|  |
| --- |
| **A description...**  CICLO FORMATIVO DE GRADO SUPERIOR  DESARROLLO DE APLICACIONES MULTIPLATAFORMA |
| PROYECTO FIN DE CICLO |
| **Move Your School** |
|  |
| Raúl Ordás Fernández, Andrés Pérez Gómez y Cristina Díez Sobrino |
| **CURSO 2018-19** |

**TÍTULO** : Move Your School

**AUTORES**: RAÚL ORDÁS FERNÁNDEZ

ANDRÉS PÉREZ GÓMEZ

CRISTINA DÍEZ SOBRINO

**TUTOR DEL PROYECTO**: Ernesto Ramiro Córdoba

**FECHA DE LECTURA**: ... Marzo de 2018

En Madrid

Ernesto Ramiro Córdoba

Tutor del PFC

**RESUMEN:**

Hemos observado que muchas instituciones educativas consideran tedioso encontrar diferentes actividades para sus alumnos debido a que deben buscar en distintas plataformas información sobre las excursiones. Queremos facilitar esta búsqueda con una aplicación que contenga todos los datos sobre varias actividades para grupos educativos. Mediante nuestra aplicación se podrán subir detalles de las distintas actividades que las empresas u organizaciones ofrecen para centros educativos. Gracias a esta aplicación el profesorado podrá organizar de manera sencilla las actividades más convenientes para sus alumnos.

Por tanto, nuestra aplicación está dirigida tanto como a profesores en busca de actividades para sus alumnos, como para empresas u organizaciones que ofrezcan eventos educativos, excursiones orientadas para jóvenes, y actividades infantiles. De esta manera los centros escolares podrán encontrar fácilmente actividades que les interesen y las empresas podrán ofrecer información sobre sus servicios de una manera sencilla y llegar a un público mayor.

Mediante las valoraciones de los usuarios se podrá obtener información sobre las actividades más interesantes para los docentes que podrán filtrar por distintas áreas para facilitar la búsqueda.

Nuestro objetivo principal es proporcionar sencillez a la hora de organizar actividades curriculares y extracurriculares para los colegios, institutos, academias, universidades… Y promover la comunicación entre empresas e instituciones con objetivo educacional.

Adicionalmente, creemos que es una buena herramienta para pequeñas empresas y organizaciones que pueden ofrecer sus ofertas y servicios para grupos escolares y ser recomendadas por los usuarios. Esta puede ser una forma eficaz de hacer publicidad que llegue al tipo de cliente que buscan.

**ABSTRACT:**

(es lo mismo que el resumen, pero en inglés)

**AGRADECIMIENTOS**

Nos gustaría agradecer a los padres de Cristina, ambos maestros, por ofrecernos la idea que hizo que este proyecto surgiera.

También agradecer a nuestro profesor Ernesto el seguimiento de nuestra actividad para su correcta elaboración.



Esta obra se distribuye bajo una licencia Creative Commons.

Se permite la copia, distribución, uso y comunicación de la obra si se respetan las

siguientes condiciones:

* Se debe reconocer explícitamente la autoría de la obra incluyendo esta nota y su
* enlace.
* La copia será literal y completa
* No se podrá hacer uso de los derechos permitidos con fines comerciales, salvo permiso expreso de los autores.

El texto precedente no es la licencia completa sino una nota orientativa de la licencia

original completa(jurídicamente válida) que puede encontrarse en: <http://creativecommons.org/licenses/by-nc-nd/3.0/deed.es>

INDICE

1. INTRODUCCIÓN 1

1.1. Objetivos 1

1.2. Motivación 1

1.3. Antecedentes 2

2. DESARROLLO DEL PROYECTO 3

2.1. Herramientas tecnológicas 3

2.2. Planificación 3

2.3. Descripción del trabajo realizado 3

2.4. Resultados y validación 4

3. CONCLUSIONES 5

3.1. Innovación 5

3.2. Trabajo futuro 5

4. BIBLIOGRAFÍA Y WEBGRAFÍA 7

5. ANEXOS

5.1. Presentación general de un informe I

5.2. Directrices particulares para el proyecto II

5.3. Directrices particulares para los listados de programas II

5.4. Algunas reglas mecanográficas IV

1. **Introducción**

El principal objetivo de este PFC es realizar una aplicación útil, sencilla y manejable para que los centros docentes y otras instituciones educativas puedan organizar excursiones de la manera más cómoda posible. Además debe abastecer las necesidades de búsqueda de los usuarios con filtros y otras herramientas, que ayuden a encontrar eventos acordes con la edad de los alumnos o el tipo de asignatura.

