesión por primera vez tras su descarga. La segunda ocasión cuando el usuario pulse el botón desconectar, botón que veremos en la próxima pantalla de juego. En todos los demás accesos, Google recordará al jugador, que se autenticará, y saltará directamente a la siguiente pantalla, el menú principal.

### 8.2.3 Pantalla Menu principal

La pantalla del menú principal encapsula toda la funcionalidad de “Retruque!”, proporcionando un punto de acceso a todos y cada uno de sus casos de uso. El menú sigue un patrón de diseño llamado *Dashboard*. El patrón cuenta con las siguientes características:

1. Acceso rápido a las funcionalidades principales o de mayor importancia.
2. Acceso inmediato a cuentas o perfiles.
3. Se muestra a pantalla completa.

En el uso del patrón *Dashboard*, se recomienda no superar un límite de entre 6 y 7 elementos, pues el mostrarle demasiada información al usuario puede ocasionar frustración o un no buen entendimiento de las funcionalidades.



Ilustración 30. Menú principal.

Además, ha de tener un aspecto llamativo, capaz de atraer la atención de los usuarios, que son los que han de sentirse a gusto en su interacción.

En cuanto a lo que botones se refiere, tenemos dos tipos de botones, los botones de menú y el botón de desconectarse.

- **Botón de menú:** Los seis botones tienen una animación de pulsado y para ello hemos utilizado el atributo “Selector” de la clase “Button”. “Selector” nos permite leer en qué estado se encuentra el botón (pressed, hovered, focussed y default) y en función de éste asignarle la imagen que nosotros queramos. Así pues, tendremos una imagen del botón con sombra, que representará al botón sin pulsar y otra imagen del mismo, pero esta vez sin la sombra y representando al botón pulsado. De esta forma conseguimos el efecto de pulsado, transitando con “Selector” entre los estados y fondos de botón.
- **Botón de desconectarse:** Para este botón utilizamos una librería llamada “FButton” y que describiremos en el siguiente apartado.

En cuanto a la puesta en práctica, se ha utilizado un `LinearLayout` con orientación vertical. Éste nos permite distribuir los componentes uno debajo de otro. También emplearemos el `LinearLayout` con orientación horizontal, que distribuye los widgets de forma horizontal, uno al lado del otro. De esta manera tenemos la siguiente jerarquía:

- `LinearLayout` vertical (Contenedor)
  - Logo Retruque!
  - `LinearLayout` horizontal
    - Botón uno contra uno
    - Botón dos contra dos
      - `LinearLayout` horizontal
  - Botón invitar jugadores
  - Botón ver invitaciones
- `LinearLayout` horizontal
  - Botón ranking
  - Botón logros
- Boton desconectar

Para distribuir los componentes de la forma en la que vemos en la **ilustración 30**, hacemos uso del atributo de las vistas “layout\_weight”, que nos permite dividir proporcionalmente el espacio de la pantalla entre las vistas. Tenemos entonces el logo ocupando un 25% del espacio, los LinearLayout horizontales con un 20% cada una, y el botón de desconectar con un 15%. Sus atributos “layout\_weight” serán 2.5, 2 y 1,5 respectivamente, pues Android calcula el total de la suma y les asigna a cada widget su correspondiente espacio.

Por último, usaremos el atributo “layout\_margin” para el espaciado de los componentes. Para indicar cuánto margen queremos dejar, utilizaremos la medida que nos propone Android, “densidad de píxeles (dp)”. Con la gran diversidad de tamaños y resoluciones de pantalla (Problema de fragmentación visto anteriormente), no podemos utilizar píxeles como unidad métrica ya que cada pantalla tiene un número de píxeles diferente. En su lugar usaremos los dp, que en función de los píxeles por pulgada (ppi, Pixels Per Inch), el margen será mayor o menor.

#### 8.2.4 Pantalla de espera de los jugadores

Cuando los usuarios vayan a empezar una partida (pulsando los botones de uno contra uno, o dos contra dos), se lanzará una nueva Actividad que mostrará una sala de espera. Se hará una llamada a “StartActivityForResult”, entonces esta petición será atendida por los “Google Play Game Services” que nos mostrará su sala de espera, dejando nuestra Actividad suspendida en segundo plano.

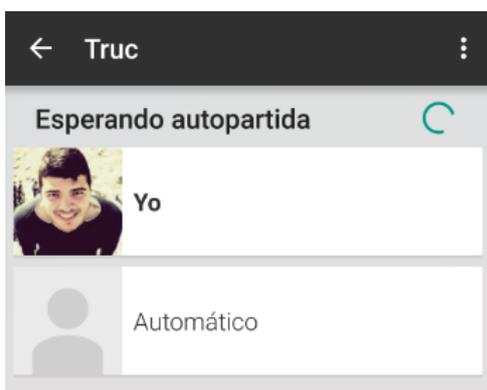


Ilustración 31. Sala de espera.

Tenemos entonces, nuestro proceso principal suspendido en segundo plano (Actividad donde hemos pulsado el botón de iniciar partida), que tras pasar por “onPause” esperará en “onActivityResult” el resultado de esta nueva Actividad lanzada (Sala de espera).

Cuando encontremos un oponente, la Actividad comunicará el resultado de la petición y se destruirá, reactivándose el proceso que dejamos en espera. De este modo desde “onActivityResult” podremos ir a la pantalla de juego y empezar la partida.

La interfaz es impuesta por Google.

#### 8.2.5 Pantalla de uno contra uno

Las pantallas de juego son los layout más completos de toda la aplicación, además de ser los que más contenedores tienen. En esta pantalla se desarrolla toda actividad relacionada con el juego, así como la interacción entre jugadores.



Ilustración 32. Pantalla de uno contra uno.

A continuación analizaremos los componentes de esta pantalla con mayor profundidad, y también los identificaremos en la **ilustración 32**.

**FabToolbar:** Círculo azul que se encuentra en la parte inferior derecha de la pantalla, con un “+” en su interior. Al pulsar el widget se desplegará nuestro menú de acciones. FabToolbar corresponde a una librería que veremos con más detenimiento en el siguiente apartado.

**CircularProgressButton:** Rectángulo con los bordes redondeados de color azul y que contiene números en su interior. Éstos son las puntuaciones de los jugadores, así pues, el CircularProgressButton actúa como marcador. También es una librería adicional que veremos en el apartado “Librerías”.

**ImageView de la mano:** Se encuentra en la parte inferior de la pantalla y representa una mano. Con objeto de hacer el juego lo más parecido a la realidad, y a fin de que no estén las cartas flotando, decidimos utilizar una mano para enriquecer la experiencia del usuario.

Aunque parezca tan solo una imagen, realmente hay dos, una imagen del dedo y otra de la mano. Esto es debido a que las cartas tienen que estar por encima de la mano, pero por debajo del pulgar.

**ImageView de las cartas:** Cada carta que vemos es un ImageView diferente. Las cartas utilizarán el efecto “drag and drop”, es decir, arrastraremos la carta y la soltaremos para lanzarla. Para esto, usaremos el evento “touch” que a diferencia del evento “click” que tan solo registra una pulsación, empieza en el momento en el que el usuario pulsa con el dedo sobre la pantalla y finaliza cuando se levanta, de esta forma haremos que las cartas se muevan sobre la pantalla siguiendo la estela del dedo del usuario. Para conseguirlo suscribimos los ImageView a un “onTouchListener”, de forma que cada vez que se registre un evento “touch” sobre las cartas, actualizaremos su posición con las nuevas coordenadas proporcionadas por el evento. Esta animación complementaria el “drag” (arrastrar), en cuanto al “drop” (soltar), utilizamos la clase Animation de nuevo. Entonces, si soltamos la carta por encima de la mitad de la pantalla, la carta se lanzará a la mesa, si por el contrario lo hacemos por debajo, la carta volverá a la mano. Para saber dónde debemos posicionar la carta, calcularemos la mitad de la pantalla con “mitadPantalla = carta.getLayoutParams().height / 2”. La función “getLayoutParams” nos devuelve las características del layout que contiene la carta, en este caso el layout ocupará toda la pantalla. Una vez calculado éste punto, cada vez que soltemos una carta se activará la función “MotionEvent.ACTION\_UP” que sobrescribiremos añadiendo una comprobación: si la cordenada Y de la carta lanzada se encuentra por debajo del “mitadPantalla” calculado anteriormente, devolveremos la carta a la mano, por otro lado si está por encima animaremos la carta hacia la mesa.

**LinearLayout verticales:** Rectángulos blanco y verde con la información de los jugadores. Estos elementos están compuestos por un “CircleImageView” con la foto de perfil de cada jugador, seguido de un “TextView” con el nombre del jugador y por último una “ProgressBar”, una barra de progreso que indica el tiempo que nos queda de turno.

- **“CircleImageView”:** Librería que nos creará un ImageView circular y que nos vendrá perfecto para las imágenes de perfil, la veremos en el siguiente apartado. Para descargar la imagen y cargarla en la aplicación utilizaremos un “AsyncTask” para que lo haga en segundo plano.
- **“ProgressBar”:** Utilizaremos estas barras de progreso para indicar el tiempo restante del turno de cada jugador. Puesto que se trata de un juego multijugador online, no podemos permitir que un jugador espere eternamente a que al otro le apetezca jugar, de esta forma hemos decidido limitar el turno a 40 segundos. Para hacer la cuenta atrás usamos la clase “CountDownTimer”. Cuando creamos este objeto le pasamos los segundos totales y la diferencia entre cada “tick”, de esta forma con “new CountDownTimer(40000, 1000)” crearemos una cuenta atrás de 40 segundos, y a cada segundo sobrescribiremos el método de la clase “onTick” para que actualice la “ProgressBar”. Aprovecharemos cada vez que iniciemos una barra de progreso para iluminar en verde el LinearLayout que la contiene, de esta forma indicaremos que jugador tiene el turno.

