

UNIVERSIDAD SAN FRANCISCO DE QUITO

Colegio de Ciencias e Ingenierías

Diseño, Desarrollo y Lanzamiento de Juego para iPhone

Germán Eduardo Villacreces Jaramillo

Fausto Pasmay, M.S., Director de Tesis

Tesis de grado presentada como requisito
para la obtención del título de Ingeniero de Sistemas

Quito, marzo del 2015

Universidad San Francisco de Quito

Colegio de Ciencias e Ingenierías

HOJA DE APROBACIÓN DE TESIS

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído la Política de Propiedad Intelectual de la Universidad San Francisco de Quito y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo de investigación quedan sujetos a lo dispuesto en la Política.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo de investigación en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma: _____

Nombre: Germán Eduardo Villacreces Jaramillo

Código de estudiante: 00011344

C. I.: 1711669307

Lugar: Quito, Ecuador

Fecha: 15 de marzo de 2015

DEDICATORIA

A toda mi familia y especialmente a mis padres. Gracias por todo el constante apoyo brindado en el transcurso de toda mi carrera universitaria y en la realización de esta tesis.

AGRADECIMIENTOS

Gracias a mis padres y a toda mi familia por el apoyo e impulso que me brindaron para acabar mi Proyecto de Tesis.

RESUMEN

El proyecto de tesis presenta el proceso completo de investigación, diseño, desarrollo, lanzamiento y promoción de un juego para iPhone. Desde la investigación inicial y diseño completo del juego, hasta el uso de diferentes métodos para promocionar el juego en redes sociales e Internet buscando maximizar las ventas del mismo. Adicionalmente, se analizan los resultados de las ventas para llegar a las conclusiones finales del proyecto de tesis.

ABSTRACT

The project presents the entire process of development, launch and marketing of an iPhone game. From the initial research and complete game design, to the use of different methods to promote and maximize the sales of the game. In addition, the sales results are analyzed to arrive to the final conclusions of the thesis project.

TABLA DE CONTENIDO

© Derechos de autor	4
Dedicatoria	5
Agradecimientos.....	6
Resumen	7
Abstract.....	8
Tabla de contenido	9
Tablas	10
Figuras.....	11
Aspector Preliminares.....	13
Justificación del Proyecto	13
Objetivo General	14
Objetivos Específicos.....	14
Metas.....	15
Marco Teórico.....	16
iPhone.....	16
App Store	16
Juegos para iPhone.....	16
Metodología de desarrollo de software Agile.....	16
Tecnologías.....	17
Herramientas de desarrollo	18
Arquitectura de Sistemas y Planificación de Proyecto	21
Arquitectura de la Aplicación y Sitio Web	21
Arquitectura de Juego	21
Arquitectura de Sitio Web	22
Historias de Usuario y Tareas	23
Organización de Tareas y Definición de Sprints	26
Desarrollo del Juego y Sitio Web.....	27
Desarrollo del Juego	27
Desarrollo del Sitio Web	48
Control de Calidad, Publicación y Marketing.....	58
Control de Calidad.....	58
Publicación del Juego en el App Store	61
Marketing	66
Análisis de Resultados y Conclusiones	71
Resultados.....	71
Conclusiones.....	75
Referencias	77
Anexo A: Tareas del Proyecto.	79
Anexo B: Comentarios del Juego en el App Store.	85

TABLAS

Tabla 3.1 Historias de Usuario del juego “Drive 2 Survive”	25
Tabla 3.2 Historias de Usuario del sitio web de Oubii Games	26
Tabla 6.1 Número de Likes en Facebook.....	71
Tabla 6.2 Número de seguidores en Twitter	72

FIGURAS

Figura 3.1	Arquitectura de “Drive 2 Survive”	21
Figura 3.2	Arquitectura del sitio web de Oubii Games	22
Figura 4.1	Escenas en Corona SDK	28
Figura 4.2	Pantalla de Introducción	29
Figura 4.3	Pantalla de Menú Principal	29
Figura 4.4	Ventana de Configuraciones	30
Figura 4.5	Pantalla de Puntajes “Game Center”	32
Figura 4.6	Ventana de Instrucciones	32
Figura 4.7	Ventana de Créditos	33
Figura 4.8	Escena de Carga	34
Figura 4.9	Juego en Acción	35
Figura 4.10	Juego en Pausa	36
Figura 4.11	Final del Juego	36
Figura 4.12	Flujo de Información Entre Módulos	37
Figura 4.13	Herramienta Physics Editor	40
Figura 4.14	Integración con Facebook	43
Figura 4.15	Integración con Twitter	44
Figura 4.16	Configuración de Tablas de Puntajes de “Game Center”	45
Figura 4.17	Pantalla de Puntajes “Game Center”	46
Figura 4.18	Muestra de Vehículo Principal	47
Figura 4.19	Sprite de los Vehículos del Juego	48
Figura 4.20	Configuración de Servidor	50
Figura 4.21	Página Inicial del Sitio Web	53
Figura 4.22	Página Sobre la Empresa	54
Figura 4.23	Página de publicación del blog	55
Figura 4.24	Formulario de Contacto	56
Figura 4.25	Página de “Drive 2 Survive”	57
Figura 5.1	Sistema Test Flight	60
Figura 5.2	Ejemplo de Correo Electrónico de Test Flight	61
Figura 5.3	Proceso de Publicación del Juego	64
Figura 5.4	Página del Sorteo	68

Figura 5.5 Anuncio de Publicidad para el Sorteo	70
Figura 6.1 Número de Likes en Facebook.....	72
Figura 6.2 Número de Seguidores en Twitter.....	73
Figura 6.3 Número de Descargas	73
Figura 6.4 Valor en Ventas	74
Figura 6.5 Calificaciones en el App Store	74

ASPECTOR PRELIMINARES

Antecedentes

Oubii Games es una empresa registrada en Estados Unidos y dedicada a la creación de juegos para dispositivos móviles. Siendo una empresa nueva, todavía no a desarrollado ningún juego.

Corona SDK es una plataforma que trabaja encima del iOS Nativo y permite crear juegos de manera mas rápida. Esta plataforma esta construida específicamente para crear juegos para el iPhone y teléfonos Android. El desarrollador puede programar juegos para el iPhone en el lenguaje Lua, que es conocido por su simplicidad y semejanza al Actionscript.

Este proyecto incluye el diseñar, desarrollar y lanzar un juego de iPhone para la empresa Oubii Games. Un juego de autos en 2D que sea entretenido además de lanzar una campaña de marketing promocionando el juego para incrementar ventas. El nombre del juego es Drive To Survive y va a tener el costo de \$0.99. El juego va a estar disponible de manera global en el App Store.

Justificación del Proyecto

Oubii Games fue creada con el fin de crear juegos entretenidos para dispositivos móviles. Como primer juego, un juego simple pero entretenido que compita con los de su categoría. La importancia de crear este juego para la categoría de autos en 2d se determinó al evaluar los juegos de la misma categoría. Los juegos existentes en el App Store de autos en 2d no tienen buenas gráficas que hacen mas entretenido y llamativo al juego.

Es importante que el primer juego creado por Oubii Games compita con otros de su misma categoría al tener ventajas sobre el resto, incluyendo mejores gráficas y mejor

dinámica dentro del juego. Son estas ventajas las que potencialmente pueden colocar al primer juego de Oubii Games en el primer puesto de su categoría. Los usuarios de iPhone tendrán una mejor opción al elegir un juego de autos en 2D.

Objetivo General

Crear un juego para el iPhone de la empresa Oubii Games y así introducir a la empresa en el mercado de juegos móviles.

Objetivos Específicos

- Crear el primer juego móvil de la empresa Oubii Games.
- Introducir a la empresa Oubii Games en el mercado de juegos móviles.
- Contratar un excelente ilustrador para crear gráficas llamativas para el juego.
- Diseñar el juego de tal manera que no tenga final y sea infinito.
- Permitir al usuario publicar el puntaje final a través de las redes sociales mas populares:
 - Twitter
 - Facebook
- Integrar el juego con la comunidad Game Center, para compartir los puntajes con otros usuarios.
- Al ejecutar el juego, mostrar un splash screen con el logo de la empresa.
- Después de mostrar el splash screen, desplegar una pantalla con el menú principal que contenga las siguientes opciones:
 - Start Engine (Empezar el juego)
 - Settings
 - Scores

- Instructions
- Credits
- Crear una pantalla para cada opción del menu.
- Empaquetar y subir la aplicación al AppStore de Apple para poder ser distribuida.
- Crear un plan de marketing para poder promocionar al juego y lograr conseguir un buen número de descargas. El plan de marketing debe incluir:
 - Crear un trailer del juego y publicarlo en YouTube y Vimeo para incitar a la gente a comprar el juego.
 - Crear un website para la empresa Oubii Games donde los usuarios puedan leer información de la empresa, el juego y enviarnos comentarios y sugerencias sobre el juego.
 - Crear un plan para actualizar el juego y mejorarlo tomando en cuenta todos los comentarios y sugerencias recibidos a través del sitio web.

Metas

- Tener creado un juego para iPhone de autos.
- Tener una cuenta de developer en Apple.
- Ser aprobado en el proceso de publicación de apps del App Store.
- Tener publicado el juego en el App Store.
- Tener definido un plan de marketing para promocionar el juego.
- Tener el sitio web de la empresa publicado para lograr promocionar el juego.
- Obtener el máximo numero de descargas después de culminar la campaña de marketing.

MARCO TEÓRICO

iPhone

El iPhone es un teléfono inteligente creado por Apple en el año 2007. Fue el primer teléfono con pantalla táctil multitouch. Adicional a la pantalla, se destacó por las diferentes características y usos. El iPhone tiene GPS, cámara de fotos y filmadora; cada año Apple ha lanzado una nueva versión de iPhone con mejores características.

App Store

El App Store es un mercado virtual donde todos los usuarios de iPhone pueden ir y descargar aplicaciones. Algunas aplicaciones cuestan, otras son gratuitas. Cualquier desarrollador puede subir un app al App Store al entrar al iOS Developer Program. Este programa cuesta \$99 USD al año y brinda todos los recursos necesarios al desarrollador para crear aplicaciones.

Juegos para iPhone

El App Store tiene, hasta la fecha, más de un millón de apps. Un gran porcentaje de esas aplicaciones son juegos y existen varias subcategorías dentro de la categoría de Juegos. El juego Drive 2 Survive se encuentra en las subcategorías de “Racing”, y “Arcade”.

Dentro de estas categorías se puede encontrar juegos como “Real Racing” y “Asphalt”. Estos son juegos de carreras en 3D. Drive 2 Survive es un juego de autos en 2D, por lo tanto pertenece a la sub-categoría de juegos de autos en 2d.

Metodología de desarrollo de software Agile

La metodología Agile se basa en iteraciones donde el desarrollo es incremental. Los requerimientos y las soluciones cambian continuamente usando herramientas de

organización y colaboración en conjunto. Esta metodología se adapta al cambio rápidamente y permite la integración de cambios durante el transcurso del proyecto. El concepto de Agile fue concebido en el 2001. Se decidió usar la metodología Agile para este proyecto por su gran flexibilidad y adaptación. Hoy en día esta metodología se usa bastante en el desarrollo de sistemas.

Tecnologías

Lua

Lua es un lenguaje de programación liviano diseñado como lenguaje de scripting enfocado en poder extenderse. Lua permite el desarrollo en cualquier plataforma y tiene un API muy simple comparado con otros lenguajes de programación. Para este proyecto se usó Lúa por que es el único lenguaje utilizado en la plataforma CoronaSDK, véase literal 2.6.2.

HTML

HTML es el lenguaje principal usado para crear páginas web, consiste de elementos que definen la estructura del contenido de una página web. El código HTML es interpretado por un navegador de internet y presentado de forma visual al usuario.

CSS

Los CSS son conjuntos de propiedades visuales que se aplican a los elementos HTML para definir su estilo de visualización. Mientras que el HTML define la estructura y componentes de una página web, los CSS se aplican para darles un formato.

PHP

PHP es un lenguaje de programación que es ejecutado del lado del servidor. Diseñado específicamente para el desarrollo web pero que también puede ser utilizado

como un lenguaje de programación general. Una gran cantidad de sitios web en el internet utilizan PHP, tiene una gran popularidad por ser software libre. El código PHP es interpretado por un procesador por lo tanto no requiere ser compilado.

Drupal

Drupal es un manejador de contenido mantenido por una comunidad de mas de medio millón de usuarios y desarrolladores. Es software libre desarrollado en PHP con un sistema modular que permite a desarrolladores extender su funcionalidad a través de módulos. Este sistema se utiliza para desarrollar sitios web de una manera rápida, y modular.

Facebook

Facebook es una red social creada en el 2007 por Mark Zuckerberg. Facebook permite a millones de usuarios conectarse y compartir información. Esta red social provee una plataforma que permite a otras aplicaciones compartir información desde distintos dispositivos, incluyendo el iPhone. El juego Drive 2 Survive se conecta a la plataforma de Facebook y permite a los usuarios compartir el puntaje logrado en el juego, dentro de Facebook.

Twitter

Twitter es una red social de “microblogging” que permite a un usuario compartir mensajes en texto plano. Los usuarios pueden conectarse con otros usuarios y ver los mensajes de texto publicados en lo que se denomina el “Timeline”. Drive 2 Survive se conecta a la plataforma de Twitter y permite a los usuarios publicar su marcador para que sus seguidores puedan ver.

Herramientas de desarrollo

XCode

XCode es el ambiente para desarrollo de Apple, se distribuye a todos los desarrolladores de forma gratuita junto con el sistema operativo Mac OS X. Incluye una colección de compiladores del proyecto GNU y puede compilar C y C++.

Corona SDK

Corona SDK es una plataforma que trabaja encima del iOS Nativo y permite crear juegos de manera mas rápida. Esta plataforma esta construida específicamente para crear juegos para el iPhone y teléfonos Android. El desarrollador puede programar juegos para el iPhone en el lenguaje Lua, luego compilar el juego para que funcione en el iPhone y después poder subir el app compilado al App Store.

TestFlight

Test Flight es una herramienta para desplegar aplicaciones iOS “Over The Air” sin ningún tipo de complicación. Permite a los desarrolladores invitar a personas a probar la aplicación con una invitación a través de correo electrónico. El usuario que recibe la invitación solo debe aceptarla y abrir el link donde se puede descargar la aplicación. TestFlight facilita el control de calidad al permitir desplegar rápidamente un app en todos los dispositivos que tienen acceso.

Photoshop

Photoshop es una herramienta profesional para crear y editar imágenes. Con esta herramienta se pueden importar y exportar gráficos en el formato que se requiere. Photoshop se puede utilizar para editar fotografías, crear gráficos para sitios web e inclusive animaciones para juegos. Las animaciones del juego Drive 2 Survive se crearon usando la ayuda de este programa, optimizando los gráficos del juego.

Texture Packer

Texture Packer es una herramienta muy útil que se complementa con Corona SDK. Permite agrupar diferentes gráficos en una sola imagen que se llama “sprite”. El sprite es una imagen que contiene varias ilustraciones que se usan para crear una animación en base a las coordenadas de cada ilustración. Un ejemplo de un sprite usado en el juego es la explosión de los autos. Texture Packer también ayuda a importar los sprites en juegos y esto reduce el número de requests al disco, de tal manera que cargar gráficos en un app se vuelve más rápido y fácil.

Physics Editor

Physics Editor es un complemento de la herramienta Corona SDK. Permite definir objetos para el motor de física que se usa para el desarrollo de juegos. Con esta herramienta uno define la forma física de un cuerpo basado en una imagen. Este programa se usó en Drive 2 Survive para generar de manera más fácil las formas físicas de cada auto. Tomando en cuenta que cada auto tiene una forma diferente. Al definir cada auto como una forma física, la colisión de objetos en el juego se vuelve mucho más real.

Coda

Coda es un editor para todo tipo de lenguaje de programación, tiene soporte para PHP, HTML, CSS y Lua. Permite manejar proyectos por carpetas de archivos. Coda se utilizó en este proyecto para editar los archivos Lua, PHP, HTML y CSS.

ARQUITECTURA DE SISTEMAS Y PLANIFICACIÓN DE PROYECTO

Antes de empezar con la programación del juego y del sitio web, se debe definir la arquitectura de cada sistema. Luego se debe organizar el proyecto en etapas y tareas en base a las historias de usuarios. Cada etapa dentro del proyecto se llama “Sprint”.

Arquitectura de la Aplicación y Sitio Web

Este proyecto tiene dos componentes principales en el contexto del proyecto. Por un lado tenemos el juego para el iPhone, y por otro lado el sitio web de la empresa Oubii Games.