Pensamos que esta herramienta puede despertar el interés de las empresas organizadoras de eventos ya que pueden ofertar sus tarifas y servicios llegando a un gran número de gente. Además, es una herramienta que facilita el contacto con universidades, colegios, institutos, academias… que llevan grandes números de alumnos a estas actividades.

Con estos objetivos hemos desarrollado *Move Your School,* una aplicación para dispositivos Android que ofrece información sobre excursiones y eventos organizados por empresas u organizaciones. Tiene un método de búsqueda para el usuario y varios filtros para facilitar el manejo de la aplicación. Los usuarios se pueden registrar como institución educativa o empresa organizadora de eventos.

A nivel gráfico queremos que sea sencilla y muy visual para que el usuario se encuentre a gusto utilizándola y no haya ningún tipo de confusión.

Investigación previa sobre la práctica.

Describir el propósito del PFC ya en el primer párrafo. En la primera página resumir todo lo realizado. En el resto del capítulo se debería solamente describir el contexto, interés, restricciones y objetivos del PFC, sin apuntar cómo se podrían conseguir.

(La memoria no debe ser un resumen -corta/pega- de varias fuentes, si no supera el test de la herramienta de plagio se considerará, sin más, suspenso el proyecto).

* 1. Objetivos

Los objetivos de un trabajo de innovación, desarrollo o investigación forman una de las partes fundamentales en la presentación del mismo. Normalmente los objetivos suelen colocarse delante o detrás, como en este caso, de una breve introducción al trabajo realizado.

Los objetivos deben quedar claros. Antes de dar una explicación del alcance, motivación y justificación de los objetivos, resulta conveniente enumerarlos de forma clara mediante ítems o en forma de tabla, posteriormente se pueden comentar brevemente.

Ejemplo de objetivos son los siguientes:

Los objetivos de *Move Your School* son:

* Crear una aplicación Android funcional y manejable.
* Crear una base de datos Firebase que guarde los datos de la aplicación.
* Desarrollar una interfaz limpia y sencilla.
* Promover la diversidad de eventos y excursiones

A continuación se podría dar una breve explicación sobre cada uno de estos objetivos. La explicación y justificación de estos objetivos deben ser tanto más elaboradas como extenso haya sido el trabajo realizado o importantes hayan sido los objetivos. Para ello se puede incluir también el siguiente punto que puede ser interesante para trabajos de investigación y desarrollo más elaborados.

* 1. Motivación

En los trabajos de investigación sobre todo debe justificarse la razón por la que se ha emprendido el trabajo y deben justificarse los objetivos del trabajo, incluyendo una parte de motivación en las memorias presentadas. ¿Por qué elegimos este proyecto? ¿Qué elementos de él conectan con nuestros intereses?

Este proyecto surgió por nuestra inquietud cultural y por la detección de un problema a la hora de encontrar actividades en los centros educativos.

La idea surgió después de que Cristina, una integrante del proyecto, escuchara a sus padres (ambos maestros) hablando sobre las dificultades que les suponía indagar en internet y otras fuentes para obtener información de actividades adecuadas para sus alumnos.

Nos pareció una idea brillante para nuestro PFC porque cubre nuestros conocimientos y podría incluso ser utilizada por nuestro centro de educación, la Universidad Europea.