**Diálogos de alerta:** Para la interacción entre jugadores, cuando algún jugador realice una acción, al rival se le mostrará un diálogo de alerta para que conteste y pueda seguir el hilo del juego. También los utilizaremos para dar información al jugador, quien ha ganado, si hay empate etc. Para estos diálogos hemos utilizado una librería llamada “MaterialDialogs” de la que también hablaremos más adelante.

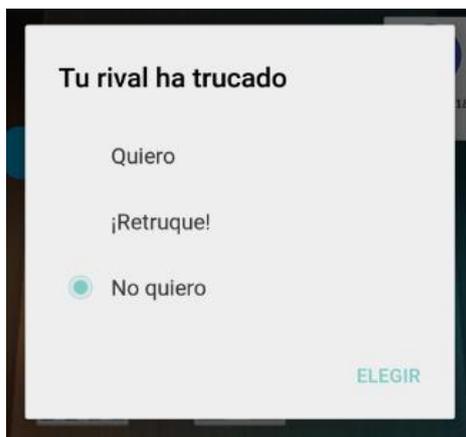


Ilustración 33. Dialogo de juego.

**Bocadillos animados:** A fin de mejorar la experiencia de usuario, cuando el jugador rival nos conteste a una acción se nos notificará mediante un bocadillo, que no es más que un “TextView”.

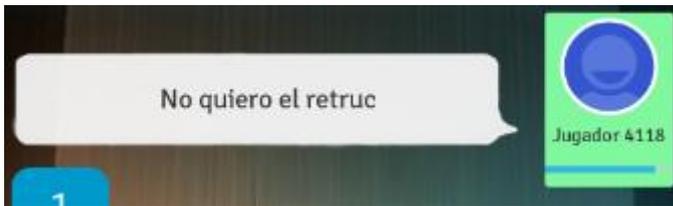


Ilustración 34. Bocadillo de juego.

### 8.2.6 Pantalla de dos contra dos



Ilustración 35. Pantalla de dos contra dos.

Si nos fijamos, en la pantalla de dos contra dos las diferencias son mínimas. En vez de contar con dos `LinearLayout`, al tratarse de una partida dónde participan cuatro jugadores, tendremos cuatro `LinearLayout`s.

En cuanto al hilo del juego, se siguen las mismas directrices que para el de dos jugadores, solo que ésta vez el juego es en equipo. El jugador que tenemos arriba será nuestro compañero, por ejemplo en la **ilustración 35**, Guillermo será compañero de Carlos Alba.

La diferencia más significativa con respecto al modo de dos jugadores, es el sistema de señas. En este modo cobra un especial sentido el uso de señas, de esta forma podemos comunicarnos con nuestro compañero y diseñar una estrategia. Cada jugador podrá hacer un total de dos señas como máximo o podrá no enviar ninguna seña si así lo prefiere. Estas señas tan solo las verá el propio compañero, pero si el rival es lo suficientemente hábil, podrá descubrir una de las señas, obteniendo así ventaja tal y como pasaría en una partida real. No obstante, cabe

destacar que es muy complicado pillar una seña debido a que el dialogo aparece durante un poco menos de medio segundo. No podría ser de otra forma, ya que permitir descubrir señas con facilidad restaría dificultad al juego.



Ilustración 36. Señas del compañero.

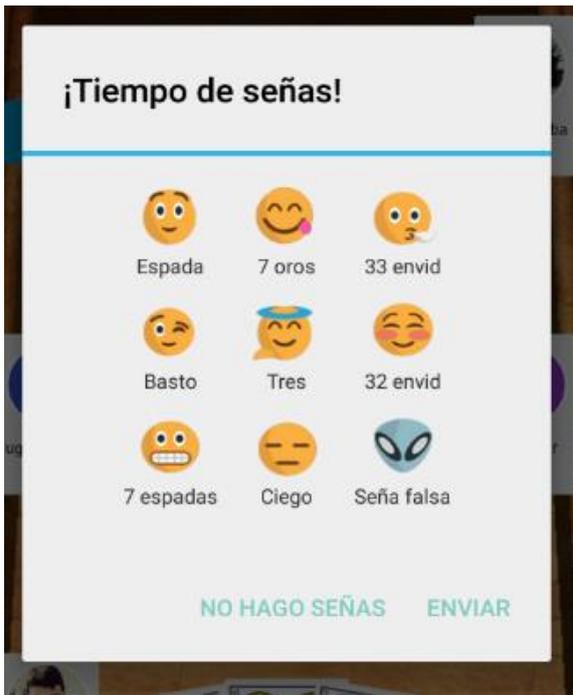


Ilustración 37. Diálogo de señas.

En la **ilustración 37** podemos observar el diálogo que nos permite escoger qué seña queremos realizar. Nos aparecerá en el inicio de cada mano, y está compuesto por: el título, “¡Tiempo de señas!”, un “ProgressBar” que supone una cuenta atrás de veinte segundos, y los botones “No hago señas” y “Enviar”. Nos aparecerá un tic en cada seña que marquemos.

Al acabar el tiempo de señas, empezaremos la partida, pero antes se nos mostrarán las señas que nos haya hecho nuestro compañero, en caso de que haya hecho alguno. Véase la **ilustración 36**.

### 8.2.7 Pantalla de partida finalizada

Cuando finalizamos una partida, podemos obtener dos resultados, partida ganada o partida perdida.



Ilustración 38. Victoria.

Ilustración 39. Derrota.

Tanto si ganamos como si perdemos debido a una desconexión o abandono de nuestro oponente, se nos informará en ésta pantalla. Véase la **ilustración 38**.

El botón de volver al menú principal es un “FButton”. Para futuras versiones, esta pantalla ofrecerá más información, por ejemplo el resultado final.

Cuando se finalice la partida y nos aparezca esta pantalla por primera vez, el botón de volver al menú principal no nos aparecerá. Esto es debido a que ésta es la pantalla elegida para mostrar la publicidad y queremos que los usuarios la vean. Una vez la hayan cerrado habilitaremos el “FButton” para volver al menú principal.

Para la publicidad haremos uso de la clase “InterstitialAd” y también la clase “AdRequest”. Con “AdRequest” solicitaremos a “Admob” de Google un anuncio que cargaremos en nuestro “InterstitialAd”. Un Interstitial es un anuncio a pantalla completa, que mediante la página web de “Admob” podremos configurar. En nuestro caso, hemos optado por reproducir un video que podrá cerrarse a los 5 segundos.

### 8.2.8 Pantalla para invitar

En lo que respecta a la parte social de la aplicación, las pantallas que vamos a ver a continuación son las más importantes. Puesto que utilizamos Google+ para registrarnos en Retruque!, podremos invitar a gente de nuestros círculos en la red social, familia, amigos, etc. Pertener a nuestros círculos en Google+ significa que entre los usuarios hay algún tipo de relación.

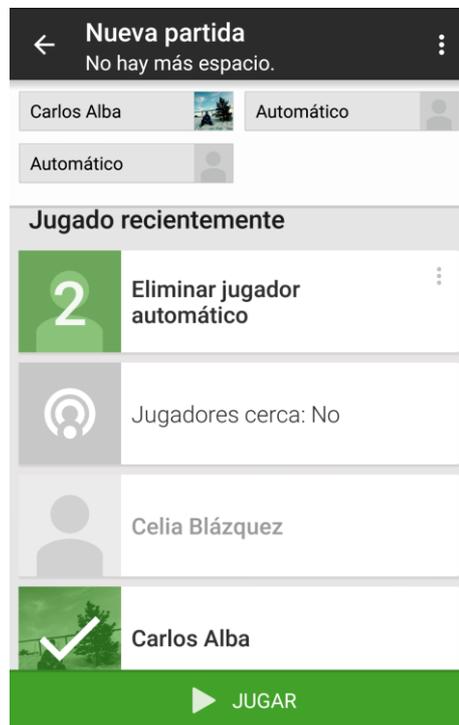


Ilustración 40. Invitar a amigos de Google+.

Cuando pulsemos el botón de invitar a jugadores, lanzaremos un “starActivityForResult”, tal como lo hacíamos en las partidas de dos y cuatro jugadores, para iniciar una nueva Actividad de Google. El resultado será una sala de espera o habitación, configurada con los jugadores invitados.

La pantalla nos mostrará una lista de usuarios a los que mandar nuestra invitación. Primero veremos con quien hemos jugado recientemente y tras ellos, jugadores de nuestros círculos. Podremos seleccionar uno, o tres jugadores. Si seleccionamos uno, jugaremos una partida uno contra uno contra él, si por el contrario seleccionamos tres, jugaremos una partida dos contra dos.

Podremos añadir tanto jugadores de nuestro entorno, como jugadores automáticos. De modo que si invitamos a una persona de nuestros círculos y dos personas aleatorias, nuestro compañero de equipo será nuestro amigo de Google+. En caso de invitar a dos amigos y un tercer jugador automático, el usuario que haya realizado la invitación, siempre tendrá un compañero conocido, siendo aleatoria la elección de éste de entre los jugadores invitados. Por último, en el caso en el que se inviten a tres jugadores conocidos, la correspondencia o “matching” se hará de forma aleatoria.

Para finalizar, tras pulsar el botón de “Jugar”, se creará la sala de espera configurada con los jugadores previamente invitados. Tras aceptar las invitaciones, comenzará la partida.

### 8.2.9 Pantalla de invitaciones

En esta pantalla podremos consultar si algún amigo nos ha mandado una invitación y podremos aceptarla, lo que nos llevaría a una sala de espera con la partida que haya configurado el usuario que nos ha invitado, o rechazarla.