Arquitectura de Juego

Primero vamos a definir la arquitectura de la aplicación, el juego. La siguiente figura explica de manera gráfica los diferentes componentes del juego “Drive 2 Survive”:

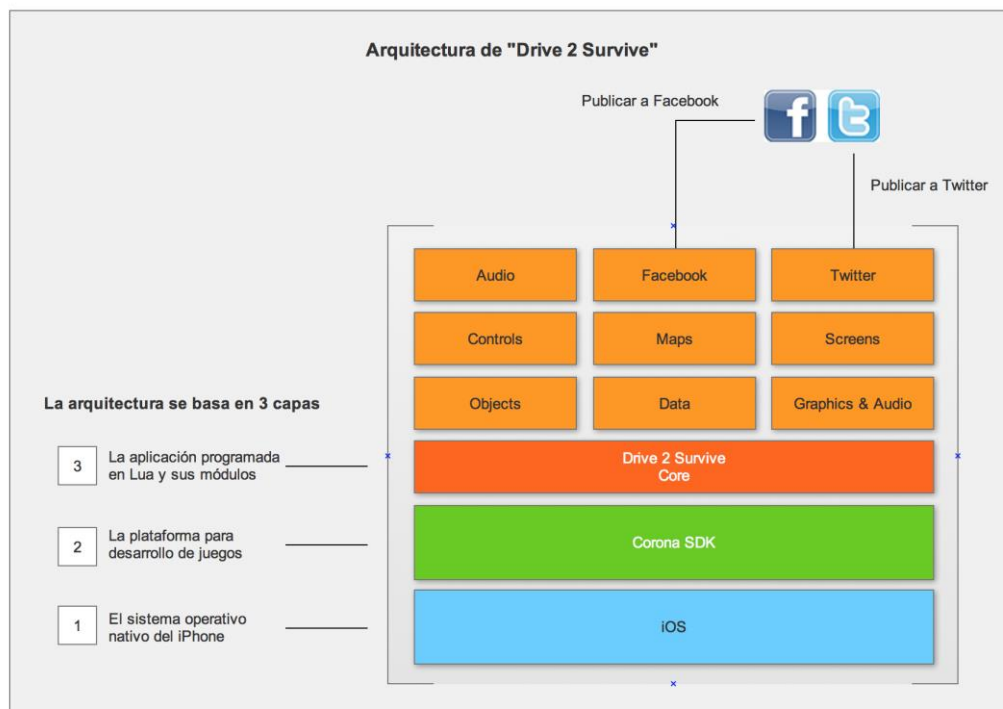


Figura 3.1 Arquitectura de “Drive 2 Survive”

La arquitectura del juego tiene tres componentes principales:

1. Sistema Operativo iOS: El sistema operativo nativo del iPhone.
2. Plataforma Corona SDK: La plataforma de desarrollo que compila el código Lua y crea una aplicación nativa para iOS.
3. Core de “Drive 2 Survive”: En esta capa se encuentran todos los archivos Lua, donde está programado el core del juego y sus módulos.

Arquitectura de Sitio Web

A continuación se define la arquitectura del sitio web de Oubii Games. Este sitio web tiene varios componentes, y la siguiente figura las describe:

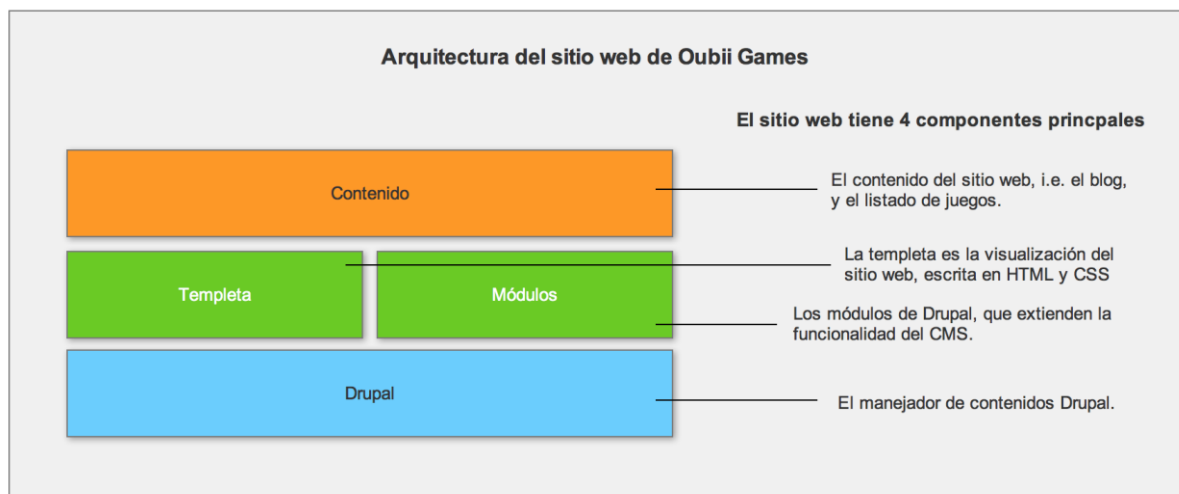


Figura 3.2 Arquitectura del sitio web de Oubii Games

El sitio web de Oubii Games es construido con el manejador de contenidos llamado Drupal. Por un lado tenemos la plantilla, también llamada “theme”. Todos los elementos se visualizan a través de la plantilla. Y por otro lado tenemos los módulos, que extienden la funcionalidad del sitio web. Adicionalmente, se pueden instalar módulos para aumentar más funcionalidad. Por último, tenemos el contenido, en el que se ingresa la información

que se despliega en cada página. Los menús, artículos y diferentes páginas son parte del contenido del sitio web.

Historias de Usuario y Tareas

La planificación del proyecto empieza con las historias de usuarios. Esto nos ayuda definir las tareas y especificaciones de lo que se debe hacer. Estas historias describen lo que los usuarios pueden hacer con los sistemas. Luego organizamos estas historias en grupos para definir los diferentes componentes y subcomponentes que tiene cada sistema. Al definir estos componentes y sus historias, se puede proceder a crear y organizar las tareas y así poder organizar el proyecto. Las historias de usuario para este proyecto se dividen en las historias del juego y las historias del sitio web de Oubii Games.

Historias de Usuario para el Juego

La tabla 3.1 muestra, sin ningún orden, las historias de usuario para el Juego:

Id	Historia
HJ01	El usuario puede abrir el juego desde el iPhone y ver la pantalla de presentación con el logo Oubii Games.
HJ02	El usuario puede ver en el juego la pantalla principal con todas las opciones del juego antes de empezar a jugar.
HJ03	El usuario puede configurar el nivel de volumen del menu.
HJ04	El usuario puede configurar el nivel de volumen del juego.
HJ05	El usuario puede configurar el nivel de volumen de los sonidos especiales.
HJ06	El usuario puede configurar el nivel de sensibilidad del control del auto.
HJ07	El usuario puede decidir si usa el acelerometro del iPhone o botones en pantalla para controlar el auto.
HJ08	El usuario puede ver su mejor puntaje y los puntajes de otros jugadores a

	través de la comunidad “Game Center”.
	El usuario puede leer instrucciones de cómo se juega.
HJ10	El usuario puede leer los créditos del juego.
HJ11	El usuario puede ir al sitio web de Oubii Games desde el juego.
HJ12	El usuario puede empezar el juego desde la pantalla principal.
	El usuario debe ver una pantalla descriptiva mientras espera a que el juego se cargue para empezar.
HJ14	El usuario puede jugar y visualizar todos los detalles de la carretera al empezar el juego, con el auto ya en movimiento.
HJ15	El usuario puede usar los controles definidos para mover el auto principal de en sentido de izquierda o derecha.
HJ16	El auto que maneja el usuario en el juego debe ser color rojo.
HJ17	El usuario puede esquivar otros autos que están en tráfico y así evitar chocarse.
HJ18	El usuario puede pausar en cualquier momento el juego, sea para regresar al menu principal del juego o para continuar después de una pausa.
HJ19	La cantidad y ubicación de los otros autos debe ser enfocada en que la experiencia del juego sea adictiva y aumente la dificultad mientras progresa el jugador.
HJ20	El usuario puede jugar para siempre, sin que el juego tenga fin obligado.
HJ21	El usuario puede ver un puntaje en pantalla que se incrementa mientras avanza en el juego. Mientras más avanza el usuario en el juego, mejor puntaje se le otorga.
HJ22	El usuario debe poder destruir otros autos que estén en el camino para abrir paso, usando un máximo de dos armas.
HJ23	El usuario debe visualizar la animación de una explosión al ver que un auto se destruye.
HJ24	El usuario puede ver distintos paisajes mientras avanza en el juego.
HJ25	El usuario puede ver nubes en movimiento encima de los autos.
HJ26	El usuario puede ser advertido de choques que sucedan en la carretera para poder esquivarlos.
HJ27	El auto del usuario puede ser perseguido por la policia y podrá usar maniobras para hacer que el auto del policía se choque.
HJ28	El usuario podrá recargar sus armas encontrando íconos en la carretera que represente el arma. (Puede ser cohetes)
HJ29	El usuario deberá perder al entrar en contacto con cualquier de los otros autos.
HJ30	El usuario deberá ver la explosión de su auto cuando pierde.
HJ31	El usuario puede ver una pantalla de “Game Over” cuando pierde, y escoger entre volver a jugar o regresar al menú principal.
HJ32	El usuario puede compartir su puntaje final al perder a la red social Facebook con un mensaje.

HJ33	El usuario puede compartir su puntaje final al perder a la red social Twitter con un mensaje predeterminado.
HJ34	El usuario puede escuchar sonidos especiales para cada objeto.

Tabla 3.1 Historias de Usuario del juego “Drive 2 Survive”

Historias de Usuario para el Sitio Web

La tabla 3.2 muestra, sin ningún orden, las historias de usuario para el Juego:

Id	Historia
HW01	El usuario puede acceder al sitio web de Oubii Games a través de los dominios www.oubiigames.com o www.oubii.com .
HW02	El usuario puede ver la página principal del sitio web con imágenes del juego “Drive 2 Survive” y seleccionar diferentes opciones de navegación.
HW03	El usuario puede ver el logo de la compañía en todas las páginas web del sitio.
HW04	El usuario puede ver el slogan de la compañía en la página principal.
HW05	El usuario puede ver el blog de la compañía.
HW06	El usuario puede ver los juegos de la compañía basados en fecha de lanzamiento.
HW07	El usuario puede ver la página web de la campaña de marketing.
HW08	El usuario puede hacer Like a la página de Facebook de Oubii Games.
HW09	El usuario puede hacer Follow a la cuenta de Twitter de Oubii Games.
HW10	El usuario puede leer información sobre la compañía.
HW11	El usuario puede contactar a la compañía a través de un formulario.

Tabla 3.2 Historias de Usuario del sitio web de Oubii Games

Organización de Tareas y Definición de Sprints

Con las historias de cada sistema, podemos crear y organizar las tareas. Este procedimiento consiste en agrupar las historias en base al contexto que representan y con eso definir su componente y subcomponente (en caso de aplicarse). A continuación podemos observar un ejemplo:

Agrupamos las historias HJ01 – HJ13 en un componente principal llamado Menus. El resto de historias, HJ13 – HJ34, las agrupamos en otro componente llamado Gameplay. Las agrupamos de esta manera ya que el primer grupo de historias se refiere a las pantallas del juego antes de empezar a jugar, y el resto se refiere al juego en sí. Tenemos ahora definidos dos componentes principales: Menus y Gameplay.

Ahora realizamos un procedimiento similar para cada componente, creando las tareas en base a cada historia. Por ejemplo para el caso de la historia de usuario HJ01, creamos la siguiente tarea:

D2S-001: Crear pantalla de presentación que se muestre al iniciar el juego mostrando el logo de Oubii Games.

Hacemos lo mismo para cada historia, algunas historias van a generar mas de una tarea. El resultado de este procedimiento es una tabla con todas las tareas para desarrollar cada sistema. El siguiente paso es incluir un estimado de horas para cada tarea y agruparlas en Sprints. Un sprint es una iteración. Al definir cada sprint, tenemos completo el cronograma de nuestro proyecto:

(Anexo A)

DESARROLLO DEL JUEGO Y SITIO WEB

Después de crear todas las tareas creadas y organizarlas en Sprints, el desarrollo del juego y el sitio web pudo empezar. A continuación se ven los detalles del proceso de desarrollo de cada uno de los sistemas.

Desarrollo del Juego

El primer objetivo fue completar con el desarrollo del juego, el cuál tiene varios componentes:

- Pantallas y navegación
- Clases y objetos físicos
- Lógica de juego
- Integración con redes sociales
- Gráficos

La parte gráfica de un juego es muy importante, debe ser llamativa para que usuarios que visiten el App Store quieran descargar el juego.

En la plataforma Corona SDK, cada pantalla que se ve se denomina una escena. Cada escena puede tener varios componentes visuales. La figura 4.1 explica gráficamente como funcionan las escenas en esta plataforma:

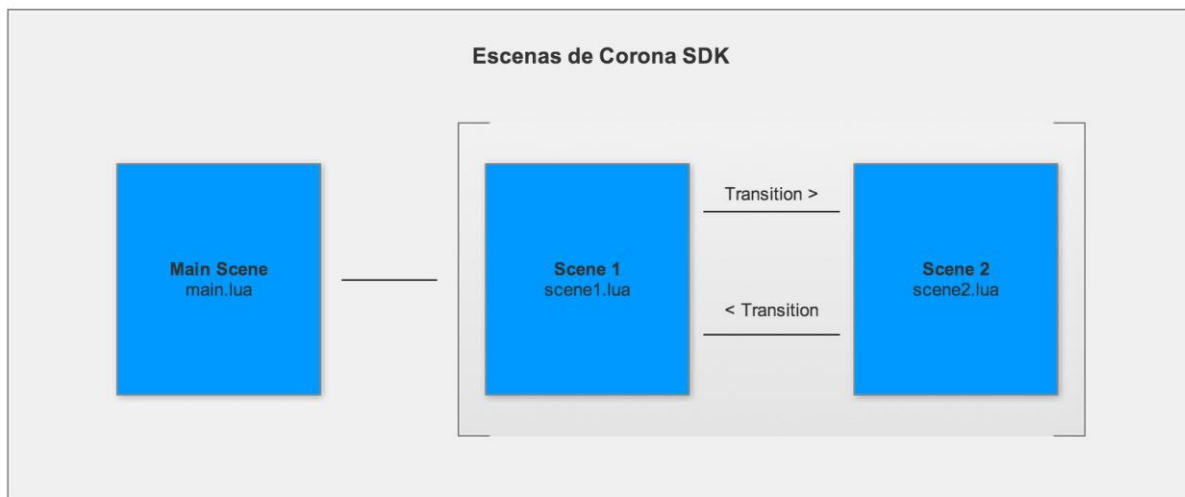


Figura 4.1 Escenas en Corona SDK

Cada escena se almacena en memoria en la forma de un stack, y en cualquier momento podemos colocar a nuestro juego en la escena que queremos y consumir la mínima cantidad de memoria. El juego “Drive 2 Survive” está dividido en 4 escenas:

1. Introducción
2. Menú Principal
3. Carga (“Loading...”)
4. Juego

Las escenas son los componentes padres de todo el juego, se pueden mostrar u ocultar ya que se mantienen en memoria. El desarrollo de cada escena, junto con los gráficos incluidos se detalla a continuación.

Escenas y Navegación

Cuando hablamos de la navegación nos referimos al menú principal del juego y sus diferentes ventanas. La primera, es la del logotipo de la empresa Oubii Games, que podemos observar en la figura 4.2:



Figura 4.2 Pantalla de Introducción

El desarrollo de la escena de introducción fue trivial, la plataforma de Corona SDK incluye animaciones que se pueden usar con cada escena. Se usó la herramienta Photoshop para crear la imagen con fondo blanco y el logotipo en el centro. Al empezar el juego, la imagen de la Figura 4.2 se muestra con un efecto de aparición.

La figura 4.3 muestra una de las escenas más importantes, el menú principal:



Figura 4.3 Pantalla de Menú Principal

De esta pantalla se puede empezar el juego, cambiar las configuraciones, ver los puntajes en la comunidad “Game Center”, leer las instrucciones para jugar, leer los créditos y empezar a jugar.

Cada opción del menú principal se muestra como una ventana semi-transparente. La primera ventana de opciones, es la de las Configuraciones (“Settings”), como se muestra en la figura 4.4:

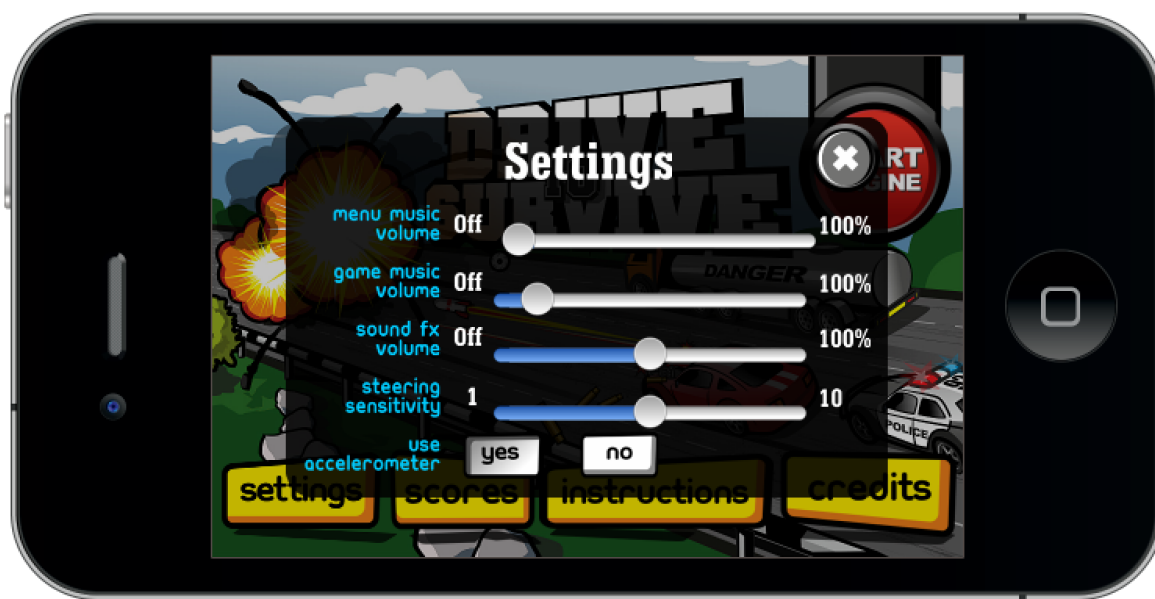


Figura 4.4 Ventana de Configuraciones

En la ventana de configuraciones, el usuario puede cambiar el nivel de volumen de la música de los menús, de la música del juego y de los efectos especiales. El usuario también puede configurar la sensibilidad del control del auto y decidir usar el acelerómetro del iPhone, o botones de pantalla.