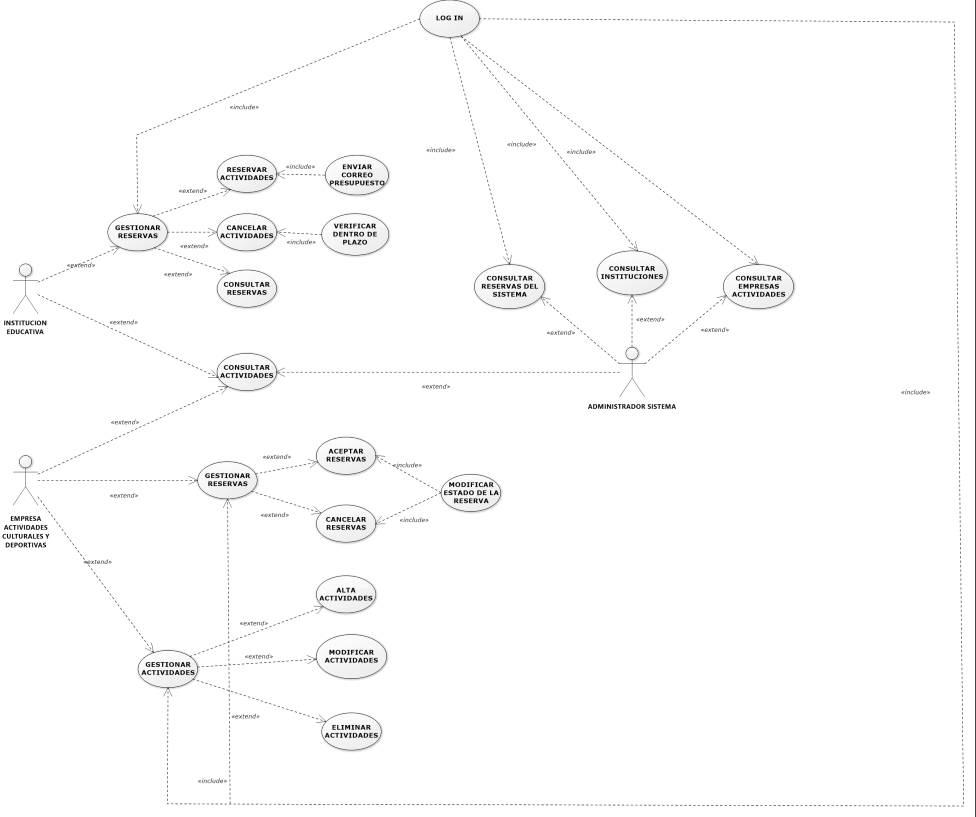
* 1. Antecedentes

Un trabajo de investigación y desarrollo tiene unos antecedentes que es necesario conocer. A esta parte se le llama también “estado del arte” o mejor, estado de la investigación. El término “estado del arte” parece tener su origen en el término inglés “state of the art” donde art no se refiere a bellas artes sino a una determinada destreza, en nuestro caso destreza científica y técnica.