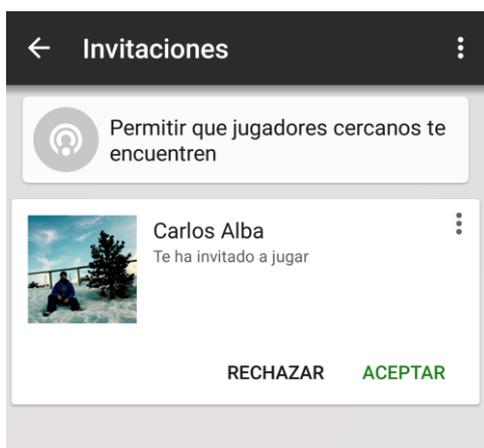


Ilustración 41. Bandeja de invitaciones.

### 8.2.10 Pantalla de Ranking

Llamaremos de nuevo a los servicios de Google para que nos muestre esta pantalla. El resultado será una lista de jugadores, encabezados por el usuario que más partidas, tanto de uno contra uno, como de dos contra dos, haya ganado.

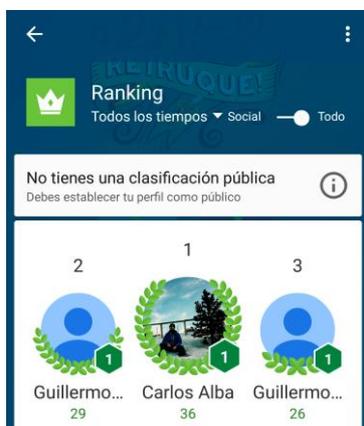


Ilustración 42. Ranking de jugadores.

### 8.2.11 Pantalla de logros

También haremos uso de los servicios de Google para consultar de qué logros dispone el juego, y cuáles de ellos hemos conseguido desbloquear. Podremos ver el porcentaje de cada uno de ellos.

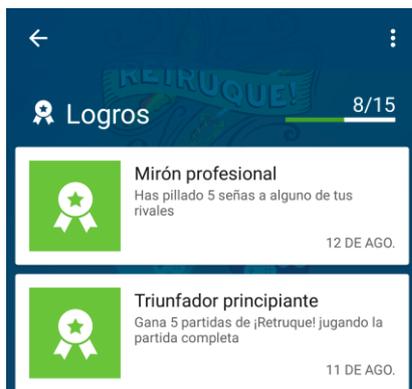
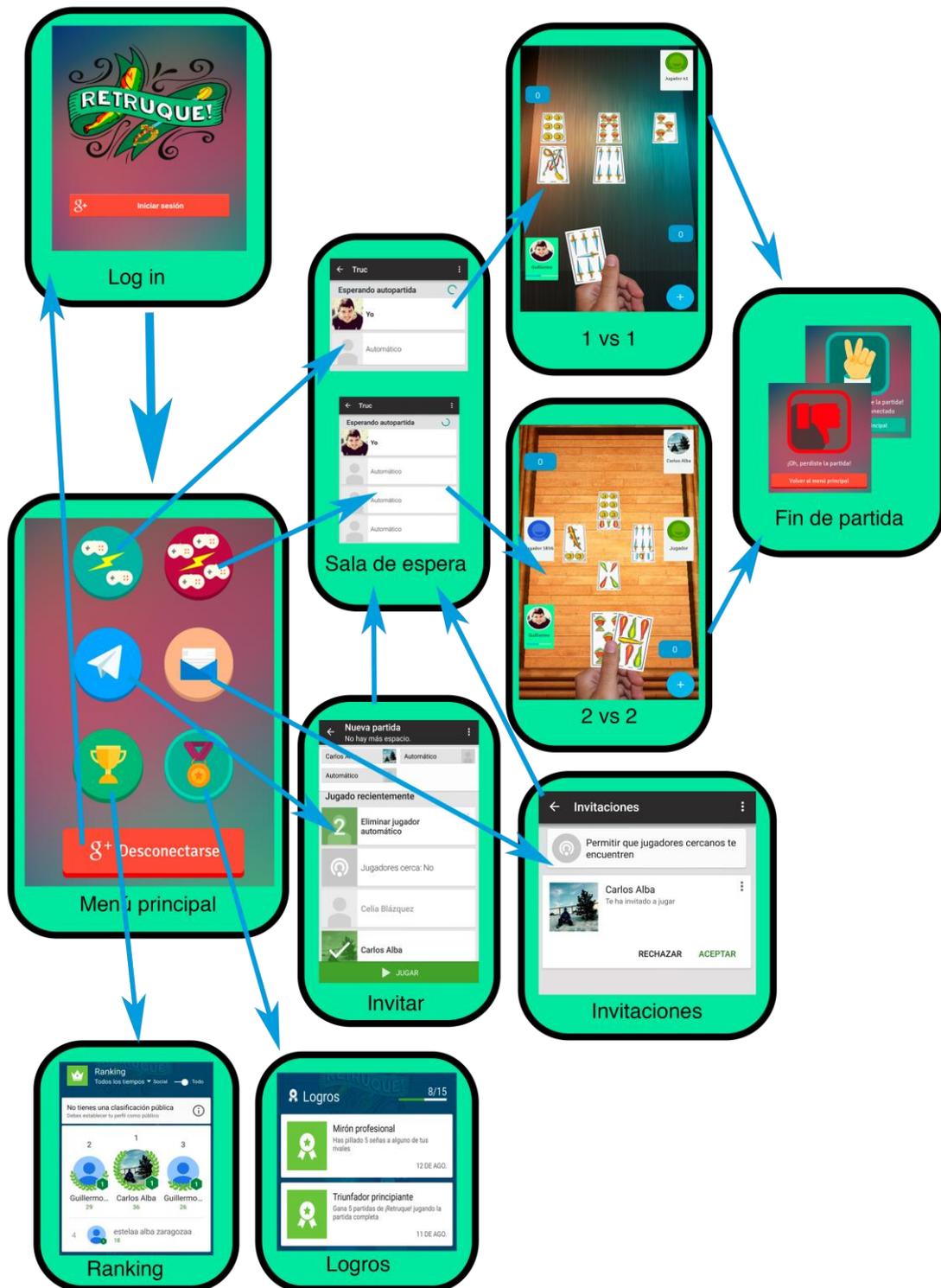


Ilustración 43. Logros del jugador.

### 8.3 Diagrama de flujo

El diagrama de interacción o navegación queda de la siguiente manera:



Una vez navegando sobre las flechas, siempre veremos la pantalla de cargando entre las diferentes vistas de nuestra aplicación. Advertir también que, aunque en el diagrama no veamos flechas de vuelta al menú principal, en todas y cada una de las vistas (representadas dentro de los rectángulos verdes) podremos volver atrás, con el botón “←” de la barra de navegación, o con el de la barra del menú.

## 8.4 Librerías

En este apartado vamos a ver algunas de las librerías con las que hemos compilado nuestro proyecto. Librerías completamente externas a Android y que nos han facilitado extremadamente la implementación de ciertas funcionalidades.

```
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:22.2.1'
    compile 'com.google.android.gms:play-services:7.5.0'
    compile 'com.google.android.gms:play-services-ads:7.5.0'
    compile project(':BaseGameUtils')
    compile 'com.afollestad:material-dialogs:0.7.4.2'
    compile 'com.github.alexkolpa:floating-action-button-toolbar:0.5.1'
    compile 'com.akexorcist:RoundCornerProgressBar:1.2.0'
    compile 'com.github.dmytrodanylyk.circular-progress-button:library:1.1.3'
    compile 'info.hoang8f:fbutton:1.0.5'
    compile 'com.orhanobut:dialogplus:1.7@aar'
    compile 'pl.tajchert:waitingdots:0.2.0'
    compile ('com.github.worker8:tourguide:1.0.12-SNAPSHOT@aar') {
        transitive=true
    }
}
```

Ilustración 44. Dependencias.

Con “Gradle” podemos añadir las librerías de la manera que aparece en la **ilustración 44**. Las cuatro primeras que observamos son la librería “AppCompat” para la compatibilidad de versiones Android, “Play Services” para los servicios de Google, “Play Services Ads” para la publicidad y “BaseGameUtils” para algunas utilidades.

Las dependencias que vamos a comentar serán las ocho siguientes.

### 8.4.1 Material dialogs

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'com.afollestad:material-dialogs:0.7.4.2'
```

Esta librería nos proporciona unos diálogos de alerta personalizados y que sigue la línea de diseño general de la aplicación. Además nos permite mostrar diálogos con un “layout” totalmente personalizado, como lo es el diálogo de las señas.



Ilustración 45. Material Dialogs.

Los diálogos funcionan con “Callbacks”, de forma que se activa la llamada al pulsar o bien el “positiveButton” (aceptar generalmente), o bien el “negativeButton” (cancelar generalmente).

Para los diálogos con selecciones (truc y envid) habrá que sobrescribir el método “onSelection” para que dependiendo de que opción se seleccione, realizar una u otra acción.

#### 8.4.2 Floating action button toolbar

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'com.github.alexkolpa:floating-action-button-toolbar:0.5.1'
```



Ilustración 46. Floating action button toolbar.

El “Floating Action Button Toolbar” va a ser nuestro menú dentro de las pantallas de juego. Desde él podremos realizar cualquier acción siempre y cuando sea nuestro turno.

Crearemos nuestro objeto FabToolbar, que haremos que aparezca y desaparezca (con una animación), dependiendo de si tenemos o no el turno.

Al pulsar sobre él se abrirá como vemos en la **ilustración 46**, y mediante la función “hide” lo cerraremos. Al abrirlo, tendremos por un lado unos botones fijos, que van a ser el botón “me voy”, el botón “frases”, el botón “tapar”, el botón “abandonar” y el botón “cerrar”. Por otro lado están los botones que, dependiendo de la etapa del juego, serán: uno de entre “truco”, “retruco”, “cuatro val” y “joc fora”, “envido” y “la falta”. Estos dos últimos solo estarán visibles durante el primer turno.

En definitiva, tenemos un menú desplegable, que no nos va a ocultar información en pantalla, usable y con un diseño adecuado para este contexto, que nos va a encapsular todas las acciones en un solo menú, proporcionándonos una interfaz muy limpia.