Al programar esta ventana, utilice los componentes de la plataforma CoronaSDK llamados “Widgets”. Estos componentes nos ahorran varias horas de programación ya que sólo hay que declararlos y enviar los parámetros necesarios para que funcionen. También son configurables en su apariencia, simplemente debemos crear las imágenes para cada

elemento del componente. En este caso, yo utilize el componente llamado slider. El slider es una barra que tiene un boton que se puede deslizar de un lado para el otro para permitir al usuario escoger un valor dentro de un rango. El ejemplo 4.1 muestra la declaración del slider para el volumen de los menús:

```

201     local assetDir = "graphics/widget_ios/"
202     local musicVolumeSlider = widget.newSlider{
203         top = 102,
204         left = 180,
205         background = assetDir .. "slider/sliderBg.png",
206         fillImage = assetDir .. "slider/sliderFill.png",
207         fillWidth = 2, leftWidth = 12,
208         handle = assetDir .. "slider/handle.png",
209         handleWidth = 32, handleHeight = 32,
210         value = GameSettings.MusicVolume*100,
211         listener = setMusicVolume,
212         width = 200
213     }
214

```

Ejemplo 4.1 Declaración de Slider

El usuario puede salir del menú de configuraciones utilizando el boton de cerrar X en la esquina superior derecha de la ventana.

La siguiente opción del menú principal es la pantalla de Puntajes, mostrada en la figura 4.5:

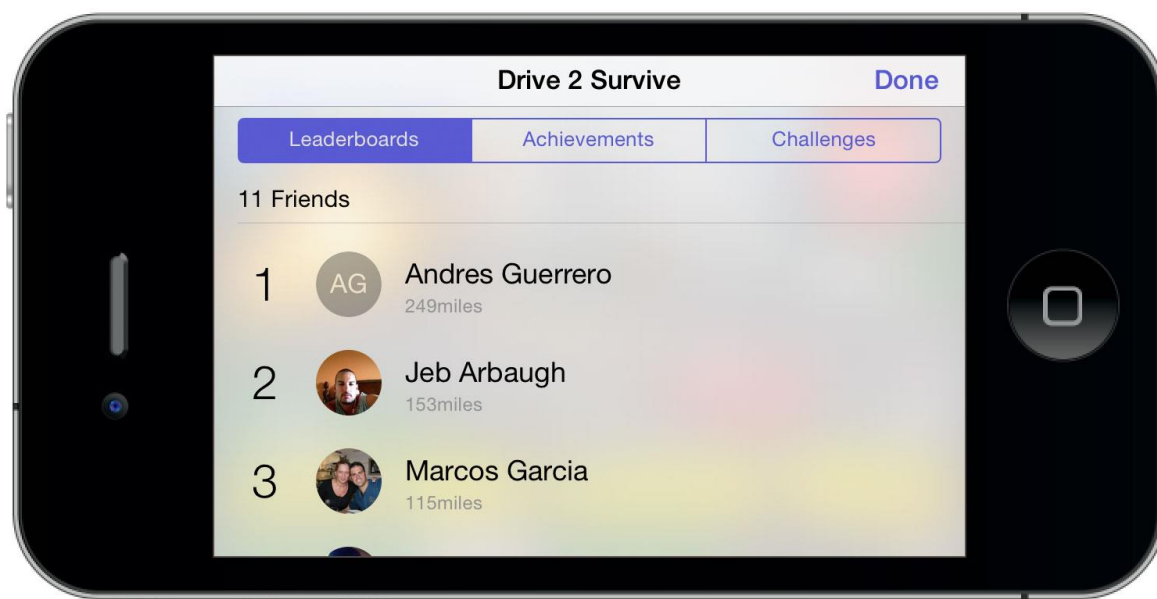


Figura 4.5 Pantalla de Puntajes “Game Center”

Cuando el usuario selecciona la opción “Scores” del menú principal, el juego se conecta con la comunidad “Game Center” para mostrar la tabla de puntajes y hazañas del juego.

La ventana de instrucciones es la más simple, sólo despliega la imagen con las instrucciones del juego como se puede ver en la figura 4.6:



Figura 4.6 Ventana de Instrucciones

Para programar esta ventana simplemente coloque la imagen con el texto y los iconos de las armas y el iPhone rotando. No se necesitó ninguna programación adicional.

La última ventana del menú principal es la de los créditos, tampoco fue complicado crear esta ventana ya que sólo muestra información sobre las personas que aportaron a este proyecto. Esto se puede ver en la figura 4.7:



Figura 4.7 Ventana de Créditos

Para crear esta ventana, se colocaron 3 componentes:

1. El logo del “Drive 2 Survive”
2. El logo de Oubii Games
3. El texto de los créditos

Cada componente es una imagen, y el logo de Oubii Games tiene un vínculo hacia el sitio web de Oubii Games.

La escena que se muestra en la figura 4.8 conecta el menú principal con el juego en sí. Se llama la pantalla de carga, y es necesaria para poder cargar todos los gráficos de la siguiente escena.

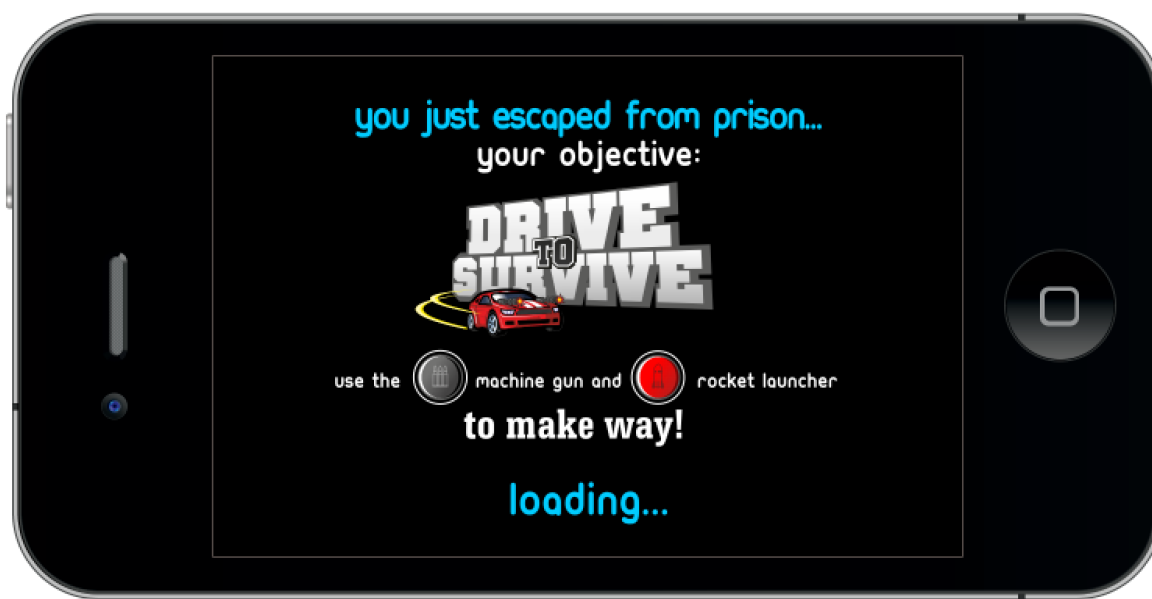


Figura 4.8 Escena de Carga

Cuando una escena carga muchos recursos como imágenes, sonidos especiales y sprites, se demora en aparecer. Por eso es importante tener una escena de “Loading...” para que el usuario no sienta la demora. Se aprovechó esta escena para también incluir un texto y distraer al usuario mientras se carga la escena del juego. El texto es una pequeña frase en la que se explica el objetivo del juego.

La siguiente escena que se carga es la más importante, el juego. Es la escena que mayor recursos consume ya que tiene varias imágenes, animaciones, sonidos especiales y música. La figura 4.9 muestra una captura de pantalla de la escena del juego en acción:



Figura 4.9 Juego en Acción

En la captura de pantalla se puede observar el auto principal, de color rojo, que controla el usuario con el acelerómetro. Cuando el usuario rota el teléfono para la izquierda o derecha, el auto se mueve en la misma dirección y con la misma velocidad con la que se hizo el movimiento. El objetivo del juego es mover el auto de un lado para el otro y así evitar el tráfico de la carretera. Para salir de aprietos, el usuario puede usar dos armas diferentes. Una metralleta con un cierto número de balas, que se recargan automáticamente poco a poco, y un lanza cohetes, con 3 cohetes. El usuario también puede encontrar cohetes en la carretera.

El usuario puede pausar el juego en cualquier momento, usando el botón de la parte superior derecha, la ventana mostrada en la figura 4.10 aparece:

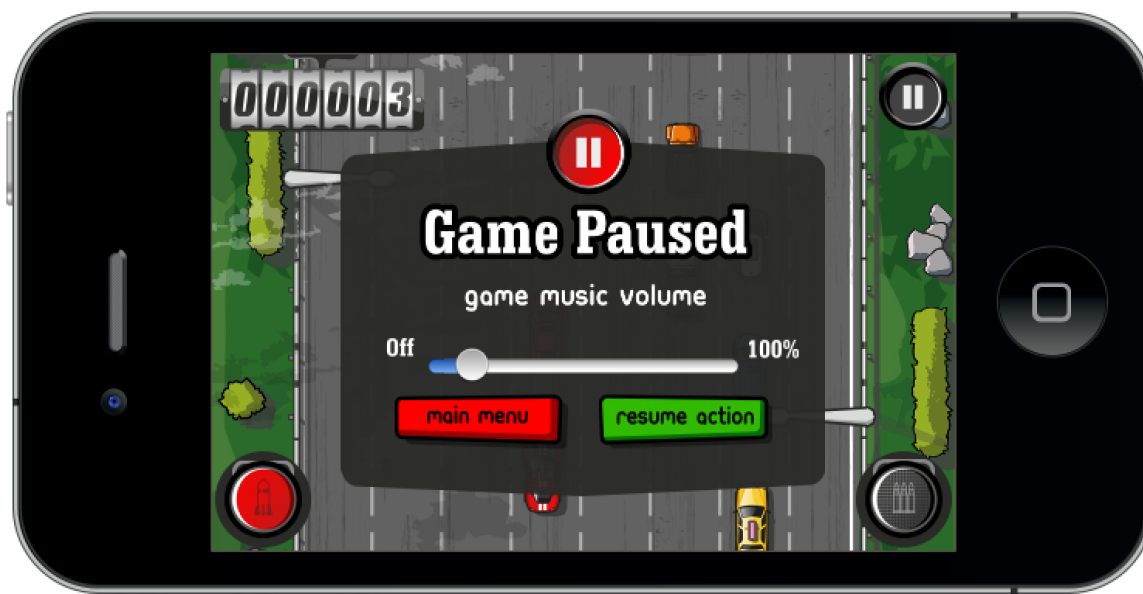


Figura 4.10 Juego en Pausa

En este estado, el usuario puede continuar, salir o regular el volumen del juego.

El juego se acaba cuando el auto del jugador se choca con algún otro vehículo. La ventana de “Game Over” aparece, como se muestra en la figura 4.11:

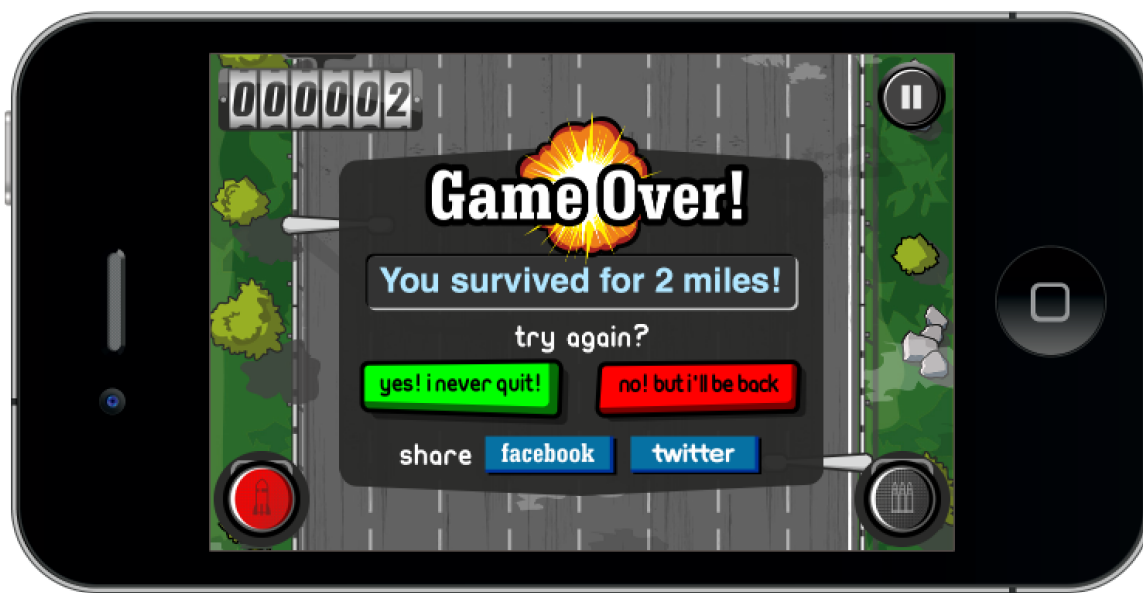


Figura 4.11 Final del Juego

Cuando el juego acaba, el usuario puede compartir su puntaje en las redes sociales Facebook o Twitter. El usuario también puede volver al menú principal, o volver a intentar jugar.

Flujo de Información y Organización de Clases

La comunicación entre los diferentes componentes se hace a través de variables globales ubicadas en el archivo principal (main.lua). Un ejemplo de esto son las configuraciones del juego, las mismas que se guardan en la pantalla de “Settings” y son utilizadas en el juego. El flujo de información se muestra en la figura 4.12:

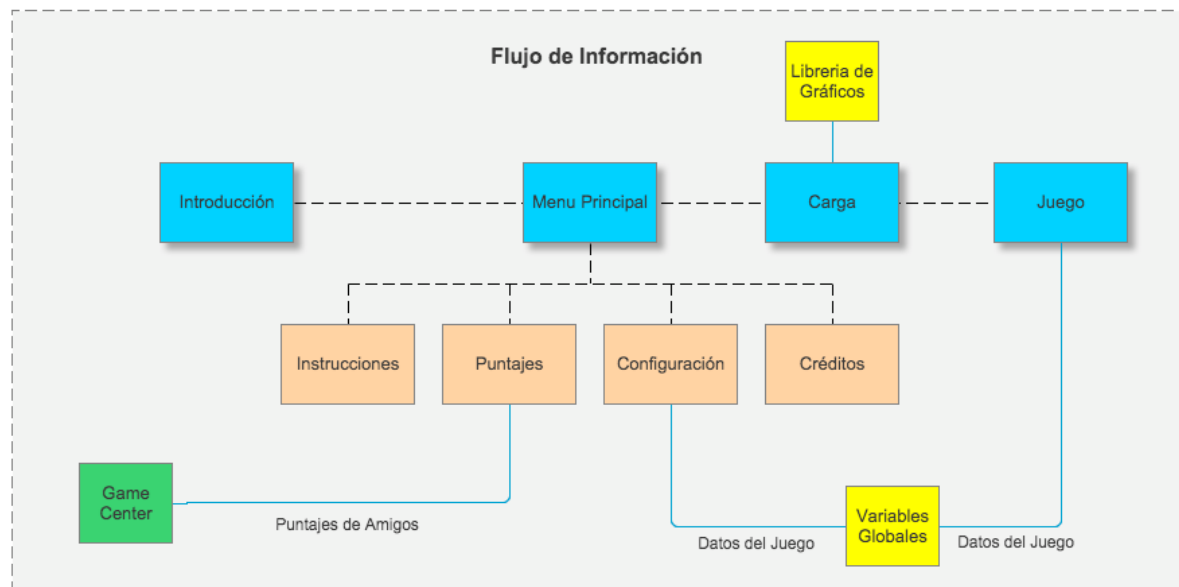


Figura 4.12 Flujo de Información Entre Módulos

La plataforma Corona SDK organiza los juegos en escenas. Todos los proyectos deben tener una escena principal, y en Corona SDK la escena principal debe ser un archivo llamado main.lua. Desde la escena principal se pueden definir variables globales, escenas adicionales y cargar librerías. Cada escena adicional es un archivo con extensión .lua. El juego “Drive 2 Survive” tiene 4 escenas adicionales, por lo tanto tenemos 5 archivos de escenas:

- Escena Principal: main.lua
- Escena Introducción: splash.lua
- Escena Menú: menu.lua
- Escena Carga: loader.lua
- Escena Juego: game.lua

En Lua técnicamente no hay clases, pero existe una forma de declarar objetos con métodos que simula una clase. De esta manera se organizaron las clases en paquetes. A continuación mostramos la estructura de archivos que tenemos en el proyecto:

com/ – paquetes de archivos fuente

graphics/ – gráficos del juego

music/ – música del juego

sounds/ – sonidos especiales del juego

main.lua – archivo principal del proyecto

config.lua – archivo de configuraciones del proyecto

build.settings – archivo para compilación

Icon.png – Icono de aplicación

Icon@2x.png - Icono de aplicación en resolución retina

Las clases y archivos Lua se encuentran organizadas en el paquete com, siguiendo los estándares de programación tradicionales para paquetes. Estos paquetes están organizados de la siguiente manera:

com.oubii.d2s.libs: librerías externas

com.oubii.d2s.audio: música y sonidos especiales

com.oubii.d2s.buttons: botones de pantalla

com.oubii.d2s.controls: clases de controles

com.oubii.d2s.data: datos abstractos

com.oubii.d2s.graphics: fondos, elementos visuales, etc.

com.oubii.d2s.objects: objetos como autos, piedras, etc.

com.oubii.d2s.screens: escenas

com.oubii.d2s.titles: textos en pantalla

com.oubii.d2s.weapons: armas del auto principal

com.oubii.d2s.widgets: componentes de iOS como el slider.