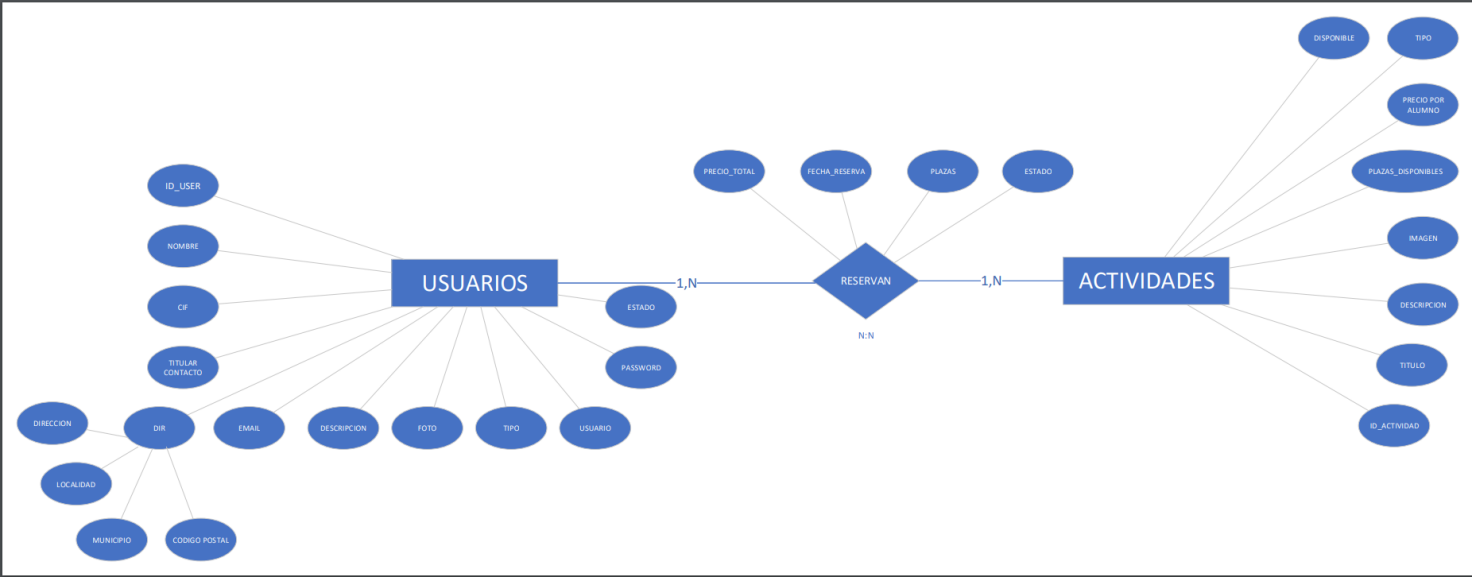
Los antecedentes son especialmente importantes en un trabajo de investigación, ya que permiten conocer los desarrollos que tienen algo que ver con lo que se está presentando. Sirve por tanto para presentar un mapa con las tecnologías y conocimientos relacionados con el trabajo presentado y permite situar nuestro trabajo en este mapa del conocimiento.

Es el lugar donde se explica el contexto del trabajo desarrollado, su relación con otras iniciativas, aplicaciones existentes, etc.

Antes de comenzar con la realización del proyecto, realizamos un diagrama de caso de uso para plantear los servicios de la aplicación e identificar los tipos de usuarios que manejarán la aplicación:



Realizamos un esquema conceptual de como debemos montar la base de datos:



En este esquema se pueden observar las relaciones y cardinalidad existentes entre las tres entidades que forman parte de la persistencia y los atributos de los usuarios, de las reservas y de las actividades.

Principales competidores:



*Eventbrite* es una aplicación que sirve para organizar eventos y vender las entradas. También sirve de buscador de eventos.

Lo que diferencia a *Eventbrite* de *Move Your School* es que no está orientada para grupos ni para colegios o instituciones educativas.



Es similar a la app anterior, [All Events in City](http://allevents.in/welcome.php) te muestra eventos en ciudades concretas.

No proporciona tampoco datos para actividades colegiales y está más centrado en la fiesta y la ciudad.

1. **DESARROLLO DEL PROYECTO**

Trata sobre la realización del trabajo en sí. En esta parte se describe lo que se ha hecho, cómo se ha llevado a cabo, por qué se ha hecho así y no de otra manera, qué materiales o herramientas ha sido necesario utilizar, qué metodología de trabajo y validación se ha utilizado, etc.

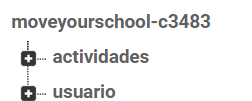
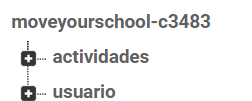
Una parte esencial del proyecto es la base de datos, la cuál hemos desarrollado con tecnología *Firebase* de *Google*.



Se ha escogido un tipo de base de datos no relacional, que se basa en una estructura de nodos donde hay pares de clave y valor.

Un ejemplo de nuestra aplicación:

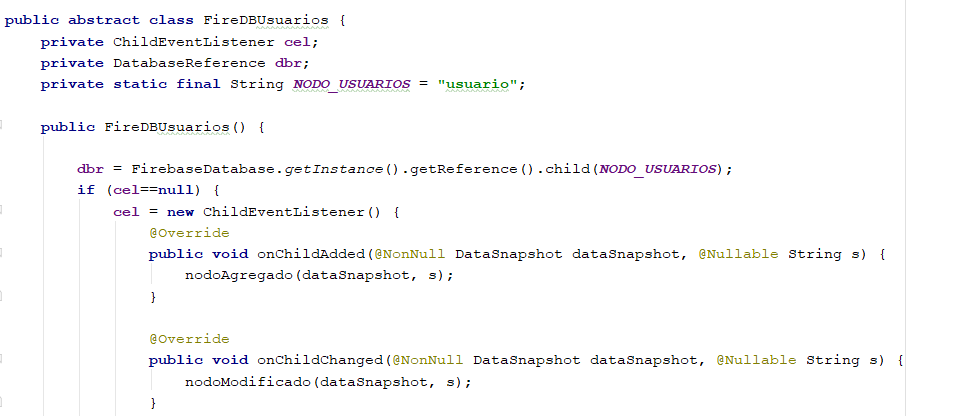
Podemos observar que se divide en actividades, usuarios... etc.



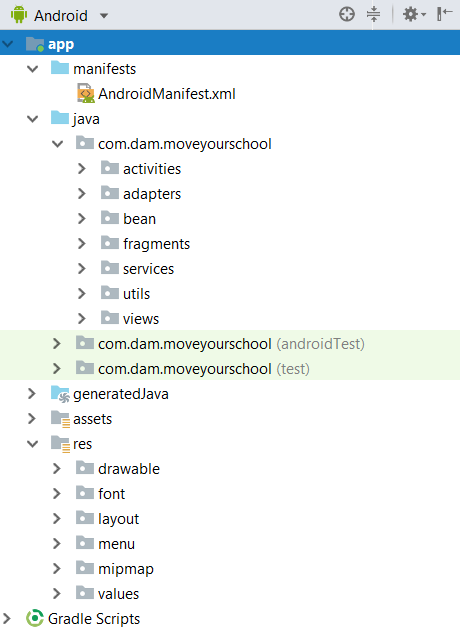
Las actividades tienen una clave y esta corresponde a la de un usuario, y el valor es un objeto actividad.



Respecto al Backend, se corresponde a una arquitectura de la aplicación MVC, se ha escogido una arquitectura con servicios desacoplados, utilizando clases abstractas para intentar separar la vista de la lógica de negocio.



La arquitectura de los paquetes se divide en funcionalidades:



* 1. Herramientas tecnológicas

A continuación ofrecemos un resumen informativo sobre las tecnologías que utilizamos para nuestro proyecto:

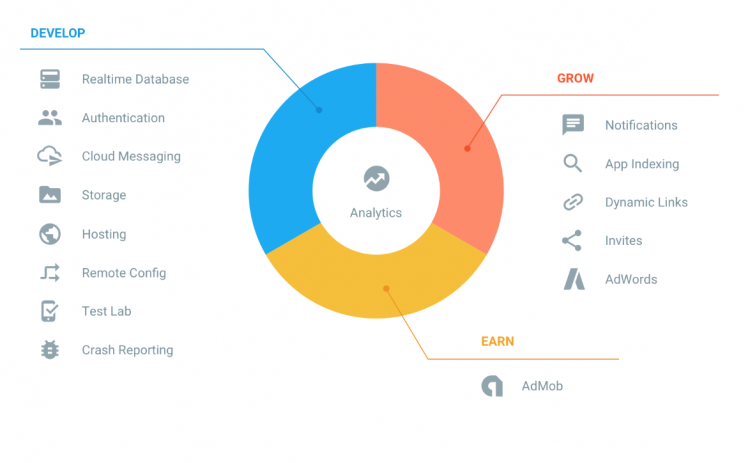
**Firebase:**

La base de datos de *Move Your School* está desarrollada en *Firebase*.

[](https://firebase.google.com/?hl=es-419)

Ctrl + Click para acceder a sitio web

Firebase es la plataforma de desarrollo móvil en la nube de Google. Fue comprado por Google en 2014 y luego la continuó mejorando con la compra del equipo de Divshot.