#### 8.4.3 Circular progressbutton

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'com.github.dmytrodanylyk.circular-progress-button:library:1.1.3'
```

El “Circular ProgressBar” va a ser nuestro marcador. Es un rectángulo con los bordes redondeados, que cada vez que se estén calculando los puntos se va a convertir en un “ProgressBar”. Esta vez, a diferencia de las barras de progreso vistas anteriormente, se trata de una barra de progreso circular que da vueltas sobre sí misma.

Utilizaremos el método “setIndeterminateProgressMode” para activar el modo “ProgressBar”, y cuando estén disponibles las puntuaciones, con “setCompleteText” actualizaremos los marcadores con ellas. Obtenemos entonces una sensación de cargando, que queremos conseguir mientras acabamos una mano y nos preparamos para la siguiente.

#### 8.4.4 FButton

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'info.hoang8f:fbutton:1.0.5'
```

El “FButton” no es más que un “Button”, funciona de la misma manera. Son botones con un estilo definido y que podemos editar y configurar para componer una interfaz más atractiva.

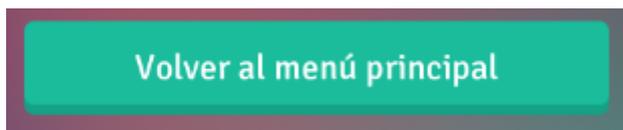


Ilustración 47. FButton.

#### 8.4.5 DialogPlus

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'com.orhanobut:dialogplus:1.7@aar'
```

Esta librería nos muestra una lista de frases que podremos enviar a nuestros oponentes y que se les mostrará en un bocadillo animado.



Cuando pulsemos el botón de “frases” nos aparecerá este menú desde abajo hacia arriba, y ocupando la pantalla. Podremos navegar por el para decidir qué decir.

Utilizaremos la clase “DialogPlus” para, mediante el patrón “Builder”, construir nuestro diálogo.

También la clase “ListHolder”, que nos proporcionará un adaptador entre la lista de frases y el diálogo mostrado.

Por último, estableceremos los siguientes disparadores: “OnItemClickListener” y “OnBackPressedListener” para responder ante los eventos de pulsado sobre una frase y pulsado de la tecla atrás, respectivamente. Cuando pulsemos sobre una frase, la enviaremos a los demás jugadores, asimismo cuando pulsemos el botón atrás, cerraremos el diálogo.

Ilustración 48. DialogPlus.

#### 8.4.6 Waiting dots

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile 'pl.tajchert:waitingdots:0.2.0'
```

Esta librería tan solo nos proporciona el componente “DotsTextView”, un “TextView” personalizado que representa unos puntos suspensivos animados.

Nos es útil mientras el usuario se encuentra en la pantalla de cargando, con la intención de dar la sensación de que la app no se ha colgado y está en proceso de carga.

#### 8.4.7 Tour Guide

Para poder utilizar esta librería, deberemos añadir la siguiente línea al fichero Gradle de nuestro proyecto:

```
compile ('com.github.worker8:tourguide:1.0.12-SNAPSHOT@aar') {transitive=true}
```

Primeramente, se decidió utilizar esta librería con objeto de informar al usuario de cierta información, que consideramos imprescindible para el buen hacer de la aplicación. Entonces implementamos un pequeño tutorial para detallar la parte social de Retruque!



Ilustración 49. Tour Guide 1.

Al iniciar por primera vez la aplicación nos aparecerá el diálogo de la **ilustración 49**. Una vez pulsemos el botón, y volvamos al menú principal, nos sorprenderá el diálogo de la **ilustración 50**. Tan solo podremos verlos la primera vez que ejecutemos la aplicación, o cuando borremos los datos persistentes de ésta y volvamos a relanzarla.

Para esto, utilizamos “SharedPreferences” que es un mecanismo ligero de almacenamiento persistente y recuperación de pares clave-valor. Cuando acabemos el tutorial guardaremos en preferencias de la aplicación que ya hemos hecho el “tour guide”, de forma que cuando volvamos a iniciar la aplicación consultaremos las preferencias y no volveremos a hacer el tutorial.



Ilustración 50. Tour Guide 2.

## 9. Publicación en play store

Llegados a este punto, en el que la aplicación ya está acabada, debemos empezar a pensar cómo compartirla con el mundo para que los usuarios de los dispositivos móviles Android puedan descargarla y disfrutar de ella. Para ello, la subiremos a Google Play Store. Veamos qué es y cuál es el procedimiento a seguir para subir nuestra App.

### 9.1 ¿Qué es play store?



Ilustración 51. Play Store.

Google Play Store (anteriormente Android Market) es una plataforma de distribución digital de aplicaciones móviles para los dispositivos con sistema operativo Android, así como una tienda en línea desarrollada y operada por Google. Esta plataforma permite a los usuarios navegar y descargar aplicaciones (desarrolladas mediante Android SDK), juegos, música, libros, revistas y películas. También se pueden adquirir dispositivos móviles como ordenadores Chromebook, teléfonos inteligentes Nexus, Google Chromecast, entre otros.

Las aplicaciones se encuentran disponibles de forma gratuita, así como también con costo. Pueden ser descargadas directamente desde un dispositivo con Android a través de la aplicación móvil Play Store.

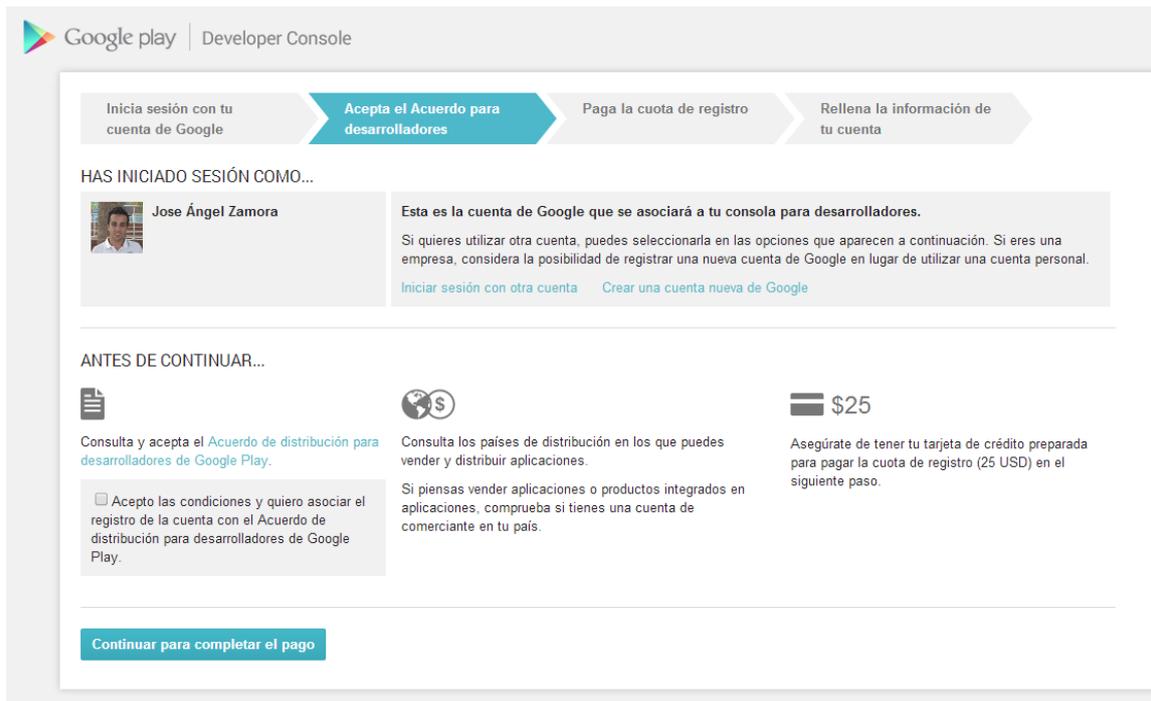
En marzo de 2012, con la fusión de Android Market con Google Music, el servicio fue renombrado a Google Play, como resultado de la nueva estrategia de distribución digital de Google. En julio de 2013, se anunció que Google Play había sobrepasado 1 millón de aplicaciones publicadas y se habían registrado más de 50 mil millones de descargas.

### 9.2 Darse de alta como desarrollador

Ahora que ya sabemos lo que es Play Store, veremos cómo podemos subir nuestra aplicación.

Lo primero que tendremos que hacer será darnos de alta como desarrolladores, para que Google nos permita subir nuestras Apps a Play Store. Este registro no es gratuito, el alta como desarrollador cuesta 25 dólares, pero este pago lo realizaremos una única vez y ya seremos desarrolladores.

Para darnos de alta como desarrolladores accederemos a Google Play Developer Console y nos logearemos con nuestra cuenta de Google. Ahí es donde deberemos realizar el pago único de los 25 dólares.



**Ilustración 52. Darse de alta como desarrollador.**

Una vez pagado, ya somos oficialmente desarrolladores de Apps y tenemos los permisos para subirlas a Play Store.

### 9.3 Cómo publicar una aplicación

Ahora que ya podemos subir nuestras aplicaciones, veamos qué debemos hacer para incorporarlas correctamente a Play Store. Subir una aplicación sin Game Services es más sencillo, pero en este apartado explicaremos cómo hacerlo incluyendo estos servicios. Tal y como hicimos para subir Retruque!

Lo primero que tenemos que hacer es crear un Servicio de juegos, que será lo que le proporcione a nuestra App el uso de los Game Services y todos sus elementos.



**Ilustración 53. Servicios de juegos en Developer Console.**

Una vez creado, podemos rellenar una ficha para el servicio, aunque esta no se verá en Play Store.