Una característica única del juego, que se incluyó para diferenciarse del resto es la lógica de colisiones. El objetivo de esto es que cuando los autos choquen entre sí, exista la ley de fuerza y reacción. Si un auto colisiona con otro, los dos deben reaccionar a esta fuerza. Esto fue uno de los mayores retos del juego, ya que requiere de varios cálculos matemáticos para que parezca real. Afortunadamente, ya hay un motor de física que nos hace todo el trabajo difícil. El motor de física al que me refiero se llama Box2D. Corona SDK incluye soporte nativo de este motor de física y solo es necesario declarar cada objeto, con una densidad y una forma. Por ejemplo, si queremos crear un juego con cajas que tengan densidad y reaccionen a la gravedad y al chocarse entre sí, solo debemos declarar cada caja como un objeto físico con un valor de densidad y una forma cuadrada. La declaración para hacer esto es la siguiente:

```
physics.addBody( object, { properties } )
```

En este caso, *object* es el objeto en la escena al que se aplica las leyes de física. Y *{properties}* son las propiedades de ese objeto, incluyendo las coordenadas que definen la forma. Para casos simples, como el de la caja, no hace falta incluir las coordenadas ya que la forma predeterminada es el cuadrado.

En el caso del juego “Drive 2 Survive”, la gravedad no es un factor que debemos incluir ya que la perspectiva del juego es una vista superior y la gravedad en este caso no aplicaría. Sin embargo, igual podemos aprovechar de este motor de física para las colisiones. Lo único que debemos hacer es declarar el coeficiente de gravedad en 0 m/s^2 , en lugar de 9.8 m/s^2 . Los objetos físicos que declaramos son los vehículos. Y cada vehículo tiene una forma diferente e irregular. Definir las coordenadas para cada uno de los objetos iba a tomar mucho tiempo, y era una tarea muy laboriosa. Es por esto, que se acudió a la herramienta Physics Editor, la cuál hizo el trabajo de crear las declaraciones de objetos físicos y lo único que se debía hacer era definir la forma del objeto en el editor gráfico de la herramienta. La figura 4.12 muestra como funciona Physics Editor:

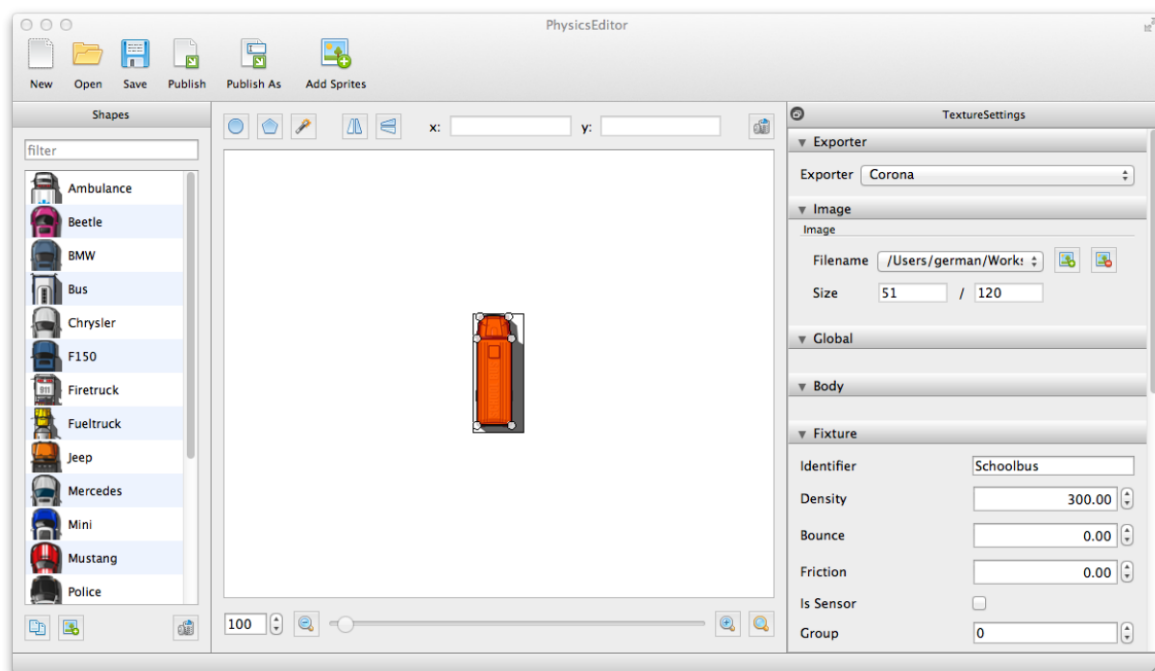


Figura 4.13 Herramienta Physics Editor

El ejemplo de la Figura 4.13 muestra uno de los vehículos, el bus escolar. Se puede observar las coordenadas creadas en cada punto importante. De esta herramienta se puede

exportar el código necesario para declarar a cada vehículo como un objeto físico y con la definición precisa de la forma.

Lógica Principal del Juego

La lógica principal del juego se enfoca en la aparición de los vehículos. Se usó el modelo incremental, es decir, se aumenta la cantidad de autos que aparecen mientras transcurre el tiempo. De esta manera mientras más avanza el usuario en el juego, más difícil se hace el juego. Para crear el algoritmo que es el centro de este juego, se consideraron cuatro diferentes factores, aquí se define cada uno:

Frecuencia: Se refiere a que tan rápido mostramos mas autos, esto también influye mucho en el juego ya que mientras mayor sea la frecuencia, mas difícil va a ser encontrar espacios para mover el auto principal.

Cantidad: Se refiere a la cantidad de vehículos que instanciamos cada vez. Mientras más vehículos instanciamos a la vez, mas difícil es el juego. Este factor tiene un límite. Si el límite no existiera, el juego se volvería imposible ya que la cantidad de autos supera el espacio disponible en la pantalla y el jugador no tiene espacio para poder esquivar los vehículos.

Variedad: Se refiere a los diferentes tipos de autos que se dispone, se crea categorías para colocar cada auto, en base a sus características. Esto es importante para que el juego muestre siempre una variedad de autos y no se repitan mucho. Esto aporta a la percepción del realismo del juego hacia el usuario.

Ubicación: Se refiere a la ubicación de cada auto en el juego. La ubicación inicial de cada vehículo se basa en los 8 carriles que existen en el juego y en la profundidad dentro de cada carril. Este factor define el espacio disponible para que el usuario pueda esquivar los autos.

El algoritmo se compone primero por la frecuencia. Declaramos el método que se ejecuta para instanciar vehículos. Cada vez que se ejecuta este método, se crea una ola de vehículos en diferentes lugares, por eso llamamos a este método “wave()”. El factor de la frecuencia está en que tan seguido llamamos a este método. En cada ola de vehículos que instanciamos primero definimos cuantos vehículos se crean, y es por eso que entra aquí el factor de cantidad. La cantidad de vehículos se incrementa después de cierto tiempo hasta llegar a su límite. Después de saber la cantidad de vehículos que vamos a instanciar, definimos por cada uno el resto de factores: Variedad y Ubicación. La variedad se define primero, escogemos de manera aleatoria una categoría y nuevamente de manera aleatoria un vehículo de esa categoría. Finalmente escogemos de manera aleatoria el carril en el que va a aparecer el vehículo y la profundidad dentro de un rango.

Inicialmente, cuando se aplicó este algoritmo, tenía sus fallas y se afinaron los factores mientras se realizaron pruebas de ensayo y error.

Integración con Facebook, Twitter y Game Center

La idea de integrar el juego con Facebook no era únicamente para que el usuario pueda compartir con sus amigos el puntaje que logró hacer en el juego, sino también como una estrategia de marketing para promocionar el juego. Cuando un usuario compartía su puntaje en Facebook, la publicación incluía el vínculo al sitio web de Oubii Games y también al lugar de descargar en iTunes. En la figura 4.14 se muestra cómo aparece la ventana de Facebook para publicar un puntaje:

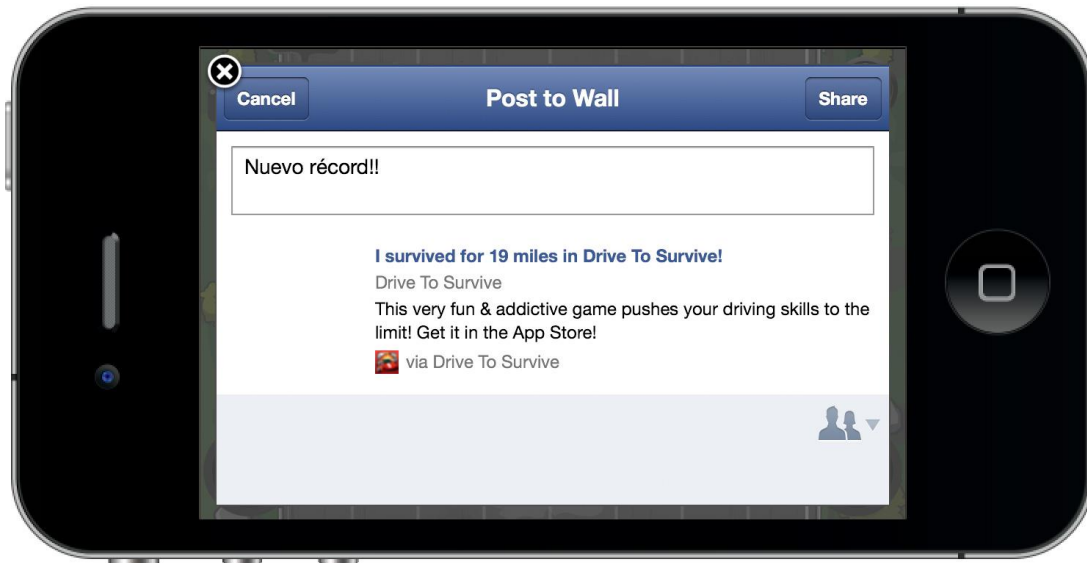


Figura 4.14 Integración con Facebook

Para poder integrar el juego con Facebook se tuvo que buscar ejemplos en internet y probar diferentes valores para las propiedades. Con los parámetros correctos el resto de la integración es automática.

La integración con Twitter tiene la misma idea de la integración con Facebook, permitir al usuario compartir con sus seguidores el puntaje logrado en el juego y al mismo tiempo promover el juego. En el caso de Twitter, la integración fue diferente. La plataforma Corona SDK no tiene integración nativa con la aplicación Twitter como lo tiene con Facebook. Sin embargo, si fue posible integrarlo a través de un componente de iOS llamado Webview. Un Webview es un contenedor que muestra un url específico. Como Twitter tiene integración con todos los browsers, usar el componente Webview fue la mejor solución. La figura 4.15 muestra la integración de Twitter con el componente Webview:

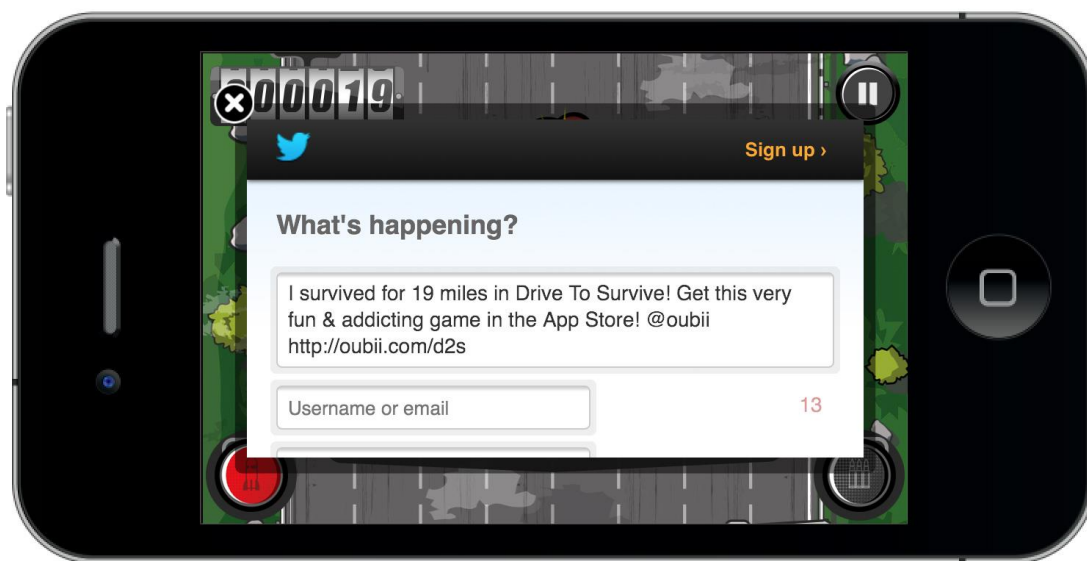


Figura 4.15 Integración con Twitter

La integración de “Game Center” es muy sencilla. Apple desarrollo esta comunidad con un API para que sea fácil de integrar para los programadores. El primer paso fue entrar a la herramienta “iTunes Connect”. Esta herramienta es exclusiva para desarrolladores con una membresía activa en el “Apple Developer Program”. Utilizamos esta herramienta para crear y administrar las aplicaciones que subimos al App Store. En una de las opciones de esta herramienta, podemos configurar a nuestra aplicación para que utilice “Game Center”. Podemos crear una o varias tablas de puntaje e inclusive crear hazañas, en inglés llamados “Achievements”. La figura 4.16 de la configuración de tablas de puntaje en la herramienta “iTunes Connect”:

Leaderboards

Leaderboards

Leaderboards allow users to view the top scores of all Game Center players of your app. Leaderboards that are live for any app version cannot be removed.

[Add Leaderboard](#)
[Move All Leaderboards into Leaderboard Sets](#)
[Manage Scores and Players](#)
[Delete Test Data](#)

1 Leaderboard

	Reference Name	Leaderboard ID	Type	Default	Status
☰	High Scores	HighScores	Single	<input checked="" type="radio"/>	Live

Figura 4.16 Configuración de Tablas de Puntajes de “Game Center”

Podemos crear varias tablas de puntaje, sin embargo, para el juego de “Drive 2 Survive” solo necesitamos una. En la figura se puede ver la tabla de puntajes que creamos para usar en el juego. En el contexto de la programación, para poder ingresar un puntaje a esta tabla, solo debemos hacer una llamada simple cada vez que el usuario pierde; el sistema de “Game Center” se encarga del resto. El ejemplo en la figura 4.16 muestra la llamada que hacemos al API de “Game Center” para ingresar el puntaje del usuario cuando pierde:

Achievements

An achievement is a distinction that a player earns for reaching a milestone, or performing an action, defined by you and programmed into your app. Once an achievement has gone live for any version of your app, it cannot be removed.

[Add Achievement](#)

20 Achievements

	Reference Name	Achievement ID	Points	Status
☰	1st Car Destroyed with Machine Gun	1_kill_machinegun	5	Live
☰	1st Police Chase	1_police_chase	5	Live
☰	1st Police Car Destroyed	1_police_destroyed	10	Live
☰	2nd Police Car Destroyed In a single game	2_police_destroyed	20	Live
☰	3rd Police Car destroyed in a single game	3_police_destroyed	30	Live
☰	Car Destroyed with Rocket	1_kill_rocket	5	Live
☰	1st Accident	1_accident	5	Live
☰	2nd Accident in a single game	2_accident	10	Live
☰	3rd Accident in a single game	3_accident	20	Live
☰	Survive 25 miles	25_miles	5	Live
☰	Survive 50 miles	50_miles	10	Live
☰	Survive 100 miles	100_miles	20	Live
☰	Survive 250 miles	250_miles	30	Live
☰	Survive 500 miles	500_miles	40	Live
☰	Survive 1000 miles	1000_miles	50	Live
☰	Survive 5000 miles	5000_miles	100	Live
☰	Destroy a Fuel Truck	destroy_fuel_truck	25	Live
☰	First 25 Miles - No Weapons	25_miles_no_weapons	10	Live
☰	First 50 Miles - No Weapons	50_miles_no_weapons	20	Live
☰	First 100 Miles - No Weapons	100_miles_no_weapons	30	Live

Figura 4.17 Pantalla de Puntajes “Game Center”

Cuando se publicó el juego en el App Store, la tabla de puntajes empezó a mostrar información de usuarios que jugaban el juego. Algunos lograron llegar a las primeras posiciones con puntajes de más de 150 millas.