Estadísticas de jugadores Descripción general Detalles de ingresos Datos demográficos Estadísticas de funciones Descripción general Detalles de interacción Misiones <b>Información del juego</b> Aplicaciones vinculadas Eventos Logros Marcadores Testing Publicar	<b>Nombre para mostrar *</b> Inglés (Estados Unidos) – en-US	Retruque! <small>9 de 30 caracteres</small>
	<b>Descripción</b> Inglés (Estados Unidos) – en-US	Por fin el popular juego de cartas valenciano del Truc en tu móvil, como nunca antes lo habías visto. Nunca había sido tan fácil jugar una partida de Truc, estés donde estés. ¡Desafía a tus amigos con el modo multijugador en línea y demuestra que eres el mejor!  <b>Características del juego:</b> - Multijugador en línea 1 vs 1. - Multijugador en línea 2 vs 2. - Juega con amigos ingresando con tu cuenta de Google+. - Juega con gente desconocida buscando una partida. - Juega con gente cercana a tu localización. - Sistema de logros. - Ranking de victorias. - Sistema de señas para el modo 2 vs 2. <small>Finca en la noche</small> <small>1250 de 4000 caracteres</small>
	<b>Categoría</b>	Cartas
	<b>Juegos guardados</b>	<input type="checkbox"/> SÍ <input checked="" type="checkbox"/> NO
	<b>ARCHIVOS DE IMÁGENES</b> <small>Añade todos los recursos gráficos que se describen a continuación o utiliza recursos de imágenes de una de tus aplicaciones de Android.</small>	
	<b>Icono de alta resolución</b> 512x512 <small>Archivo PNG de 32 bits (alfa)</small>	<b>Imagen destacada</b> 1024x500 <small>Archivo JPG o PNG de 24 bits (no alfa)</small>
	 	

**Ilustración 54. Ficha del juego.**

Cuando tengamos la ficha rellena, tenemos que acceder a la pestaña llamada **Aplicaciones vinculadas**, en la que crearemos una nueva aplicación que hará referencia a nuestro juego.

APLICACIONES VINCULADAS <span style="float: right;"> <input type="button" value="Vincular otra aplicación"/> o <input type="button" value="Ir al siguiente paso"/> </span>			
NOMBRE	DETALLES	ID DE CLIENTE DE OAUTH 2.0	ESTADO
 Retruque!	com.game.guillermo.truc	✓	✓ Publicada

**Ilustración 55. Aplicaciones vinculadas.**

Este es el punto más importante de la creación del servicio, ya que será donde lo vinculemos con nuestro juego y si lo hacemos mal no proporcionará los servicios deseados.

Tenemos que darle un nombre (sólo es una referencia, no hace falta que sea el nombre del juego) y el nombre del paquete. Éste sí que debe ser el mismo que el paquete que tengamos en nuestro código, de lo contrario el juego no se vinculará correctamente. Además, marcaremos con un sí la opción de multijugador en tiempo real.

Una vez guardado, se nos proporcionará un código llamado ID de aplicación, que tendremos que añadir en nuestro código.



**Nombre de la aplicación**  
Inglés (Estados Unidos) – en-US  9 de 30 caracteres

---

**Nombre del paquete**

---

**Preferida para nuevos jugadores**  Para nuevas instalaciones en teléfonos Android.

---

**AJUSTES DE MULTIJUGADOR**

**Multijugador por turnos**

---

**Multijugador en tiempo real**

---

**ANTIPIRATERÍA**

**Habilitar antipiratería**

---

**AUTORIZACIÓN**

**ID de aplicación** 648419090213

---

**ID de cliente de OAuth2** 648419090213-temqaactrpc1ehhnpb9smeb72mq944d.apps.googleusercontent.com

---

Solo tienes que incluir el ID de la aplicación (648419090213) en tu aplicación Android.  
Para obtener más información al respecto, consulta la [documentación para desarrolladores](#).

Ilustración 56. Ajustes de los servicios.

```
ids.xml x
<?xml version="1.0" encoding="utf-8" ?>
<resources>
  <string name="app_id">648419090213</string>
</resources>
```

Ilustración 57. ID de nuestra aplicación.

Si hemos hecho el proceso correctamente, nuestro servicio ya estará creado y sólo nos faltará subir nuestra APK. Pasemos a ver cómo se hace esto.

NOMBRE DE LA APLICACIÓN	PRECIO	INSTALACIONES ACTUALES/TOTALES	VALORACIÓN MEDIA / TOTAL	ERRORES Y ANRS	ÚLTIMA ACTUALIZACIÓN	ESTADO
Retruque! 1.03	Gratuita	—	★ 5,00 / 23	—	21/8/2015	Publicada

Ilustración 58. Aplicaciones en Developer Console.

Tenemos que ir a la pestaña llamada Tus aplicaciones y dar clic en el botón verde de la derecha: Añadir nueva aplicación.

**Retruque!**  
com.game.guillermo.truc [Ver en Play Store](#)  
PUBLICADA 21 de agosto de 2015 Anular publicación de aplicación

Estadísticas  
Valoraciones y opiniones  
Errores y ANRs  
Sugerencias de optimización  
Cloud Test Lab

**APK**  
Ficha de Play Store  
Clasificación de contenido  
Precio y distribución  
Productos integrados en la aplicación  
Servicios y APIs

**APK**

PRODUCCIÓN Versión **4**

BETA TESTING Configurar el betatesting de tu aplicación

ALPHA TESTING Configurar el testing alpha de tu aplicación

CONFIGURACIÓN DE PRODUCCIÓN [Subir nuevo APK de producción](#)

APK ACTUAL Fecha y hora de publicación: 21/8/2015 8:57:42

Dispositivos compatibles **8246**  
[Ver lista](#)

Dispositivos excluidos **0**  
[Administrar dispositivos excluidos](#)

▼ VERSIÓN	FECHA DE SUBIDA	ESTADO
4 (1.03)	21/8/2015	en producción

#### Ilustración 59. Subiendo el App

Primero, nos iremos a la pestaña APK, en la que tenemos que subir el APK firmado de nuestra aplicación pulsando sobre Subir nuevo APK de producción. Este APK lo podemos exportar fácilmente utilizando Android Studio. Tardará unos minutos en subir y ya tendremos subida nuestra aplicación al servidor. Pero el trabajo aún no está hecho, todavía tenemos que configurar algunas cosas para llamar la atención de los usuarios.

**Título\***  
Español (Latinoamérica) – es-419

Retruque!  
9 de 30 caracteres

**Descripción breve\***  
Español (Latinoamérica) – es-419

¡Desafía a tus amigos a una partida de Truc con Retruque y su modo multijugador!  
80 de 80 caracteres

**Descripción completa\***  
Español (Latinoamérica) – es-419

Por fin el popular juego de cartas valenciano del Truc en tu móvil, como nunca antes lo habías visto. Nunca había sido tan fácil jugar una partida de Truc, estés donde estés. ¡Desafía a tus amigos con el modo multijugador en línea y demuestra que eres el mejor!

Actualmente, hay algunos dispositivos móviles en los que el juego no funciona correctamente. Estamos trabajando para solucionar pronto este problema.

Características del juego:

- Multijugador en línea 1 vs 1.
- Multijugador en línea 2 vs 2.
- Juega con amigos ingresando con tu cuenta de Google+.
- Juega con gente desconocida buscando una partida.
- Juega con gente cercana a tu localización.

Sistema de logros.  
1480 de 4000 caracteres

Consulta estas [sugerencias sobre cómo crear descripciones de aplicaciones](#) acordes a la política para evitar que la aplicación se suspenda por motivos comunes.

#### Ilustración 60. Descripción de la App.

Seguidamente, tenemos que hacer clic en la pestaña Ficha de Play Store, en la que tendremos que rellenar todos los datos que nos pidan. Estos sí que serán los que vea el usuario final en Play Store, cuando abra la ficha del juego, por lo que es conveniente rellenarlos detenidamente.

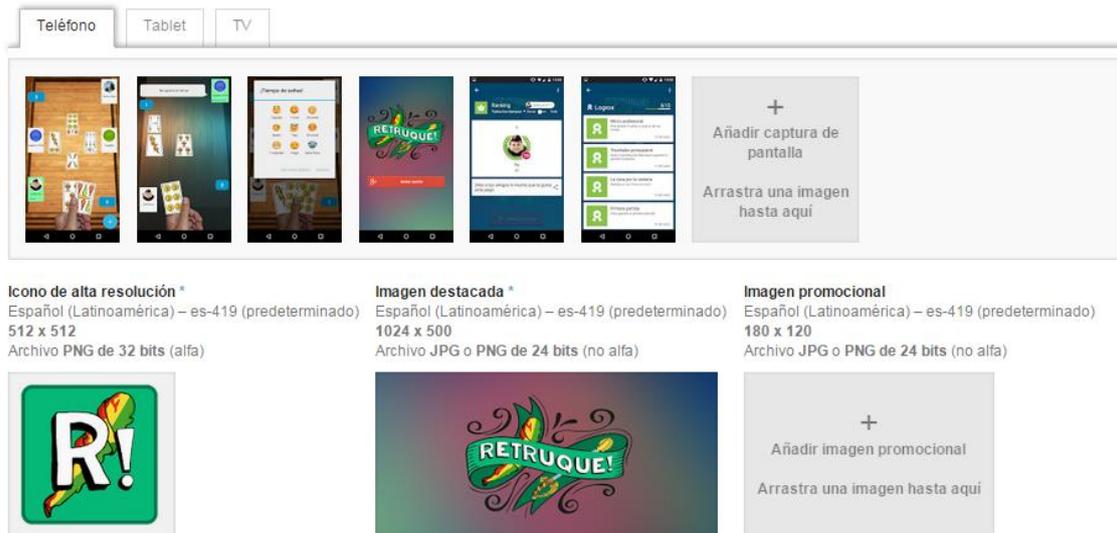


Ilustración 61. Capturas de la App.

Más hacia abajo, tenemos que añadir obligatoriamente un icono para la aplicación, una imagen destacada que será un fondo para la ficha en Play Store y, al menos, una captura de pantalla de la aplicación, aunque es muy recomendable subir más para llamar la atención de los jugadores.



Ilustración 62. Clasificación de contenido.

Para finalizar, solamente nos quedará rellenar un pequeño cuestionario para la clasificación de contenido de nuestro juego. Una vez hecho, veremos como se muestra una clasificación que ha sido asignada a nuestra aplicación. También tenemos que fijar el precio que costará la descarga de la aplicación (en nuestro caso gratuita).

Si hemos hecho bien todo el proceso, podemos proceder a publicar nuestra aplicación y esperar un par de horas para verla y descargarla en Google Play Store.

## 9.4 Precio de la publicación

Sabemos que hay que pagar para darse de alta como desarrollador de Apps y poder subirlas, pero... ¿cuánto tengo que pagar por cada App que subo? La respuesta es nada.

Google nos cobrará lo mismo tengamos una aplicación o 30, así que podemos subir las que queramos. Pero cuidado, no conviene subir cualquier aplicación. Si nos queremos ganar una fama como desarrolladores debemos de programar Apps de calidad y sólo subir las que nos puedan servir para obtener un beneficio. Subir muchas Apps nos puede hacer ganar algo más de dinero pero a la larga la calidad pesa más que el dinero que pudiéramos sacar de subir un gran número de Apps sin mucha calidad.



## 10. Resultados obtenidos

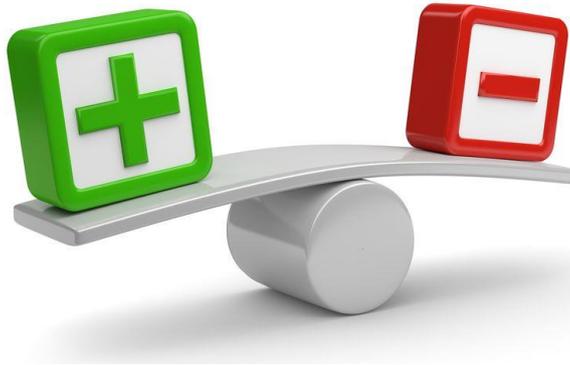


Ilustración 63. Resultados obtenidos.

Nuestra aplicación fue subida a Play Store en su primera versión (1.0) el día 20 de agosto del 2015. Sin embargo, distintas versiones han sido subidas posteriormente debido a que algunos dispositivos móviles experimentaban problemas a la hora de jugar una partida. Esto ha sido arreglado y el juego funciona correctamente.

Actualmente, la versión disponible para descarga es la 1.1. Veamos los resultados que hemos obtenido hasta la fecha de publicación de esta memoria, el día 2 de septiembre del 2015.

### 10.1 Descargas

En los primeros 10 días tras la publicación de la App obtuvimos 125 descargas. Un dato que no está mal, ya que es una aplicación que nadie conocía y que debe ser expandida mediante las redes sociales y la ayuda de los propios jugadores.

Tras solamente 14 días (fecha límite hasta la publicación de esta memoria), conseguimos triplicar las descargas y los ingresos, así como los usuarios que juegan a la App. A 2 de septiembre, hemos superado las 300 descargas.



Ilustración 64. Instalaciones actuales.

Pero este aumento no lo es todo. En dos semanas hemos conseguido ser el Top 1 en aplicaciones nuevas en toda España, en la categoría de cartas de Play Store.

También somos el juego más popular de cartas del momento, superando a juegos que tienen miles de descargas.

Como último dato, aparecemos en la posición 66 en el ranking de mejores juegos de España de cartas de todos los tiempos.

Teniendo en cuenta el cortísimo tiempo de vida de la aplicación, podemos asegurar que esto es todo un éxito.

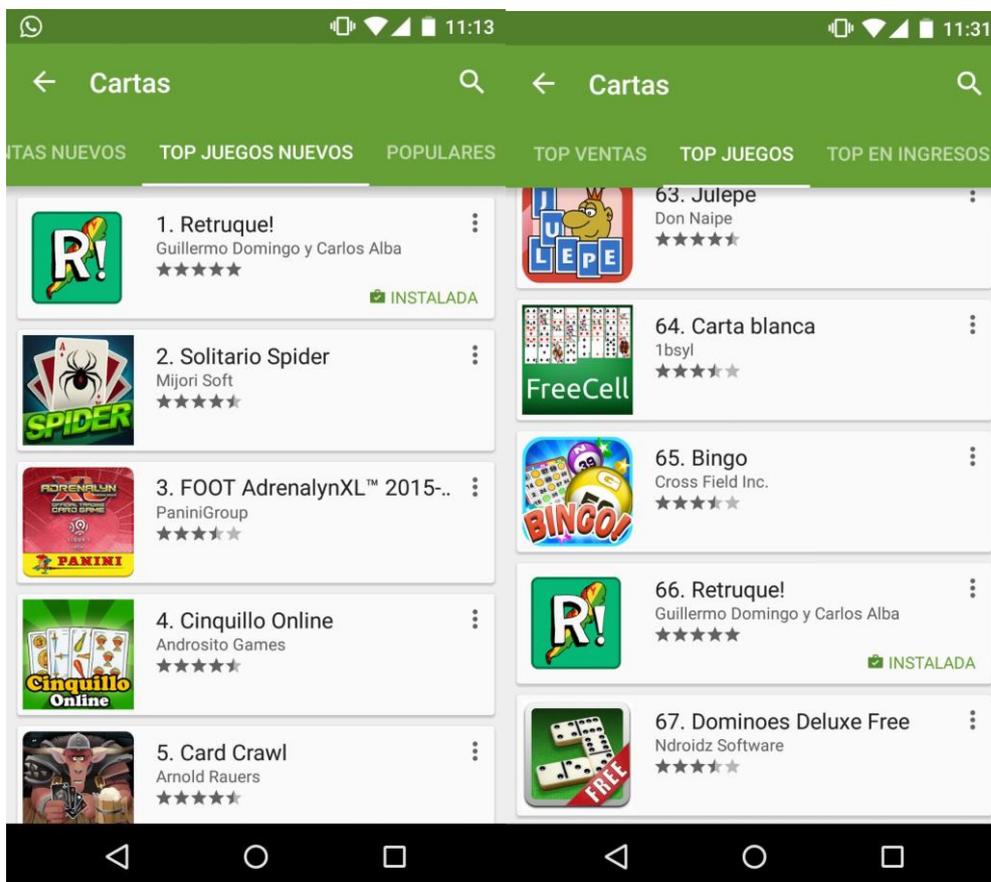


Ilustración 65. Top juegos nuevos de cartas y top juegos de cartas.

Este trabajado de expansión está en proceso y esperamos aumentar las descargas considerablemente en poco tiempo.

## 10.2 Usuarios de la App

Otro término distinto que también conviene analizar son los usuarios activos del juego. Sigamos con el caso de los 10 primeros días. De las 125 descargas que hemos obtenido, tenemos 62 usuarios activos, lo que quiere decir que prácticamente la mitad de los jugadores juega alguna partida habitualmente, mientras que los demás juegan con menos asiduidad.

Cuatro días más tarde, esta cifra se ha visto aumentada como se aprecia en la imagen anterior y más del 60% de los jugadores siguen activos en el juego.

JUGADORES		LOGROS
Jugadores nuevos ?	Jugadores activos ?	Desbloqueado ?
Último día <b>108</b> +280,0%	Último día <b>149</b> +188,1%	Último día <b>179</b> +842,1%
Últimos 7 días <b>189</b> +155,4%	Últimos 7 días <b>224</b> +190,9%	Últimos 7 días <b>198</b> +2.100,0%
Últimos 30 días <b>263</b>	Últimos 30 días <b>266</b>	Últimos 30 días <b>223</b>
MARCADORES		PARTIDAS EN TIEMPO REAL
Puntuaciones publicadas ?	Recuento de partidas iniciado ?	
Último día <b>66</b> +725,0%	Último día <b>277</b> +1.438,9%	
Últimos 7 días <b>75</b> +400,0%	Últimos 7 días <b>308</b> +150,4%	
Últimos 30 días <b>87</b> +1.640,0%	Últimos 30 días <b>586</b> +112,3%	

Ilustración 66. Estadísticas de usuarios.

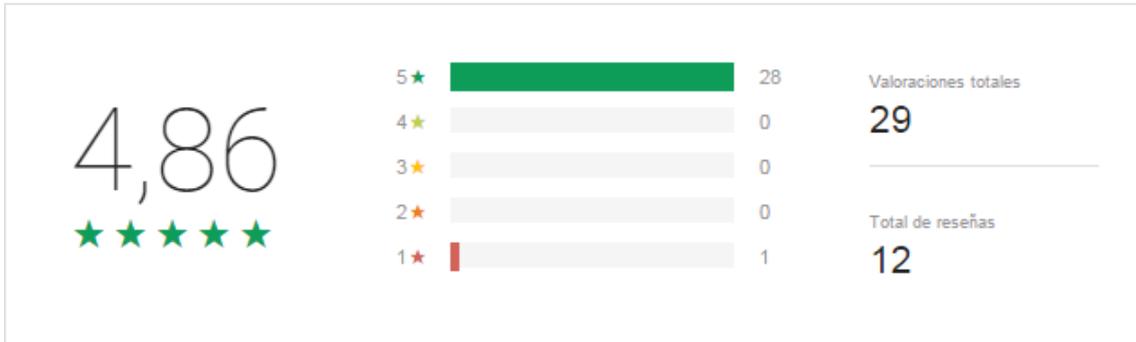
Como podemos ver en la **ilustración 66**, los usuarios han crecido en los últimos días con unos porcentajes enormes. Podemos observar algunos como 2.100% o 1.640%, lo que nos indica que se está produciendo un crecimiento altísimo de usuarios.

Podemos destacar las 586 partidas que ya se han jugado hasta la fecha de hoy y obviamente, este número va creciendo con el paso de las horas.