Gráficos del Juego

Los gráficos del juego fueron creados por un ilustrador profesional, su nombre es Artem Dvorzhak. El ilustrador es de Ucrania y se logró ubicarlo a través del sitio web Dribbble (<http://www.dribbble.com>). Dribbble es una comunidad de diseñadores e ilustradores donde publican sus trabajos y reciben comentarios del resto. Ya existía una relación profesional con el ilustrador, ya que se le contrató para el diseño del logotipo de

Oubii Games. El portafolio del ilustrador muestra muchos logotipos con ilustraciones de personajes y se decidió hacerle una propuesta. El presupuesto era de \$2000, y el aceptó. El ilustrador nunca había creado gráficos para juegos, sin embargo, le gustó la idea y así se empezó a trabajar en conjunto. El primer paso fue crear un ejemplos y decidir en un estilo. La primera muestra que el ilustrador entregó fue del vehículo principal del juego como se puede ver en la figura 4.18:



Figura 4.18 Muestra de Vehículo Principal

Esta primera muestra fue la escogida y se empezó a integrar los gráficos mientras se desarrollaba el juego. El ilustrador entregaba nuevos gráficos cada 3 – 5 días y se los incluía en el juego progresivamente. A continuación la lista de todos los gráficos que el ilustrador creó:

- Fondo del menú principal
- Botones del menú principal
- Ventanas del menú principal
- Pantalla de carga
- Vehículos
- Carretera
- Paisajes y sus elementos
- Botones de armas

- Visor de puntaje
- Ventanas de pausa y de fin del juego
- Explosiones

Para la integración de los gráficos de los vehículos, se creó un sprite con todos los vehículos como se puede ver en la figura 4.19:

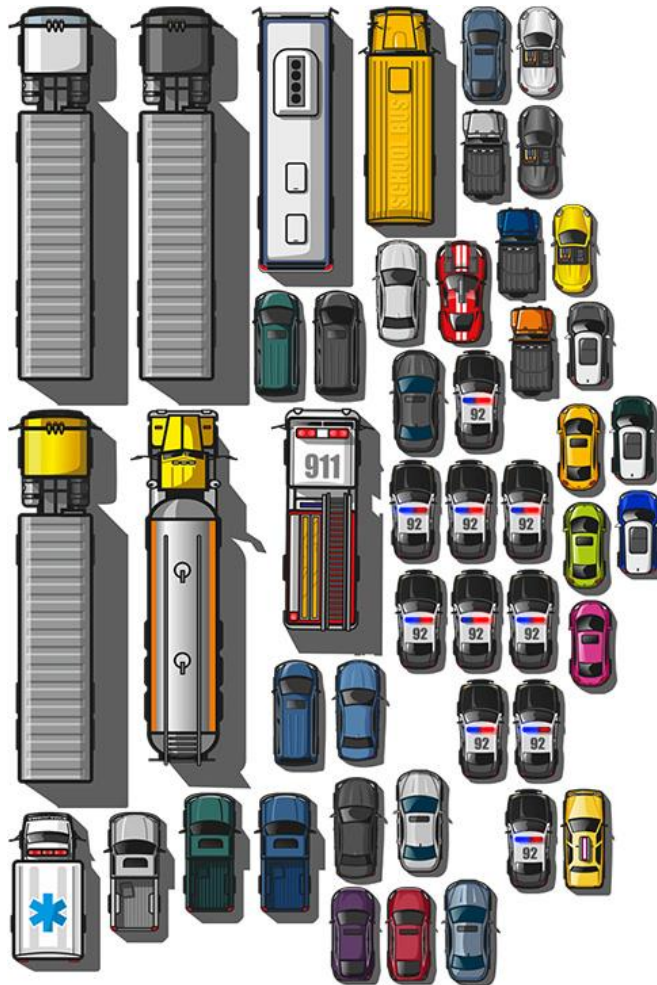


Figura 4.19 Sprite de los Vehículos del Juego

En la figura 4.19 se muestran todos los vehículos del juego, y se puede ver el vehículo policial repetido varias veces. La razón por esas repeticiones es porque el vehículo de la policía es el único con animación, la sirena se ilumina de un lado al otro.

Desarrollo del Sitio Web

Antes de empezar el desarrollo del sitio web, se creó el diseño gráfico para cada página. Se usó la herramienta Photoshop para crear los diseños.

El primer paso en el desarrollo del sitio web fue el hospedaje y compra del dominio; se compraron dos dominios: www.oubii.com, www.oubiigames.com.

La idea detrás de los dos dominios es facilitar el acceso al sitio web. Ambos dominios apuntan al mismo servidor y el mismo sitio web. Para el hospedaje, se contrató a la compañía DigitalOcean. Esta compañía ofrece hospedaje en máquinas virtuales Linux. Tienen un costo de \$5 mensuales por un servidor con las siguientes características:

- 512 MB de RAM
- 1 Procesador
- 20GB disco de estado sólido (SSD)
- 1TB mensual de transferencia de datos

Las características bastan para un sitio web como el de Oubii Games. La cantidad de tráfico estimada se calcula que es entre 200 y 300 visitas diarias inicialmente. La otra razón por la que se usó este proveedor es por su fácil configuración que va a permitir aumentar la capacidad del servidor mientras aumentan el número de visitas.

El segundo paso para el desarrollo del sitio web es la configuración de un repositorio para versionamiento. La mejor solución en el mercado actual es GIT, con hospedaje en sus propios servidores a través del sitio web llamado GitHub. GitHub permite hospedar cualquier número de repositorios de manera segura. La cuenta básica de este servicio cuesta \$7 el mes e incluye hasta 5 repositorios. El plan de Oubii Games es tener un repositorio para el sitio web, y el resto para cada juego creado.

Con el hospedaje y el repositorio configurado, se llevó a cabo la instalación de Drupal 7 en el servidor. Drupal 7 es un manejador de contenidos de software libre basado

en PHP. La instalación de Drupal 7 empezó con la configuración apropiada del servidor web Apache con PHP y la instalación de MySQL. La figura 4.20 muestra la configuración del servidor:

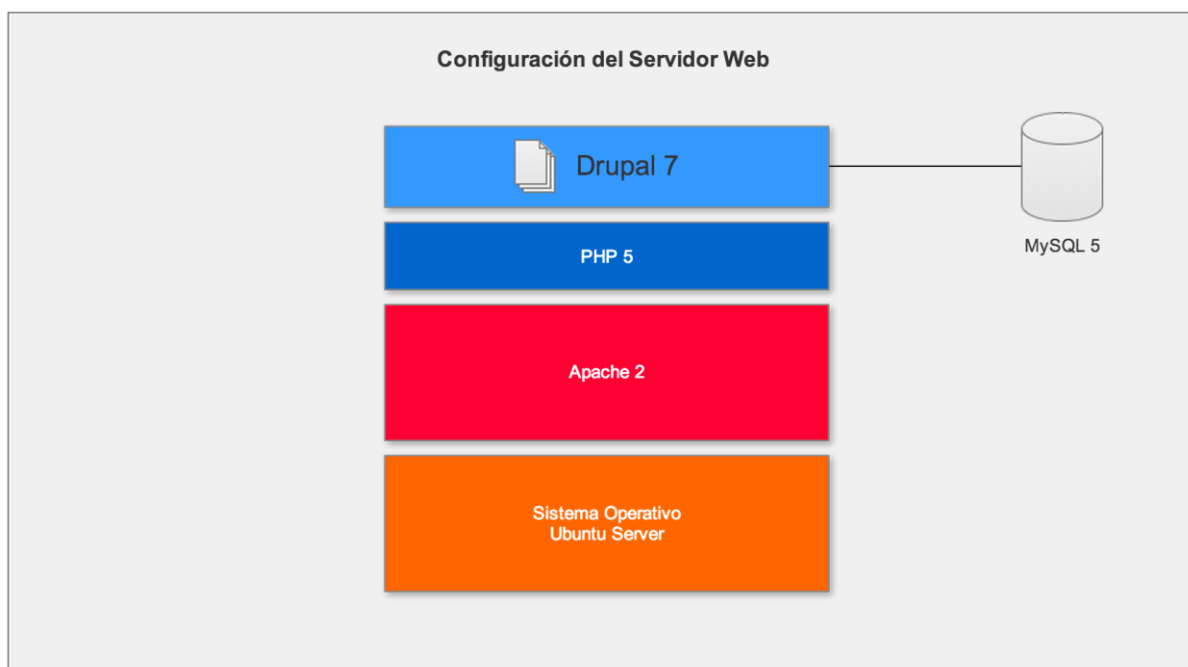


Figura 4.20 Configuración de Servidor

El sitio web de Oubii Games fue configurado en Apache como un “Virtual Host”, y recibe todas las llamadas por el puerto 80. Ya configurado el servidor con Drupal instalado, se procedió a programar el sitio web.

En Drupal, el contenido se categoriza en tipos; un tipo de contenido contiene campos como el título y cualquier campo adicional que se necesite. Drupal ya incluye dos tipos de contenido:

Página: El tipo de contenido más importante, representa una página web simple con el campo de título y el campo del contenido de la página. El campo de contenido es un bloque de texto que puede tener formato.

Blog: El blog tiene los mismos campos que la Página con la diferencia que se puede crear un resumen del campo del contenido. El resumen se crea automáticamente y se puede mostrar en otras páginas web del sitio.

El tipo de contenido llamado Página permite crear páginas web sencillas. Cada instancia de un tipo de contenido se llama nodo.

Drupal permite crear más tipos de contenido en caso de ser necesarios. Para el sitio web de Oubii Games se creó un nuevo tipo de contenido llamado Juego. La empresa únicamente tiene un juego ahora, sin embargo el sitio web se desarrolló para que se puedan agregar más juegos después. Por ahora, se creó un solo nodo de tipo Juego con el título “Drive 2 Survive”, se agregaron campos como una imagen para el ícono del juego y una fecha para la publicación del juego. Con este contenido inicial, se procedió a crear las páginas web del sitio.

Mapa del Sitio y Navegación

El sitio web se compone de seis secciones:

1. Página de inicio: la página principal que se presenta a todos los usuarios que entrar al sitio web usando cualquier de los dos dominios creados.
2. Sobre Nosotros: una página informativa con una breve descripción de la compañía Oubii Games.
3. Games: La lista de juegos, inicialmente sólo uno con una página para cada juego individual.
4. Blog: El listado de todas las publicaciones del blog, inicialmente sólo una publicación. Esta sección también incluye una página por cada publicación.
5. Contacto: Ventana con formulario para contactarse con el equipo de Oubii Games.

6. Campaña de Marketing: Una página promocional para la campaña de marketing.

Página de Inicio

La página de inicio es la más importante del sitio web, ya que incluye los componentes más importantes para el usuario:

- La imagen principal con capturas de pantalla del juego “Drive 2 Survive”
- El slogan de la empresa
- Una pequeña sección con una descripción resumida del juego
- Un link a la página del juego y un botón de descarga que lleva a los usuarios a la tienda virtual de iTunes donde se lo puede comprar
- Finalmente, la lista de publicaciones del blog, de igual manera con un link a la página individual de cada publicación

En la figura 4.21 se muestra el resultado de la construcción de la página principal del sitio:



Figura 4.21 Página Inicial del Sitio Web

Sobre la Compañía

Una sola página web con una breve descripción de la empresa. Para crear este contenido se buscó un servicio de escritura profesional para que el contenido sea de alta calidad. La figura 4.21 muestra una captura de pantalla del resultado:

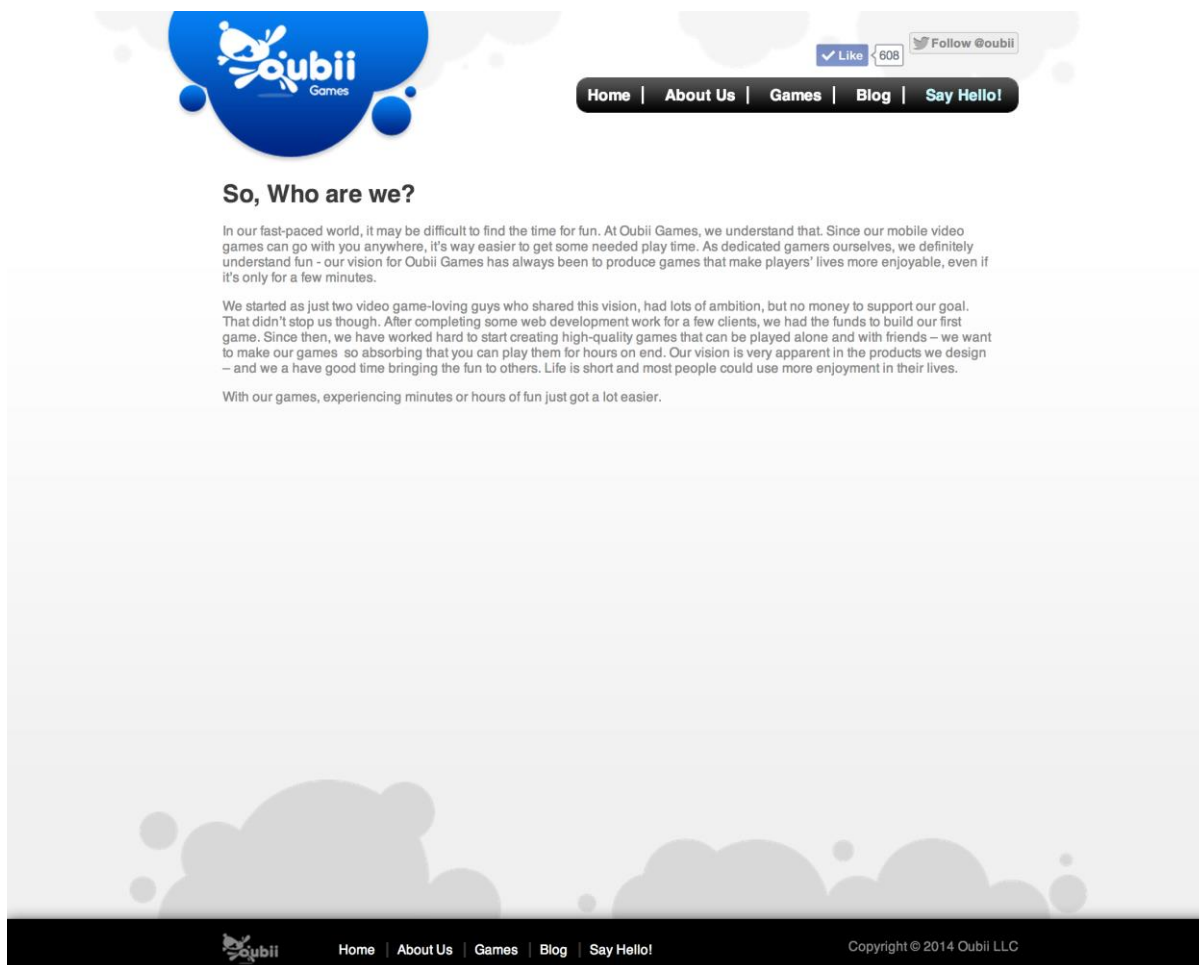


Figura 4.22 Página Sobre la Empresa

Blog

La sección del blog incluye la lista de todas las publicaciones y la página individual para cada publicación. Esta sección es necesaria para informar a los usuarios sobre actualizaciones y noticias de la empresa.

Estas páginas se crearon a través del manejador de contenidos. El resultado se presenta en la figura 4.23:

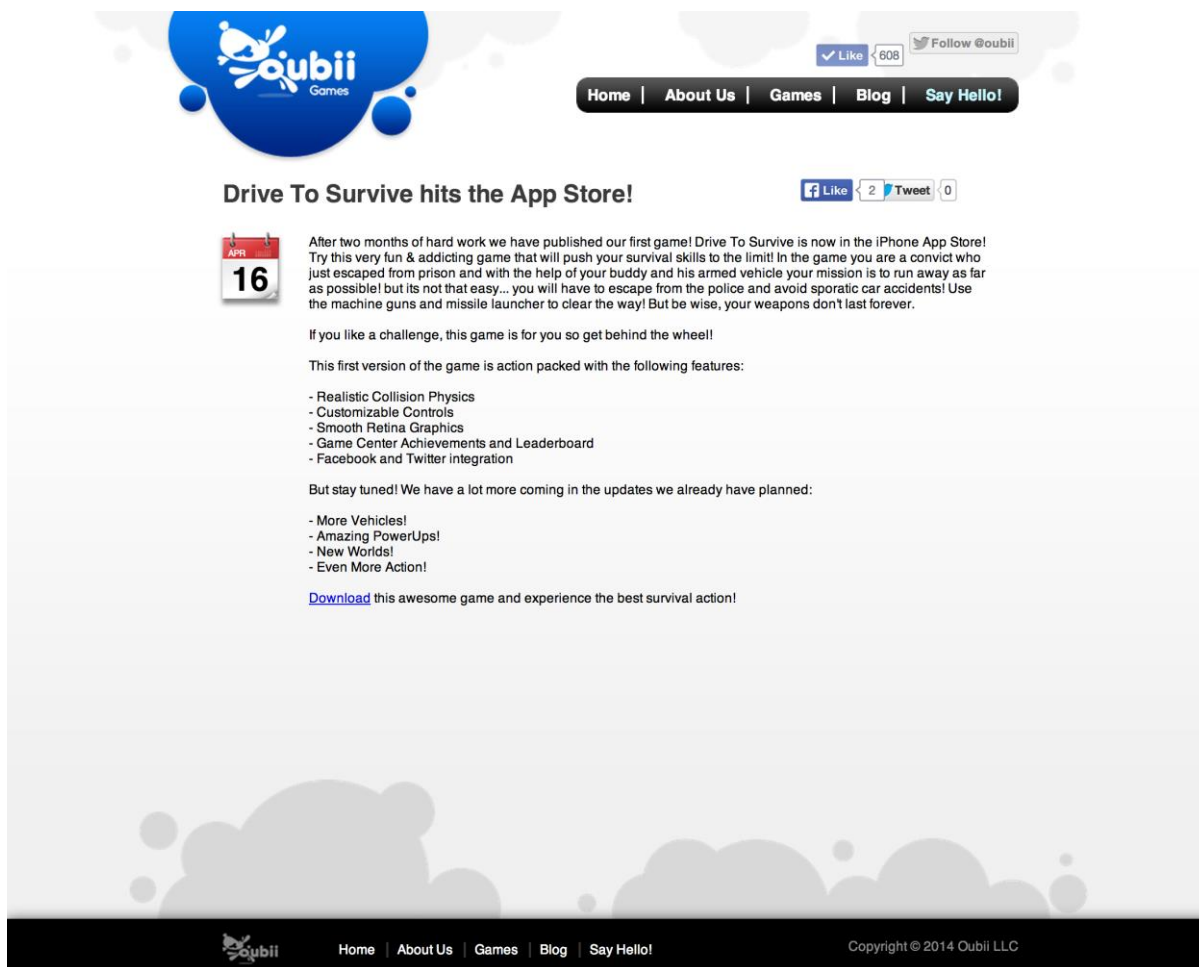


Figura 4.23 Página de publicación del blog

Contacto

Es importante para Oubii Games tener alguna forma de contacto con sus usuarios, y por eso se creó un formulario de contacto que se despliega en una ventana modal. Al completar el formulario y enviarlo, la información recibida se envía en un correo electrónico a la dirección hello@oubii.