### 10.3 Opiniones de los jugadores en Play Store

Una de las cosas que más debemos y queremos tener en cuenta son las opiniones de nuestros jugadores, ya que ellos son los que nos proporcionaran beneficios y nuestro trabajo es mantenerles enganchados a nuestro juego.

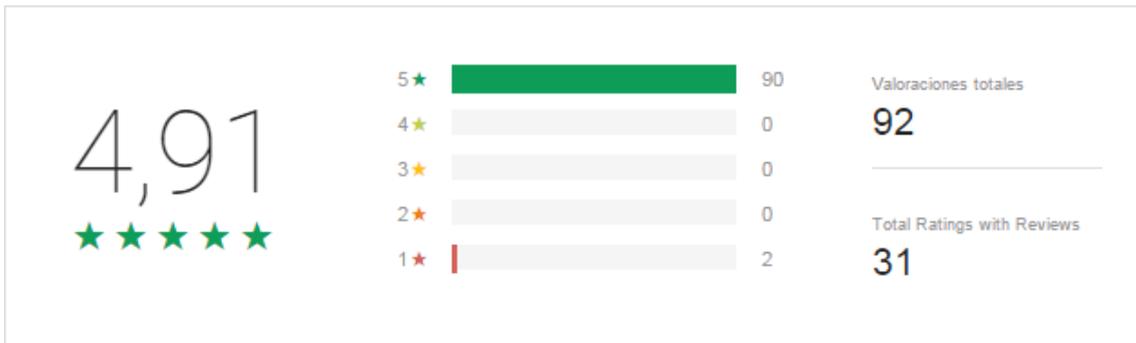
Las votaciones en Play Store tienen un rango del 1 al 5, donde el 1 es muy malo y el 5 es muy bueno. En los primeros 10 días, recibimos 29 votaciones, y en 12 de ellas los usuarios nos han dejaron una reseña escrita.



**Ilustración 67. Votaciones Play Store 1.**

De estas 29 votaciones, 28 usuarios votaron con un 5 y 1 usuario votó con un 1 (**ilustración 67**). Lo que nos da una media de 4,86 sobre 5. Las críticas fueron muy positivas.

Veamos ahora el cambio que se produjo tras 14 días (**ilustración 68**).



**Ilustración 68. Votaciones Play Store 2.**

Hemos obtenido un total de 92 valoraciones y 90 de ellas nos han otorgado 5 puntos. Las otras 2 restantes, nos han puntuado con un punto. La media ha subido a 4,91 y el número de reseñas escritas por los usuarios es de 31.

Aquí tenemos algunas de las reseñas que hemos leído en la ficha del juego en Play Store:



**Ilustración 69. Reseña 1.**



**Jorge Gallego** 25/8/2015 a las 1:00  0  0

★★★★★

Un juego de 10! Excelente juego en el que pasar las horas muertas, joc fora.

Responder a esta opinión

Dispositivo: LG L80 Single (w6)      Versión de la aplicación: 4 (1.03)      Sistema operativo: Android 4.4

Ilustración 70. Reseña 2.



**Chulviyomuss** 21/8/2015 a las 15:03  1  0

★★★★★

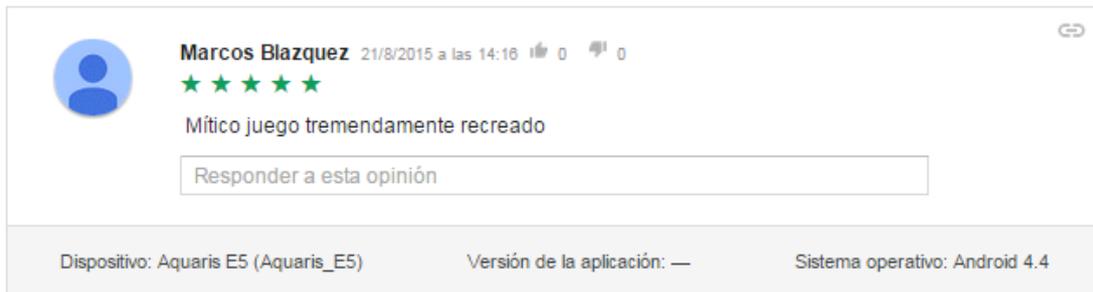
Traducción automática del Catalán      Mostrar opinión original

Un juego por tierra Muy logrado, lo mejor las frases en valenciano. 🐼

Responder a esta opinión

Dispositivo: Galaxy S5 (kte)      Versión de la aplicación: 2 (1.01)      Sistema operativo: Android 5.0

Ilustración 71. Reseña 3.



**Marcos Blazquez** 21/8/2015 a las 14:16  0  0

★★★★★

Mítico juego tremendamente recreado

Responder a esta opinión

Dispositivo: Aquaris E5 (Aquaris\_E5)      Versión de la aplicación: —      Sistema operativo: Android 4.4

Ilustración 72. Reseña 4.



**Salva Escorihuela Sanfeliu** 31/8/2015 a las 14:45  0  0

★★★★★

Superentretenido .... la verdad q ya hacia tiempo q no encontraba un juego de cartas tan divertido

Responder a esta opinión

Dispositivo: Galaxy S4 Mini (serranotte)      Versión de la aplicación: 5 (1.1)      Sistema operativo: Android 4.4

Ilustración 73. Reseña 5.



Ilustración 74. Reseña 6.



Ilustración 75. Reseña 7.

Estas son algunas de las reseñas más relevantes que hemos encontrado. Todas son positivas y nos ayudan a seguir trabajando en el juego, ya que se le ve a los jugadores contentos y con ganas de jugar partidas.

Analizando las respuestas, vemos que existe una gran aceptación entre los jugadores pero, una de ellos nos dice que de momento faltan jugadores. Esto se irá solucionando con el paso del tiempo y el crecimiento de las descargas.

La única pega que podemos sacar son los dos votos negativo que no san asignado con un 1. Sin embargo, la mayoría absoluta nos ha puntuado con la máxima nota.

## 11 Trabajo futuro

En el siguiente epígrafe vamos a establecer el trabajo futuro para nuestra aplicación. El desarrollo de una aplicación móvil, de la misma manera que empieza mucho antes de escribir la primera línea de código, tampoco acaba al escribir la última.

Deberemos por tanto, centrarnos en labores de mantenimiento, mejora, soluciones a problemas y “bugs”, además de acabar con todas las tareas que se nos quedaron en el backlog.

También nos propondremos internacionalizar el juego, y extenderlo a otras regiones donde también se juega al “Truc”, como por ejemplo Argentina.

### 11.1 Tareas pendientes

Si le echamos un vistazo al mapa de características de nuestro proyecto, veremos que no solo logramos implementar todas las características del primer sprint, sino que también gran parte de las del segundo, quedándonos tan solo “Apuestas con monedas” e “Instrucciones”.

- Apuestas con monedas: la idea es utilizar un modelo basado en pagar unas monedas para poder jugar una partida. Todos los jugadores pagarían unas monedas y el o los vencedores se quedarían con las del rival, creando de esta forma un sistema de apuestas. Con esto lo que pretendemos es hacer evolucionar la aplicación a un modelo “Freemium”, en el que se paga por ciertos servicios extra. El usuario podría comprar monedas cuando se le acabasen. De este modo, nosotros conseguiríamos un plus económico.
- Instrucciones: tan solo se juega al “Truc” en Valencia o regiones cercanas. En este contexto, se pretende extender el rango de usuarios, mediante la incorporación de un sistema de instrucciones. Todavía no se ha decidido de qué forma, pero contemplamos varios escenarios. La primera opción es, en el propio juego, crear una pantalla con las instrucciones. Por el momento es la que más fuerza coge. En un segundo plano tenemos la opción de crear un portal web.

En cuanto al tercer sprint tenemos pendientes todas las tareas. Con estas tareas el juego estaría finalizado a nivel de módulos. Muchas de estas características son propias de los juegos más exitosos disponibles.

La primera de ellas sería la inclusión de “login con Facebook”. Sabemos que Google+ no es la principal red social del momento, así que queremos aumentar nuestro número de usuarios permitiendo acceder a la aplicación con una cuenta de Facebook. Hemos comprobado que, aunque el 90% de gente cuenta con un perfil de Google+, la gran mayoría no ha agregado a sus conocidos a sus círculos. Con Facebook desaparecería este problema.

La siguiente tarea tiene que ver con el proceso de transformación a un modelo “Freemium” y es la de establecer una tienda o market para la compra de funcionalidades adicionales. Entre ellas queremos incluir personajes para jugadores, tipos de barajas, además de paquetes de monedas etc.

Con la intención de mejorar la interacción entre jugadores durante la partida, propusimos la inclusión de un chat. Así pues los jugadores podrán comunicarse cuanto quieran, pasando el sistema de frases a formar parte de este nuevo sistema de chat.

En última instancia, posiblemente la creación de torneos sea la novedad más importante. Los jugadores podrán participar en diferentes torneos basándonos en su situación en el ranking. Los ganadores sumarán monedas y experiencia a partes iguales. Creemos que estos torneos servirán de motivación y animarán a los jugadores a demostrar quién es el mejor.

## **11.2 Posibles mejoras**

A continuación veremos unas posibles mejoras.

Se ha pensado en sacar una versión tanto de iOS como web. Con la versión para el sistema operativo de Apple completaremos la cuota de mercado de los smartphones y con la versión web podríamos competir con el juego de “Truc” con el que cuenta Facebook.

Introduciremos la compatibilidad con tablets, de modo que ya cubriremos todos los dispositivos tanto Android como iOS. Evidentemente, cuanto mayor sea el número de dispositivos compatibles, mayor será el número de jugadores.

## **11.3 Expansión a otros países**

Durante nuestro estudio del arte, descubrimos que no solo se jugaba al “Truc” aquí en Valencia. En algunos países del sur de América existe una variante llamada “Truco”, entre ellos Argentina, Colombia o Uruguay.

Nuestro propósito es el de dar a conocer la variante valenciana del juego allí y ver que aceptación tendría.

## 12. Conclusiones

Llegamos al último apartado de la memoria del proyecto, en el que hablaremos de las conclusiones que hemos sacado tras finalizar el juego y de todo el proceso y trabajo que esto conlleva.

### 12.1 Resultados del proyecto

Tras muchos meses de trabajo, nuestro objetivo principal era conseguir desarrollar un juego divertido, atractivo para el usuario y que pudiera tener éxito. Ha resultado complicado pero hemos conseguido una primera versión muy trabajada, incluyendo incluso más características de las inicialmente planificadas y llegando a un nivel que no pensábamos que pudiésemos alcanzar.

El código ha alcanzado una longitud aproximada de 10.000 líneas de código, métrica con la que podemos medir la complejidad del proyecto.

Con el paso de los días, estamos obteniendo un número mayor de descargas y consiguiendo poco a poco expandir el juego entre los usuarios de dispositivos Android, por lo que creemos que en poco tiempo podemos alcanzar un número considerable de instalaciones del juego.

Analizando todas estas cosas, podemos concluir diciendo que el proyecto ha sido un éxito y que puede llegar a crecer mucho.

### 12.2 Aprendizaje

Para poder alcanzar este éxito del que hemos hablado en el anterior apartado, hemos tenido que aprender muchísimo de las distintas tecnologías que han intervenido en el desarrollo del juego.

Sabíamos programar en Android, pero nunca habíamos pensado en hacer algo parecido, algo tan ambicioso. Hemos tenido que adquirir muchos más conocimientos de los que teníamos para entender mejor el sistema Android. Pero esto es sólo el principio.

También hemos tenido que investigar mucho acerca de la tecnología que íbamos a usar para conectar a los jugadores, crear partidas, etc. Hasta que descubrimos Google Play Game Services. Aquí también tuvimos que hacer un gran trabajo de investigación y pruebas para conseguir avanzar en la programación.

Como no, también hemos utilizado muchas tecnologías complementarias a Game Services y Android, como Git con GitHub, AdMob, Developer Console, aprender a subir una App a Play Store... Muchas cosas.

Por tanto podemos decir que, aparte de hacer un Trabajo Final de Grado, hemos aprendido muchísimo y esto nos puede servir de mucha ayuda de cara al futuro. Si tuviéramos que hacer una aplicación parecida, nos costaría mucho menos sin lugar a dudas. Todo el desarrollo del proyecto nos ha sido muy positivo como informáticos.

### 12.3 Opinión personal

Para terminar con este proyecto, escribiré una pequeña opinión personal.

Mi valoración del proyecto es muy positiva en todas las facetas. Me ha ayudado a crecer tanto a nivel profesional como a nivel personal y esto me llena de orgullo.

Cuando comencé con este desarrollo, nunca creí que al finalizar tendría tal satisfacción personal como la que hoy tengo al escribir estas líneas. La satisfacción del trabajo bien hecho, la satisfacción de saber que tantos años de estudio han merecido la pena, la satisfacción de haber logrado los objetivos y haber alcanzado mis metas, la satisfacción de no haberlo hecho solo. En definitiva, el resultado de tantos años de esfuerzo, plasmado en forma de un pequeño trozo de tecnología que nadie me podrá quitar.

Esperamos que nuestro trabajo no finalice aquí, puesto que queremos seguir mejorando la aplicación.

Este proyecto me ha servido muchísimo como aprendizaje. Además me ha enseñado que el mundo de los dispositivos móviles me encanta y que en un futuro, quizá no tan lejano, me gustaría dedicarme a ello.

Creo que “Retruque!” está a la altura de un TFG en Grado en Ingeniería Informática y por supuesto que ojalá nos depare algunas alegrías, por pequeñas que sean.

## 13 Bibliografía

Transparencias de la asignatura “Soluciones para dispositivos móviles”, impartida por la Universidad Politécnica de Valencia durante el año 2015.

Google Play Store, subcategories: Card Games. Recuperado en mayo de 2015, de:

[https://play.google.com/store/apps/category/GAME\\_CARD](https://play.google.com/store/apps/category/GAME_CARD)

Lean Canvas. Recuperado en mayo de 2015, de:

<http://www.leanstart.es/lean-canvas/>

Android Studio Overview, Android Developers. Recuperado en mayo de 2015, de:

<http://developer.android.com/tools/studio/index.html>

Genymotion. Recuperado en junio de 2015, de:

<https://www.genymotion.com>

Conociendo GitHub (© Copyright 2012, Luciano Castillo). Recuperado en junio de 2015, de:

<https://conociendogithub.readthedocs.org/en/latest/>

Google Play Developer Console, Android Developers. Recuperado en junio de 2015, de:

<https://developer.android.com/distribute/googleplay/developer-console.html>

Publicando en Google Play Store, Android Developers. Recuperado en agosto de 2015, de:

<http://developer.android.com/distribute/googleplay/start.html>

AdMob, monetiza tus aplicaciones. Recuperado en julio de 2015, de:

<https://www.google.es/admob/>

Google Play Games Services, Google Developers. Recuperado en junio de 2015, de:

<https://developers.google.com/games/services/>

Real-Time Multiplayer, Google Developers. Recuperado en junio de 2015, de:

<https://developers.google.com/games/services/common/concepts/realtimeMultiplayer>

StackOverFlow. Página de consulta de dudas.

<http://stackoverflow.com/>

## 14. Índice de ilustraciones

Ilustración 1. Logo del juego.....	7
Ilustración 2. Gráfico de Apps más descargadas.....	10
Ilustración 3. Gráfico de Apps más descargadas en Android.....	10
Ilustración 4. Truco Free e Ilustración 5. Truco Free. ....	14
Ilustración 6. Truco Free e Ilustración 7. Truco Free.....	15
Ilustración 8. Truco Argentino.....	16
Ilustración 9. Truco Argentino.....	16
Ilustración 10. Truco Argentino Free. Ilustración 11. Truco Argentino Free.....	17
Ilustración 12. Estudio de la idea.....	21
Ilustración 13. Grafo generado por el Análisis DAFO.....	24
Ilustración 14. Proceso de desarrollo.....	26
Ilustración 15. Mapa de características.....	27
Ilustración 16. Diferentes versiones de Android.....	29
Ilustración 17. Cómo es Android.....	29
Ilustración 18. Inicialización del sistema.....	31
Ilustración 19. Ciclo de vida de una Actividad.....	33
Ilustración 20. Vista del proyecto.....	36
Ilustración 21. Archivos Gradle del proyecto.....	37
Ilustración 22. Estructura de un proyecto.....	38
Ilustración 23. Modo diseño.....	39
Ilustración 24. Multi-API version rendering.....	40
Ilustración 25. Git.....	42
Ilustración 26. Google Admob.....	43
Ilustración 27. Pantalla de cargando 2.....	46
Ilustración 28. Pantalla de cargando 1.....	46
Ilustración 29. Pantalla de Log-in.....	47
Ilustración 30. Menú principal.....	48
Ilustración 31. Sala de espera.....	49
Ilustración 32. Pantalla de uno contra uno.....	50
Ilustración 33. Dialogo de juego.....	51
Ilustración 34. Bocadillo de juego.....	52
Ilustración 35. Pantalla de dos contra dos.....	52
Ilustración 36. Señas del compañero.....	52
Ilustración 37. Diálogo de señas.....	53
Ilustración 38. Victoria e Ilustración 39. Derrota.....	53
Ilustración 40. Invitar a amigos de Google+.....	54
Ilustración 41. Bandeja de invitaciones.....	55
Ilustración 42. Ranking de jugadores.....	56
Ilustración 43. Logros del jugador.....	56
Ilustración 44. Dependencias.....	58
Ilustración 45. Material Dialogs.....	58
Ilustración 46. Floating action button toolbar.....	59
Ilustración 47. FButton.....	60
Ilustración 48. DialogPlus.....	61
Ilustración 49. Tour Guide 1.....	62

Ilustración 50. Tour Guide 2. ....	62
Ilustración 51. Play Store. ....	63
Ilustración 52. Darse de alta como desarrollador. ....	64
Ilustración 53. Servicios de juegos en Developer Console. ....	64
Ilustración 54. Ficha del juego. ....	65
Ilustración 55. Aplicaciones vinculadas. ....	65
Ilustración 56. Ajustes de los servicios. ....	66
Ilustración 57. ID de nuestra aplicación. ....	66
Ilustración 58. Aplicaciones en Developer Console. ....	66
Ilustración 59. Subiendo el App. ....	67
Ilustración 60. Descripción de la App. ....	67
Ilustración 61. Capturas de la App. ....	68
Ilustración 62. Clasificación de contenido. ....	68
Ilustración 63. Resultados obtenidos. ....	70
Ilustración 64. Instalaciones actuales. ....	70
Ilustración 65. Top juegos nuevos de cartas y top juegos de cartas. ....	71
Ilustración 66. Estadísticas de usuarios. ....	72
Ilustración 67. Votaciones Play Store 1. ....	73
Ilustración 68. Votaciones Play Store 2. ....	73
Ilustración 69. Reseña 1. ....	73
Ilustración 70. Reseña 2. ....	74
Ilustración 71. Reseña 3. ....	74
Ilustración 72. Reseña 4. ....	74
Ilustración 73. Reseña 5. ....	74
Ilustración 74. Reseña 6. ....	75
Ilustración 75. Reseña 7. ....	75



## 15. Índice de Tablas

Tabla 1. Retruque! vs Truco Free.....	18
Tabla 2. Retruque! vs Truco Argentino .....	18
Tabla 3. Retruque! vs Truco Argentino Free .....	19
Tabla 4. Lean Canvas.....	22
Tabla 5. Análisis DAFO.....	23
Tabla 6. Costos e ingresos, del primer año de vida de la App. ....	25
Tabla 7. Costos e ingresos en los años posteriores. ....	25