Para crear esta funcionalidad, fue necesario instalar un módulo de la comunidad de Drupal que se llama Webforms. Éste módulo permite crear formularios para usuario, generar reportes y enviar los resultados de cada formulario lleno a un correo específico. La figura 4.24 muestra el formulario que fue creado para contactarse con nosotros:

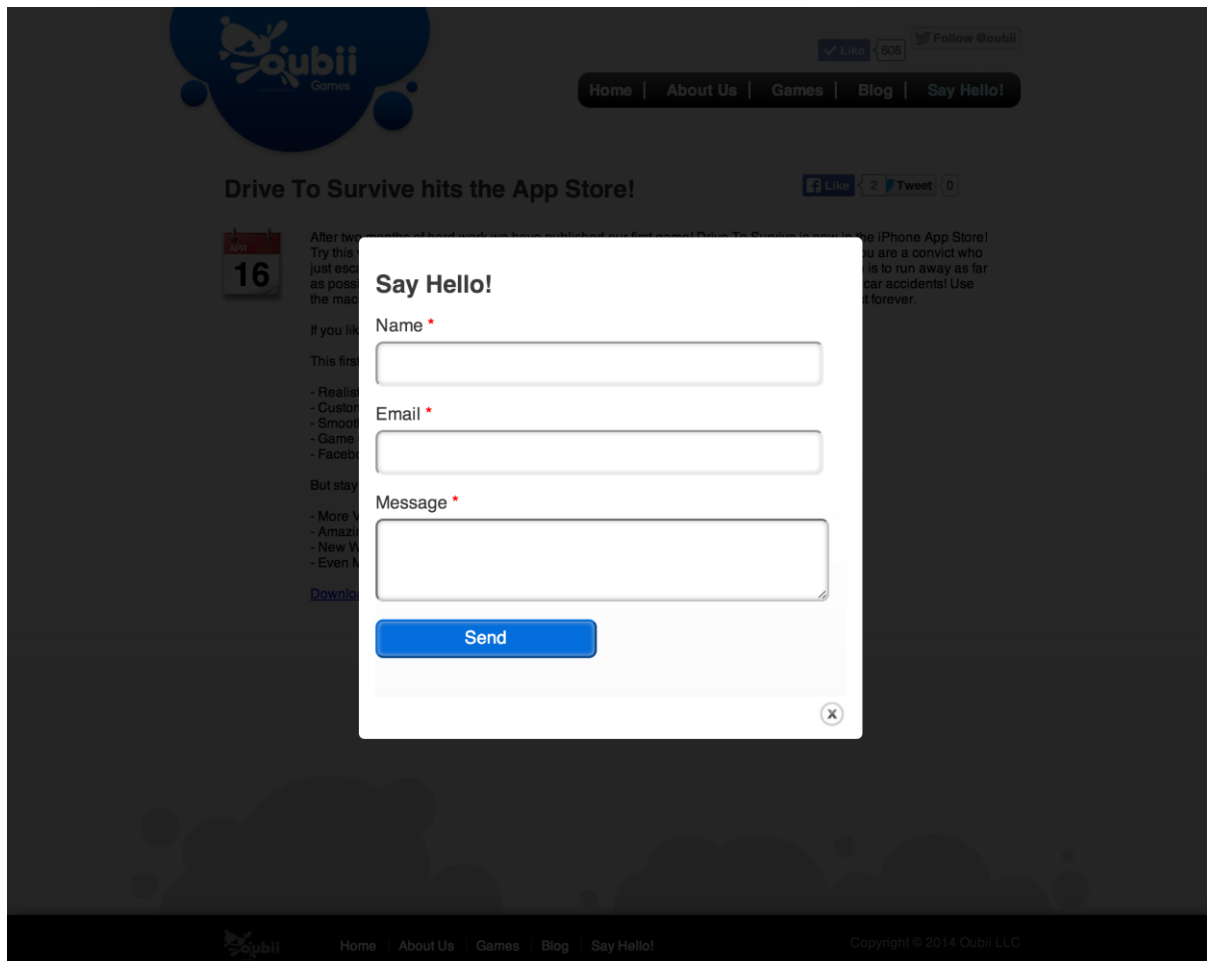


Figura 4.24 Formulario de Contacto

Este formulario también es útil para cuando los usuarios quieren reportar errores en el juego. Tuvimos dos casos en los que usuarios llenaron el formulario reportando problemas y pudimos resolverlos inmediatamente.

Juegos

La sección de juegos es muy parecida a la sección del blog. Por un lado tenemos una página web con la lista de todos los juegos (actualmente uno) y al hacer click en el título del juego el usuario puede ver en una página web separada los detalles del juego. En esta página web se puede ver como se hizo uso del nuevo tipo de contenido que se creó. La figura 4.25 muestra la página web del juego “Drive 2 Survive”.



Figura 4.25 Página de “Drive 2 Survive”

Esta página muestra información del juego como capturas de pantalla, un resumen de lo que se trata el juego y también un video del trailer del juego.

Integración con Redes Sociales

La integración con las redes sociales tiene como objetivo aumentar la cantidad de seguidores de Twitter y Facebook. Se colocaron los botones de “Like” y “Follow Us” en la cabecera del sitio web para que sea visible en todas las páginas. Al hacer esto aumentamos la visibilidad del juego dentro de las redes sociales.

CONTROL DE CALIDAD, PUBLICACIÓN Y MARKETING

Control de Calidad

Al finalizar la etapa de desarrollo del juego y del sitio web de Oubii Games, se empezó la etapa de control de calidad. Para poder asegurarnos que el juego no tenga problemas al estar publicado, fue necesario encontrar al menos cinco personas para realizar pruebas y arreglar cualquier problema. Se buscaron personas dentro del rango de edad al que apunta el proyecto. Cada persona debía tener un iPhone o iPod para poder colaborar en este proceso, y así el siguiente paso fue instalar el juego en los dispositivos de las cinco personas. Apple tiene muchas restricciones cuando se trata de instalar una aplicación no publicada en otros dispositivos, para evitar la piratería. Para poder realizar pruebas de una aplicación no publicada, es necesario crear lo que se llama un perfil de pruebas. El perfil de pruebas es como un certificado que permite a un iPhone instalar la aplicación que se especifica en el certificado. Cada certificado tiene una validez de un año. Por lo tanto, era necesario instalar el perfil de pruebas en cada dispositivo, a través de la aplicación de iTunes.

En este proceso encontramos una complicación, y es que cada persona que va a probar está en un lugar diferente y para algunos era complicado instalar el perfil de pruebas por su cuenta. Afortunadamente existe una solución mucho más fácil para realizar pruebas en dispositivos remotos, la solución se llama Test Flight. Test Flight es una herramienta en internet que facilita la creación e instalación de perfiles de prueba.

Usando Test Flight, lo único que hay que hacer es crear una cuenta e invitar a diferentes personas a ser parte del proceso de pruebas a través de un correo electrónico. Las personas invitadas pueden recibir el correo electrónico en sus iPhones que incluye un link. Cuando el usuario entra al link, se instala el perfil automáticamente y así el

dispositivo está listo para instalar el juego. Esta herramienta permite al desarrollador subir el juego al sistema, y éste automáticamente envía un correo electrónico con el link de descarga a todas las personas invitadas. Lo único que la persona debe hacer es abrir el link y automáticamente se va a instalar el juego en el dispositivo. Este proceso se puede repetir con varias versiones del juego y es muy útil ya que el desarrollador puede publicar varias versiones del juego mientras se van solucionando los problemas. Las personas invitadas a probar el juego van a recibir un correo electrónico por cada versión que el desarrollador publica en el sistema. Cada publicación tiene una breve descripción que el desarrollador puede incluir, con la lista de los problemas que se solucionaron. En la figura 5.1 se puede observar una captura de pantalla de esta herramienta que muestra las diferentes versiones publicadas:

The screenshot shows the TestFlight interface for the app 'Drive2Survive'. The 'Builds' section is active, displaying a table of beta builds. The table has the following columns: Builds, Added Date, Built For, SDK, dSYM, Crashes, Feedback, and Installs. The data in the table is as follows:

Builds	Added Date	Built For	SDK	dSYM	Crashes	Feedback	Installs
1.0.2	May 8, 2012	iPhone	×	×	0	0	2
1.0.1	April 18, 2012	iPhone	×	×	0	0	1
1.0.0	April 13, 2012	iPhone	×	×	0	0	1
1.0	April 3, 2012	iPhone	×	×	0	0	7
1.0	March 27, 2012	iPhone	×	×	0	0	5

The footer of the page contains navigation links for TestFlight (Status, Change Log), Developer (SDK, API), Products (TestFlight, Desktop App), Help (Support, FAQs, Tutorials), and Connect (Twitter, Facebook). It also includes links for Terms of Use, Privacy Policy, and DMCA.

Figura 5.1 Sistema Test Flight

En la figura 5.1 se puede ver las diferentes versiones creadas, e inclusive el número de personas que han instalado cada versión.

Al usar esta herramienta, se pudieron detectar los problemas que existían en el juego y realizar los cambios necesarios. De las cinco personas que probaron el juego, cuatro detectaron entre dos y cuatro problemas cada uno. Es por esto que se realizaron ajustes y se publicaron nuevas versiones para que las personas que detectaron problemas puedan verificar que esos problemas ya no existían. Los problemas que se encontraron en este problema fueron triviales, en su mayoría errores visuales y pocos errores de lógica del juego. Por ejemplo, la primera versión del juego permitía al jugador mover el vehículo fuera del límite de la pantalla bajo circunstancias específicas. Otro error que se encontraba en las primeras versiones, era un error en el contador de millas. Los gráficos en el contador no se mostraban correctamente después de un número específico. La corrección mas grande que se hizo, en base a las primeras pruebas, fue bajar el nivel de dificultad del juego ajustando los parámetros en el algoritmo principal del juego. En la figura 5.2 se puede observar un ejemplo del correo electrónico que llega a todos los invitados que están realizando las pruebas:

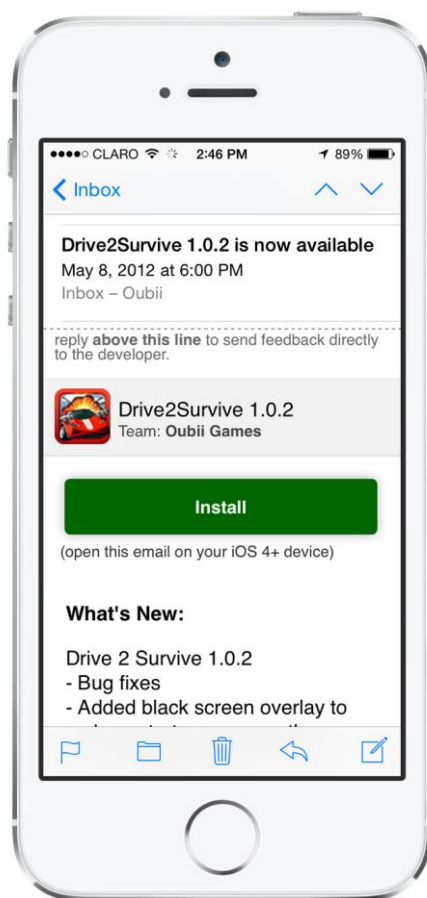


Figura 5.2 Ejemplo de Correo Electrónico de Test Flight

Test Flight nos ayudó a simplificar el proceso de control de calidad y poder publicar un juego más estable y libre de errores.

Publicación del Juego en el App Store

La publicación de una aplicación en el App Store es un proceso cuidadoso ya que cualquier error puede significar que la aplicación sea rechazada y en ese caso hay que reiniciar el proceso nuevamente. Este proceso dura aproximadamente de 8 a 10 días desde el momento en el que se sube el archivo compilado de la aplicación.

Requisitos Para Subir Aplicación

Todo empieza con el registro del nombre y la información detallada del juego. A continuación tenemos la lista de todos los campos que se deben llenar para poder publicar una aplicación:

- Nombre del App: El nombre de la aplicación
- SKU: El identificador de producto interno de la aplicación
- Descripción: La información detallada de lo que se trata el juego
- Palabras claves: Las palabras clave que se usan para el motor de búsqueda
- URL de Soporte Técnico: Una dirección web para que los usuarios puedan contactarse con soporte técnico
- URL de Marketing: Una dirección web para obtener más información del juego
- Categoría: La categoría principal a la que pertenece el juego
- Subcategoría 1: La primera subcategoría dentro de la categoría principal
- Subcategoría 2: La segunda subcategoría a la que pertenece el juego, dentro de la categoría principal.
- Capturas de Pantalla: Al menos 3 capturas de pantalla del juego para que los usuario sepan de que se trata.
- Icono: El ícono del juego, que se muestra en el iPhone cuando se instala y también en el App Store.
- Precio: El nivel de precio para la aplicación. Apple tiene una lista que va desde \$0 hasta mas de \$99.99. El precio no puede ser un valor específico.

Adicionalmente a esta información básica, Apple tiene una gran cantidad de reglas que se deben cumplir para que la aplicación no sea rechazada. A continuación encontramos la lista de las reglas más importantes:

- La aplicación no debe tener palabras reservadas en el nombre. Por ejemplo: Apple, iTunes, iPhone, iPad15.
- La aplicación no puede mostrar contenido explícito. Esto quiere decir que no se aceptan aplicaciones con contenido pornográfico.
- La aplicación no puede ser muy simple. Esta regla es la más controversial ya que la definición de simple es muy ambigua.
- La aplicación no puede consumir mucha memoria. El desarrollador debe realizar todas las pruebas necesarias para asegurarse que la aplicación no tenga fugas de memoria.
- La aplicación no puede tener funciones ocultas. Existen casos en los que el desarrollador esconde una función que no está permitida.
- La aplicación no debe violar derechos de autor en su contenido. Todas las imágenes, videos, etc deben ser propios del desarrollador o tener derechos para publicarlo.
- La aplicación no debe consumir de manera excesiva la batería del dispositivo. La duración batería es una de las características más importantes de cualquier teléfono. Apple se asegura que todas las aplicaciones en su tienda virtual no tenga un consumo excesivo del procesador, que puede llevar a drenar la batería rápidamente.

Apple puede ser bastante exigente en algunos casos, a tal punto que seguir estas reglas básicas no garantiza que la aplicación se apruebe.

Proceso de Aprobación y Publicación

Cada aplicación que se intenta publicar en el App Store tiene un estado. Cuando la aplicación está publicada, el estado se llama “Ready for Sale”. Pero antes de llegar a este estado, la aplicación debe pasar por diferentes etapas; y cada etapa tiene un estado. La siguiente figura muestra los diferentes estados:

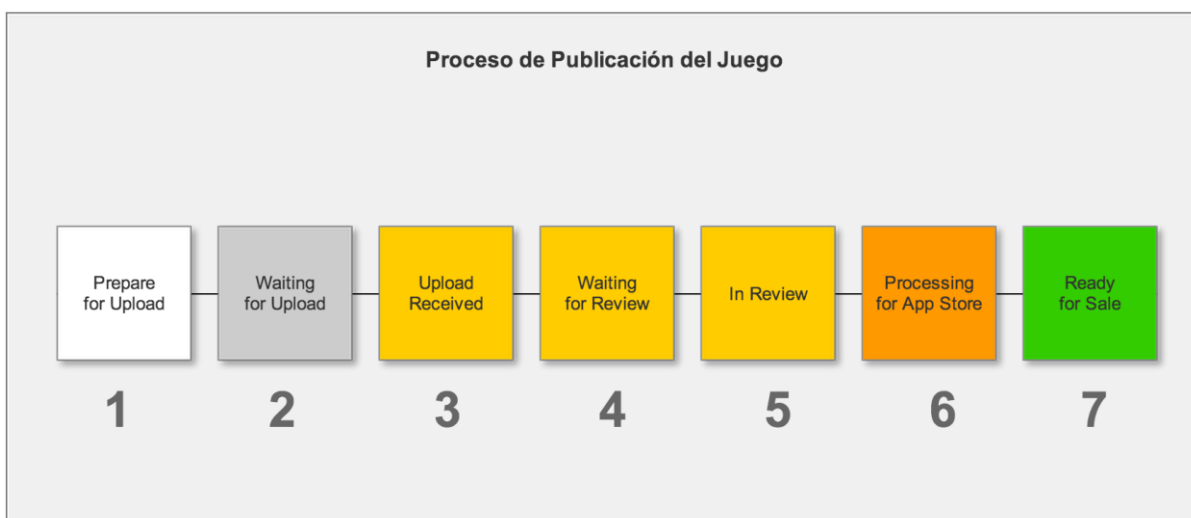


Figura 5.3 Proceso de Publicación del Juego

A continuación describimos cada estado de la publicación del juego “Drive 2 Survive”:

- **Prepare for Upload:** La primera etapa de todo el proceso, cuando “Drive 2 Survive” estaba en esta fase, faltaba ingresar la información del juego y las imágenes requeridas.
- **Waiting For Upload:** Una vez ingresada toda la información y las imágenes, se debe iniciar la carga del archivo compilado del juego al sistema. Para hacer esto Apple requiere que se use un programa llamado Application Loader. Este programa está incluido en el paquete del ambiente de desarrollo XCode. Al abrir el programa, se ingresa la información de la cuenta de Apple y automáticamente se muestra la opción de carga. Para el caso de “Drive 2

Survive”, solo se necesitó de aproximadamente 4 minutos para completar la carga ya que el archivo compilado solo pesaba 10 mb.

- **Upload Received:** Una vez recibido el archivo compilado del juego, el sistema de Apple se demora unos segundos mostrar este estado y después de unos minutos el estado cambia a “Waiting for Review”
- **Waiting for Review:** Esta es la etapa que más se demora, significa que el juego está listo para la revisión de Apple pero todavía no hay un agente disponible. David Sarno, un periodista del periódico LA Times realizó un estudio sobre la demora en el proceso de aprobación de aplicaciones para el App Store. En este estudio Sarno dice que Apple recibía aproximadamente 26,000 aplicaciones por semana en Marzo del 2012. Se desconoce el número de agentes que hacen las revisiones, pero Apple estima que una aplicación se demora entre 4 y 6 días en este estado. En el caso de “Drive 2 Survive”, se demoró 5 días.
- **In Review:** En esta etapa, Apple ha asignado un agente y la revisión está en proceso. El tiempo que se demora la revisión depende en la complejidad de la aplicación, un agente debe probar todas las funciones de la aplicación para asegurarse que se cumplan todas las reglas. En el caso de “Drive 2 Survive”, este proceso se demoró 2 días.
- **Processing for App Store:** El 18 de Abril del 2012, la revisión del juego “Drive 2 Survive” se completó y ahora se prepara la aplicación para incluirse en el App Store. Este proceso se demoró menos de una hora.
- **Ready for Sale:** Finalmente, luego de 7 días, el juego “Drive 2 Survive” se aprobó y estaba listo para la venta.

Se realizaron pruebas descargando el juego ya publicado para verificar su funcionamiento y no se encontraron problemas. Ya con el juego publicado, la campaña de marketing podía empezar.

Marketing

Se planificó una campaña de marketing para promocionar el juego, el objetivo fue aumentar el número de ventas con un presupuesto total de \$4000. De acuerdo a Jacky Yap, un escritor de la comunidad E27.com, se necesita por lo menos un presupuesto de \$30,000 para promocionar de manera exitosa una aplicación en el App Store. Jacky Yap escribió un artículo llamado “Betting on your app? You better have a \$30,000 marketing budget”¹⁴, en el que relata que ganar dinero a través de aplicaciones en el App Store no es tan fácil ya que esta tienda virtual está saturada con una gran cantidad de aplicaciones y juegos; y es por esto que es difícil destacarse entre tantos productos. El presupuesto era prácticamente el 10% de lo que Jacky Yap recomienda, y por esta razón nuestra estrategia de marketing debía ser muy creativa para poder invertir esos 4000 dólares al máximo.

La mejor oportunidad de tener una campaña exitosa era si se usaba las redes sociales. Usando las redes sociales se puede tener un alcance mucho mayor al que el presupuesto permitía. Esto es por que en las redes sociales, una campaña de marketing puede llegar a muchas personas si llama la atención. Ese era el objetivo.

Campaña en Redes Sociales

Para generar ese interés, se propuso regalar iPads. El presupuesto alcanzaba para comprar 5 iPads y sortear entre las personas que ayuden a compartir la noticia de que el juego “Drive 2 Survive” ya se puede descargar del App Store. Fue así que se llegó a definir las reglas del sorteo:

- La persona que quiera ingresar en el sorteo, debe hacer “Like” a la página de Facebook de Oubii Games, o seguir la cuenta de Twitter de Oubii Games.
- La persona debe compartir en Facebook o hacer retweet en Twitter, el mensaje promocional que Oubii Games hace para regalar cada iPad.
- La persona debe ver el video del trailer del juego “Drive 2 Survive”.
- Los ganadores se escojen entre las personas que compartieron el mensaje del iPad que se va a sortear en cualquiera de las dos redes sociales.
- El sorteo era válido sólo entre personas que puedan recibir el iPad en Estados Unidos. No se podía enviar un iPad a una dirección internacional.

Para que las reglas del sorteo sean claras de entender, se desarrolló una pequeña página web en el sitio de Oubii Games con un cuadro explicativo. La siguiente figura muestra la página web del sorteo:


Like 160 Follow @oub

[Home](#) | [About Us](#) | [Games](#) | [Blog](#) | [Say Hello!](#)

Celebrate the launch of **DRIVE TO SURVIVE** and win The New iPad!

Available on the iPhone 

We are giving away 5 new iPads in May!

See it to believe it!



How can I win an iPad 3?

Just follow the simple instructions below using Facebook **OR** Twitter

facebook.

- 1** Like us Like 160
- 2** Share & Like **our message**
- 3** Checkout the trailer of our new game **Drive To Survive** and stay tuned each day! We will announce the winner on this page!

OR

twitter

- 1** Follow Us Follow @oubii
- 2** Tweet Our Message Tweet 6,891
- 3** Checkout the trailer of our new game **Drive To Survive** and stay tuned each day! We will announce the winner on this page!

Figura 5.4 Página del Sorteo

La razón por la que decidimos regalar el iPad 3 es por que era un producto recién anunciado por Apple. Había mucho interés en el nuevo producto y se asumió que iba a generar mucha atención el regalar cinco iPads. Mientras se anunciaba un ganador cada

semana, la campaña seguiría generando interés llegando a muchas personas ya que si alguien quería ganar un iPad, tenía que compartir con sus amigos generando una posible reacción en cadena.

La proyección de esta campaña fue llegar a los 10,000 likes en la página de Facebook y por lo menos 5,000 seguidores en Twitter.

Anuncios en Internet

El sorteo estaba listo para empezar, pero necesitábamos que la gente se enterara del sorteo. Básicamente el sorteo necesitaba un impulso inicial para llegar a exponerse al público y empezar a generar interés. El resto de presupuesto alcanzó para promover el sorteo con un anuncio publicitario. La estrategia aquí fue contratar un espacio publicitario en un sitio web donde los visitantes sean el target. El target que se buscaba eran personas entre 12 y 50 años que les guste los productos Apple y que les interese tener un iPad. Fue así que se encontró el sitio web llamado CultofMac.com. Cult of Mac es una comunidad de noticias únicamente relacionadas a productos Apple, y ofrecen espacios de anuncios con tamaño 125x125 píxeles a un costo de aproximadamente \$1000 mensuales, dependiendo del número de impresiones.

Fue así que se creó un anuncio para promocionar el sorteo y se alquiló el espacio que Cult of Mac ofrecía. La siguiente figura muestra el anuncio que se usó:



Figura 5.5 Anuncio de Publicidad para el Sorteo

Con el anuncio activado, se publicó al mismo tiempo la página web del sorteo y se observaron los resultados. Se regalaron los cinco iPads en un periodo de 30 días y se observó el alcance que tenía la campaña. Los resultados de la campaña y las ventas del juego se muestran en el siguiente capítulo.

ANÁLISIS DE RESULTADOS Y CONCLUSIONES

Resultados

Se analizaron dos diferentes resultados. Por un lado el alcance logrado con la campaña de marketing y por otro la cantidad de ventas alcanzadas. Primero se presenta el alcance de la campaña de marketing, con dos diferentes indicadores:

- Número de “Likes” en la página de Facebook.
- Número de Seguidores en la cuenta de Twitter.

La tabla 6.1 y figura muestra el incremento de “Likes” en la cuenta de Facebook de Oubii Games, entre Marzo y Agosto del 2012:

Mes	Número de Likes
Marzo	0
Abril	58
Mayo	510
Junio	701
Julio	695
Agosto	683

Tabla 6.1 Número de Likes en Facebook

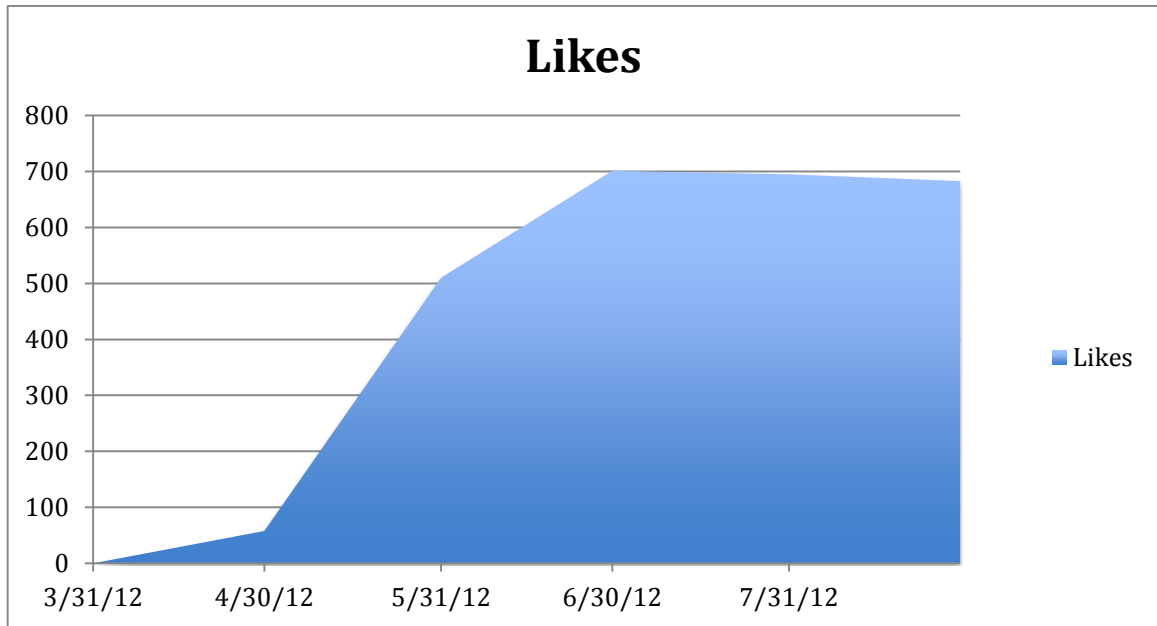


Figura 6.1 Número de Likes en Facebook

El siguiente indicador es el número de seguidores en Twitter. En la tabla 6.2 se muestra el incremento en número de seguidores dentro del mismo periodo de 6 meses:

Mes	Seguidores de Twitter
Marzo	2
Abril	131
Mayo	274
Junio	412
Julio	389
Agosto	371

Tabla 6.2 Número de seguidores en Twitter

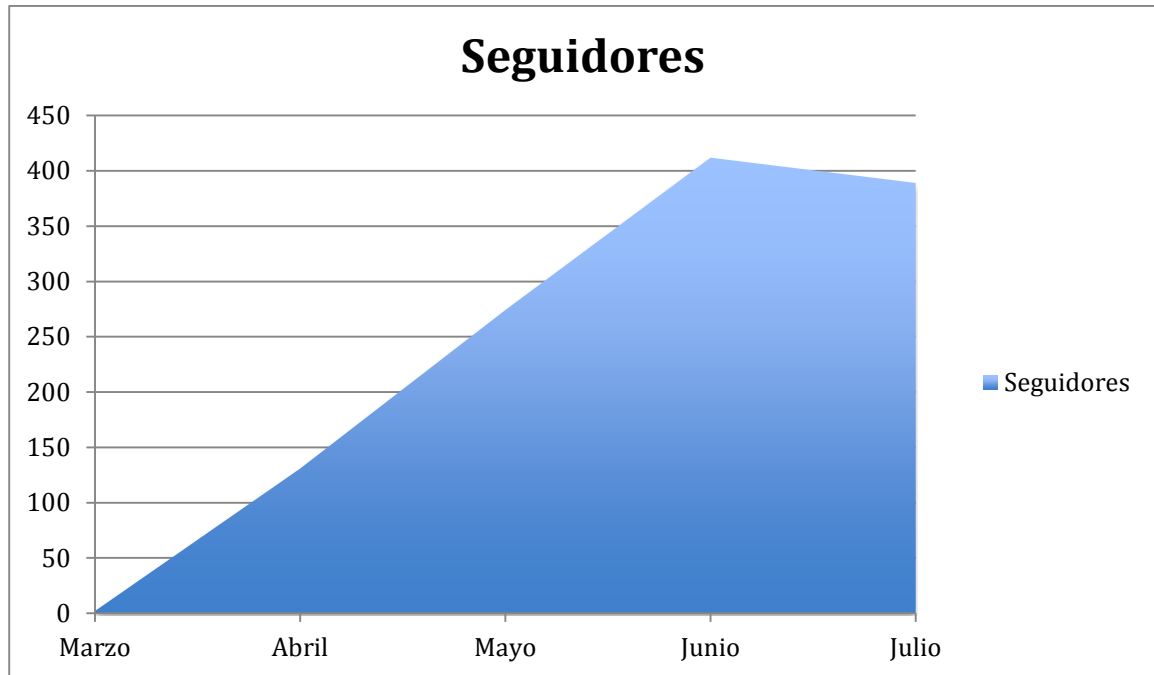


Figura 6.2 Número de Seguidores en Twitter

Los resultados presentados, indican el alcance que tuvo la campaña de marketing. Sin embargo, el objetivo final fue incrementar las ventas del juego. En la figura 6.3 se muestra el total número de descargas que se hicieron en el mismo periodo de 6 meses:

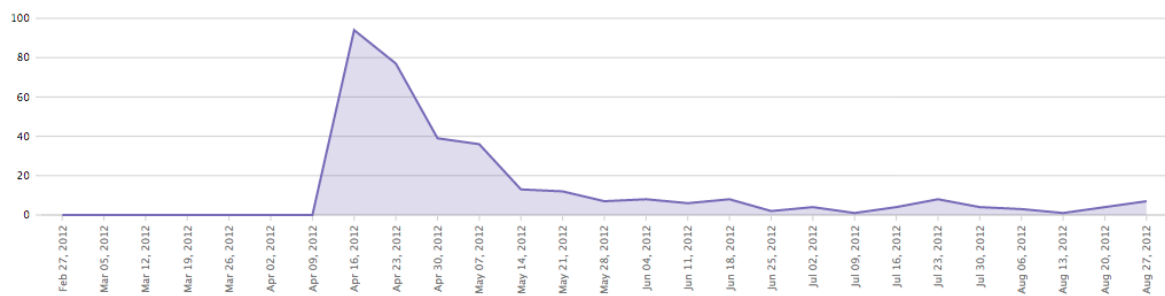


Figura 6.3 Número de Descargas

El número total de descargas dentro de este periodo de análisis fue de **338 unidades**.

El rango de precio del juego “Drive 2 Survive” es el primer nivel \$0.99. De este valor Apple retiene 30% y el 70% restante se entrega al desarrollador. En la figura 6.4 se muestra el gráfico de ventas en base al número de descargas:

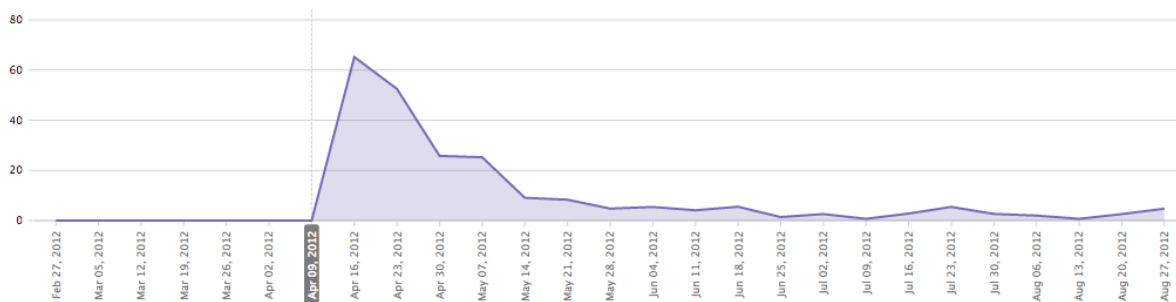


Figura 6.4 Valor en Ventas

El valor total de ventas en el periodo de análisis fue de \$232 USD. El número de ventas es evidentemente bajo. Sin embargo, es importante analizar también la experiencia de los usuarios que si compraron el juego. En la figura 6.5 se muestra las calificaciones que se otorgaron al juego en el App Store:

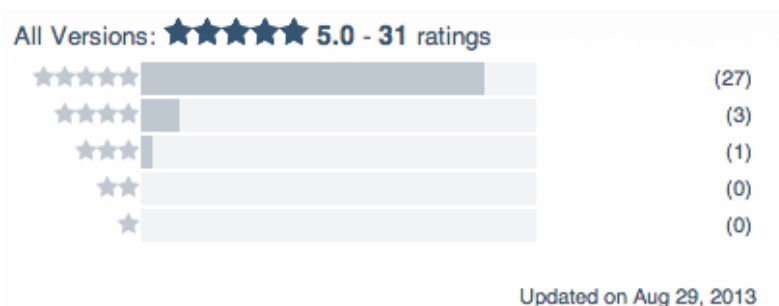


Figura 6.5 Calificaciones en el App Store

El número total de calificaciones recibidas fue de 31, de las cuales 27 son de 5 estrellas, 3 de 4 estrellas y una calificación de 3 estrellas.

El sistema de calificaciones del App Store es anónimo, pero los usuarios también pueden optar por dejar un comentario. A continuación se muestran algunos de los comentarios que se recibieron:

(Anexo B)

Adicionalmente, la comunidad de aplicaciones Appdiction publicó un artículo en su sitio web sobre “Drive 2 Survive”, con comentarios positivos sobre el juego. El artículo se lo puede leer en la dirección: <http://appdictionaries.com/drive-2-survive-fast-and-furious/>

En la siguiente sección se encuentran las conclusiones en base a los resultados presentados.

Conclusiones

Al analizar los resultados, es evidente que el alcance de la campaña de marketing no cumplió con las expectativas. No se llegaron a los 10,000 “Likes” que se esperaban en la página de Facebook de Oubii Games, sino solo 700. No se llegaron a los 5,000 seguidores en la cuenta de Twitter, sino solo 412. La falta de una campaña de marketing bien planificada afecto directamente a los resultados, solo se llegaron a venderse 338 unidades en un periodo de 6 meses desde el lanzamiento del juego en el App Store.

Si bien es cierto que existen, según Apple, mas de 500 millones de cuentas activas que compran aplicaciones en el App Store; el juego de “Drive 2 Survive” no alcanzó a vender suficiente para cubrir la inversión inicial de \$6229.

Sin embargo, los comentarios publicados en el App Store y en internet son en su gran mayoría positivos. Las calificaciones muestran que las personas que si compraron el juego, lo disfrutaron. De los 31 comentarios recibidos, solo uno es negativo. Estos hechos deben ser tomados en cuenta para llegar a una conclusión. Por un lado tenemos pocas descargas, y por otro tenemos buenos comentarios y calificaciones que muestran que el juego si fue entretenido para las personas que lo descargaron.

Es por esto que se concluye lo siguiente:

Desarrollar un juego para iPhone solo es la mitad del trabajo. Uno puede invertir mucho dinero creando un juego, pero nada de eso importa si el público no tiene conocimiento del mismo.

El enfoque principal del proyecto fue crear el juego. Se aprendió a usar diferentes plataformas como Corona SDK y herramientas los gráficos animados y así se obtuvo el producto deseado. Sin embargo, no se utilizaron las mismas prácticas al promover el juego. La campaña de marketing no tuvo fundamentos y demostró ser por lo menos igual de importante que el desarrollo del juego. Sin duda se merecía un enfoque especial para obtener mejores resultados.

El presupuesto de marketing utilizado para este proyecto fue mínimo y no tuvo ningún beneficio significativo para la venta del juego. Sin tomar en cuenta el presupuesto, la estrategia planteada en la campaña de marketing no fue la correcta. Los mismos recursos económicos se pudieron haber utilizado para otra estrategia.

En este proyecto no se hizo una consultoría de marketing, antes de organizar la campaña en internet. Con una consultoría en marketing la promoción del juego hubiese sido más exitosa.

REFERENCIAS

- Organización Agile Methodology (2015). *Agile Software Development*. Obtenido el 15 de Enero del 2015 de <http://agilemethodology.org/>
- Corona Labs (2015). *Corona Software Development*. Obtenido el 15 de Enero del 2015 de <http://www.coronalabs.com/products/corona-sdk/>
- Wiley & Sons (1996). *Lua – an extensible extension language*. Obtenido el 16 de Enero del 2015 de <http://www.lua.org/spe.html>
- PHP Organization (2015). *PHP Documentation*. Obtenido el 17 de Enero del 2015 de <http://php.net/docs.php>
- Melancon, B. (2011). *The Definitive Guide to Drupal 7*. Obtenido el 19 de Enero del 2015.
- Organización Drupal (2008). *What is Drupal?*. Obtenido el 17 de Enero del 2015 de <http://www.drupal.org>
- Kumar, K. (2012). *Drupal 7 Theming Cookbook*. Obtenido el 19 de Enero del 2015.
- Apple, Inc. (2015). *XCode Documentation*. Obtenido el 3 de Febrero del 2015 de <https://developer.apple.com/xcode>
- Zirkle, P. (2009). *iPhone Game Development*. Obtenido el 5 de Noviembre del 2014.
- Novak, J. (2011). *Game Development Essentials: An Introduction*. Obtenido el 10 de Noviembre del 2014.
- Bennet, G; Lees, B; Fisher, M (2012). *Objective-C for Absolute Beginners*. Obtenido el 13 de Noviembre del 2014.
- Learn Free Organization (2015). *What is Facebook?*. Obtenido el 9 de Enero del 2015 de <http://www.gcflearnfree.org/facebook101/2>
- Code and Web (2008). *Texture Packer Documentation*. Obtenido el 21 de Enero del 2015 de <http://www.codeandweb.com/texturepacker>
- Code and Web (2008). *Physics Editor*. Obtenido el 21 de Enero del 2015 de <http://www.codeandweb.com/physicseditor>
- Costello, S. (2013). *Number of Apps in the Apple Store*. Obtenido el 22 de Enero del 2015 de <http://ipod.about.com/od/iphonesoftwareterms/qt/apps-in-app-store.htm>
- Sarno, D. (2012). *Apple's App Store receives 26,000 submissions every week*. Obtenido el 23 de Febrero del 2015 de <http://articles.latimes.com/2012/mar/14/business/la-fi-tn-apple-26000-20120314>

- Yap, J. (2012). *Betting on your app? You better have a \$30,000 marketing budget*. Obtenido el 15 de Enero del 2015 de <https://e27.co/betting-on-your-app-you-better-have-a-30000-marketing-budget/>
- Heney, E. (2013). *App Marketing and How to Make Money with iPhone Apps*. Obtenido el 13 de Noviembre del 2014.
- Fernandez, M. (2012). *Corona SDK Mobile Game Development: Beginner's Guide*. Obtenido el 13 de Noviembre del 2014.
- Mekersa, D. (2012). *Create 2D Mobile Games with Corona SDK*. Obtenido el 13 de Noviembre del 2014.
- Lerusalimschy, R. (2013). *Programming in Lua*. Obtenido el 13 de Noviembre del 2014.
- Costa, A.; Costa, N.; Lima, L (2012). *Corona SDK - Uploading your app to the Apple Store*. Obtenido el 12 de Diciembre del 2014.
- Meyer, E. (2006). *CSS: The Definitive Guide*. Obtenido el 17 de Febrero del 2015.
- Duckett, J. (2011). *HTML and CSS: Design and Build Websites*. Obtenido el 17 de Febrero del 2015.
- Caso, C (2012). *Drive 2 Survive: Fast and Furious* – Obtenido el 15 de Enero del 2015 de <http://appdiction.com/drive-2-survive-fast-and-furious/>
- Apple, Inc. (2015). *Apple App Store Guidelines*. Obtenido el 3 de Febrero del 2015 de <https://developer.apple.com/app-store/review/guidelines/>

ANEXO A: TAREAS DEL PROYECTO.

Tarea	Sistema	Componente	Subcomponente	Descripción	Historia	Horas
Sprint 2 - Desarrollo de Juego						
D2S-001	Juego	Menus	Inicio	Crear pantalla de presentación que se muestre al iniciar el juego mostrando el logo de Oubii Games.	HJ01	2
D2S-002	Juego	Menus	Pantalla de Menú	Crear pantalla principal del juego con la navegación y el botón para empezar a jugar.	HJ02	8
D2S-003	Juego	Menus	Configuraciones	Crear pantalla de configuraciones	HJ03	4
D2S-004	Juego	Menus	Configuraciones	Crear componente de control de volumen para la música de los menus.	HJ03	4
D2S-005	Juego	Menus	Configuraciones	Crear componente de control de volumen para la música del juego.	HJ04	2
D2S-006	Juego	Menus	Configuraciones	Crear componente de control de volumen para sonidos especiales del juego.	HJ05	2
D2S-007	Juego	Menus	Configuraciones	Crear componente de control de sensibilidad para los movimientos del auto.	HJ06	2
D2S-008	Juego	Menus	Configuraciones	Crear componente de uso de acelerómetro para control del	HJ07	2

				auto.		
D2S-009	Juego	Menus	Puntajes	Integrar puntajes del juego con la comunidad "Game Center"	HJ08	8
D2S-010	Juego	Menus	Instrucciones	Crear pantalla de instrucciones para que el usuario pueda entender como funciona el juego	HJ09	4
D2S-011	Juego	Menus	Créditos	Crear pantalla de créditos donde se muestra la lista de personas que contribuyeron al proyecto.	HJ10	4
D2S-012	Juego	Menus	Créditos	Crear enlace en la pantalla de créditos para que el usuario pueda ir a la página web de Oubii Games desde el juego.	HJ11	1
D2S-013	Juego	Menus	Pantalla de Menú	Crear botón para empezar el juego.	HJ12	2
D2S-014	Juego	Menus	Pantalla de Menú	Crear pantalla de "Loading..." que el usuario pueda visualizar mientras se carga el juego.	HJ13	2
D2S-015	Juego	Gameplay	Elementos Visuales	Crear elementos visuales principales: Auto, Carretera, Paisajes.	HJ14, HJ16	8
D2S-016	Juego	Gameplay	Control de Juego	Integrar el acelerómetro con el auto y crear los movimientos de izquierda a derecha.	HJ15	8







D2S-017	Juego	Gameplay	Control de Juego	Poner en movimiento infinito a la carretera y los paisajes.	HJ14	16
D2S-018	Juego	Gameplay	Control de Juego	Crear otros autos de diferentes tipos.	HJ17	12
D2S-019	Juego	Gameplay	Motor de Física	Integrar motor de física Box2D con objetos.	HJ17	16
D2S-020	Juego	Gameplay	Motor de Física	Crear reglas de física para visualizar colisiones entre autos.	HJ17	16
D2S-021	Juego	Gameplay	Inteligencia de Juego	Crear algoritmo que crea instancias de autos y los coloca en la carretera de manera aleatoria. Mientras incrementa la dificultad del juego.	HJ19, HJ20	16
D2S-022	Juego	Gameplay	Control de Juego	Crear la opción de pausar el juego y mostrar menú.	HJ18	3
D2S-023	Juego	Gameplay	Puntaje	Crear elemento para visualizar el puntaje mientras el usuario juega, integrarlo con el avance del juego.	HJ21	6
D2S-024	Juego	Gameplay	Armas	Crear dos armas para el auto principal, una metralleta y un lanza cohetes.	HJ22	4
D2S-025	Juego	Gameplay	Armas	Crear lógica de impacto y daño entre armas y autos.	HJ22	8
D2S-026	Juego	Gameplay	Efectos	Crear animación de explosión de cada auto destruido.	HJ23	4

D2S-027	Juego	Gameplay	Paisajes	Crear lógica de cambio de paisaje mientras el jugador avanza en el juego.	HJ24	8
D2S-028	Juego	Gameplay	Paisajes	Crear movimiento de nubes mientras el auto maneja.	HJ25	2
D2S-029	Juego	Gameplay	Lógica de Juego	Crear la lógica de juego para los accidentes de tránsito, incluyendo una advertencia visual para el usuario.	HJ26	8
D2S-030	Juego	Gameplay	Lógica de Juego	Crear la lógica de juego para la persecución de la policía.	HJ27	6
D2S-031	Juego	Gameplay	Gráficos	Crear animación de luces del auto de la policía.	HJ27	2
D2S-032	Juego	Gameplay	Lógica de Juego	Crear lógica de juego para recarga de armas.	HJ28	3
D2S-033	Juego	Gameplay	Game Over	Crear lógica para determinar cuando acaba el juego.	HJ29	2
D2S-034	Juego	Gameplay	Game Over	Crear pantalla de "Game Over" que permita al usuario volver a jugar o regresar al menú principal.	HJ31	3
D2S-035	Juego	Gameplay	Game Over	Mostrar explosión del auto principal cuando se choca.	HJ30	1
D2S-036	Juego	Gameplay	Social	Integración con Facebook para publicar puntaje.	HJ32	2
D2S-037	Juego	Gameplay	Social	Integración con Twitter para publicar puntaje.	HJ33	2

D2S-038	Juego	Gameplay	Sonidos Especiales	Integración de acciones con sonidos especiales.	HJ34	8
Sprint 3 - Desarrollo de Sitio Web						
D2S-039	Web	Servidor	Configuración	Configurar servidor para hospedaje del sitio web.	HW01	4
D2S-040	Web	Servidor	Dominio	Conectar dominios con servidor.	HW01	4
D2S-041	Web	Homepage	Backend	Crear tipos de contenido y campos.	HW02	3
D2S-042	Web	Homepage	Backend	Crear vistas y panel para página principal.	HW02	4
D2S-043	Web	Homepage	Frontend	Crear estilos para página principal.	HW02	8
D2S-044	Web	Global	Logo	Colocar logo en todas las páginas.	HW03	2
D2S-045	Web	Homepage	Slogan	Incluir slogan en la página principal.	HW04	1
D2S-046	Web	Blog	Backend	Crear página de blog.	HW05	2
D2S-047	Web	Blog	Frontend	Crear estilos para página de blog.	HW05	6
D2S-048	Web	Juegos	Backend	Crear página de juegos.	HW06	2
D2S-049	Web	Juegos	Frontend	Crear estilos para página de juegos.	HW06	6
D2S-050	Web	Campaña	Backend	Crear página de campaña de marketing.	HW07	4
D2S-051	Web	Campaña	Frontend	Crear estilos para página de campaña de marketing.	HW07	8
D2S-052	Web	Social	Facebook	Integrar el sitio web con el botón de Facebook.	HW08	2
D2S-053	Web	Social	Twitter	Integrar el sitio web con el botón de	HW09	2

				Twitter.		
D2S-054	Web	About Us	Backend	Crear página de "About Us".	HW10	2
D2S-055	Web	About Us	Frontend	Crear estilos para página "About Us".	HW10	6
D2S-056	Web	Contact Us	Backend	Crear formulario para permitir a usuarios contactarse con la compañía.	HW11	4
D2S-057	Web	Contact Us	Frontend	Crear estilos para el formulario de contacto.	HW11	6
					Total	287

ANEXO B: COMENTARIOS DEL JUEGO EN EL APP STORE.

 ★★★★★ May 13, 2012 by SoCal Asian	Version 1.0.1
Fun and Addicting - This game reminds me of classic arcade games when I was a kid. It's fun and really addicting.	
 ★★★★★ May 05, 2012 by olecram123	Version 1.0.1
Excelente Juego! - Esta como para jugarlo cada vez que tengas un tiempo libre! Para no aburrirse nunca!	
 ★★★★★ May 03, 2012 by lanotti	Version 1.0.1
Addicted - I am very competitive and this game makes me want to keep playing it, really fun, and always there to entertain me. I will beat you all!!!	
 ★★★★★ Apr 28, 2012 by Mauuger	Version 1.0.1
Súper bacán - Altamente recomendado para los adictos a pa velocidad!	
 ★★★★★☆ Apr 26, 2012 by SI4891	Version 1.0.1
<p>Good start with the potential to be one of the best - Endless runners are quickly becoming ten a penny with developers all over taking their turn at making their own, some are successful, and some are not. This game is certainly off to a great start especially as its trying something a little different. As opposed to travelling from left to right as per most games in this genre you actually travel up the screen across numerous lanes of traffic in an attempt to out maneuver traffic and collisions having a rocket launcher and machine gun to assist you. The graphics and sound are great along with the controls which include using the accelerometer or buttons, both work well. There is the promise of more vehicles etc to come which is why I have given this game the benefit of the doubt and awarded 4 stars before these promises are delivered on. However for the full 5 I'd like to see missions and rewards to keep me coming back for more similar to how jetpack joyride works for example. It would also be good to see traffic travelling in the opposite direction and the ability to steer out of collisions and even cause accidents to shake off the police. I hope the developers deliver on their promise to provide more. As it stands this is still a good game but with a little tweaking it could be one of the best in the genre.</p>	
^ Collapse reviews	
 ★★★★★ Apr 26, 2012 by forrito88	Version 1.0.1
Great app! Super addicting! - Even though I have a douchebag as a brother...him and my cousin have definitely a wicked awesome app. Thumbs up	