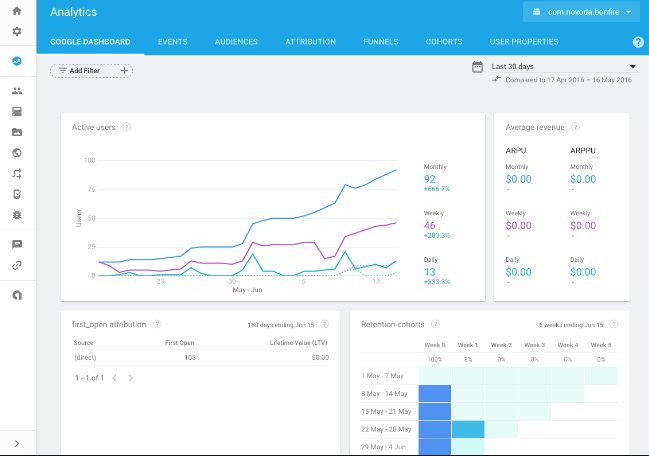


Firebase surgió para proveer una API para guardar y sincronizar datos en la nube en tiempo real.

Firebase se trata de una plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de elevada calidad de una forma rápida. La plataforma está subida en la nube y está disponible para diferente plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades.

Principales características de *Firebase*:

* **Desarrollo**: *Firebase* permite la creación de mejores apps, minimizando el tiempo de optimización y desarrollo, mediante diferentes funciones, entre las que destacan la detección de errores y de testeo, que supone poder dar un salto de calidad a la app. Poder almacenar todo en la nube, testear la app o poder configurarla de manera remota, son características destacables de la plataforma.
* **Analitica**: Tener un control máximo del rendimiento de la app mediante métricas analíticas, todo desde un único panel y de forma gratuita, es una de las ventajas que ofrece *Firebase* respecto a la analítica web.



* **Poder de crecimiento**: Permite gestionar de manera fácil todos los usuarios de las aplicaciones, con el añadido de que se pueden captar nuevos usuarios, mediante invitaciones o notificaciones.
* **Monetización**: Mediante *AdMob*, *Firebase* permite que puedas ganar dinero.
* **Rapidez**: Implementar *Firebase* puede ser fácil y rápido, gracias a su API que es muy intuitiva, sostenida en un solo *SDK*. Con *Firebase* puedes centrar tus esfuerzos en resolver los problemas de tus clientes y así poder evitar la pérdida de tiempo en la creación de una infraestructura compleja.
* **Agilidad**: *Firebase* ofrece apps multiplataforma con una *APIs* integradas a *SDK* individuales para *iOS*, *Android* y *Javascript*, de tal forma que se puede gestionar diferentes apps sin necesidad de salir de la propia plataforma.

**Android Studio**

*Android Studio* es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en*[IntelliJ IDEA.](https://www.jetbrains.com/idea/)*

[](https://developer.android.com/studio/intro/?hl=ES)

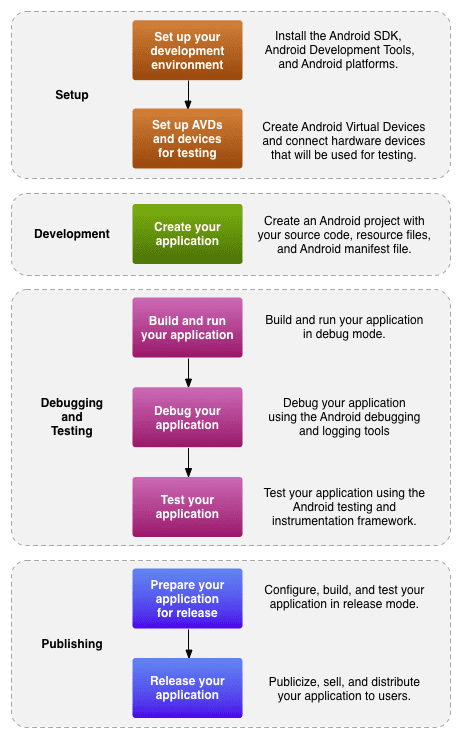
Ctrl + Click para acceder a sitio web

*Android Studio* ofrece muchas funciones que aumentan la productividad durante la compilación de apps para Android, como las siguientes:

* Un sistema de compilación basado en Gradle flexible
* Un emulador rápido con varias funciones
* Un entorno unificado en el que puedes realizar desarrollos para todos los dispositivos *Android*
* *Instant Run* para aplicar cambios mientras tu app se ejecuta sin la necesidad de compilar un nuevo *APK*
* Integración de plantillas de código y *GitHub* para ayudarte a compilar funciones comunes de las apps e importar ejemplos de código
* Gran cantidad de herramientas y frameworks de prueba
* Herramientas *Lint* para detectar problemas de rendimiento, usabilidad, compatibilidad de versión, etc.
* Compatibilidad con *C++* y *NDK*
* Soporte incorporado para [*Google Cloud Platform*](http://developers.google.com/cloud/devtools/android_studio_templates/?hl=ES), lo que facilita la integración de *Google Cloud Messaging* y *App Engine*

Entre las fases de desarrollo que abarcan la realización de aplicaciones en Android Studio encontramos cuatro etapas.

1. La primera es la **configuración de entorno**; durante esta fase se instala y configura el entorno de desarrollo. Además se realiza la conexión a los elementos en donde se pueden realizar la instalación de las app, y se crean dispositivos virtuales Android (AVDS).
2. La segunda fase abarca la **Configuración del Proyecto y Desarrollo**; durante esta se realiza la configuración del proyecto y el desarrollo del mismo. Hablamos de la creación de módulos que contengan recursos para la aplicación y archivos de código fuente.
3. La tercera fase comprende las**pruebas, depuración y construcción de la aplicación**; A esta altura se construye el proyecto en un paquete (s) depurable .apk que se puede instalar y ejecutar en el emulador o en un dispositivo con Android.
4. Ya como última fase se haya la **publicación de la aplicación**; en esta etapa se realiza la configuración y se arma la solicitud para el uso y libre distribución de la aplicación a los usuarios. Durante la etapa de preparación se construye una versión de la aplicación, que los usuarios pueden descargar e instalar en sus dispositivos de modo que se pueda vender y distribuir la versión de esta.

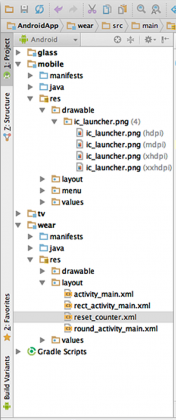


*En esta imagen podemos apreciar el diagrama de las etapas para la realización de aplicaciones en Android Studio.*

En el caso de cada proyecto, en referencia con la base modular,  la aplicación contiene uno o más módulos con archivos de código fuente y archivos de recursos.

De forma predeterminada, Android Studio muestra los archivos del proyecto en la vista del proyecto Android. En este punto se aprecia de forma organizada los módulos para proporcionar un acceso rápido a los archivos de código fuente clave.

En Android Studio se utiliza Gradle como la base del sistema de construcción de aplicaciones. Este sistema de creación, se ejecuta como una herramienta integrada en el menú Android Studio, y a su vez es independiente de la línea de comandos.

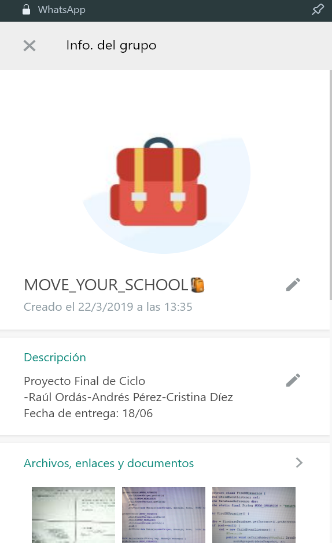


*Organización de los módulos en Android Studio*

En resumen Android Studio es  **entorno dedicado en exclusiva a la programación de aplicaciones para dispositivos Android**, proporcionando a Google un mayor control sobre el proceso de producción.

* 1. Planificación

Cómo se organizará el equipo: canales de comunicación (Slack, GitHub, Waffle, Mail, Whatsapp, etc.) tareas de cada uno, que se espera conseguir, temporalización.



**Grupo de whatssap:**

Mediante Whatssap enviamos apuntes sencillos, archivos,

Fotos y vídeos explicatorios para los demás integrantes

del grupo.

Será la vía de comunicación más directa pero

con menos carácter formal.

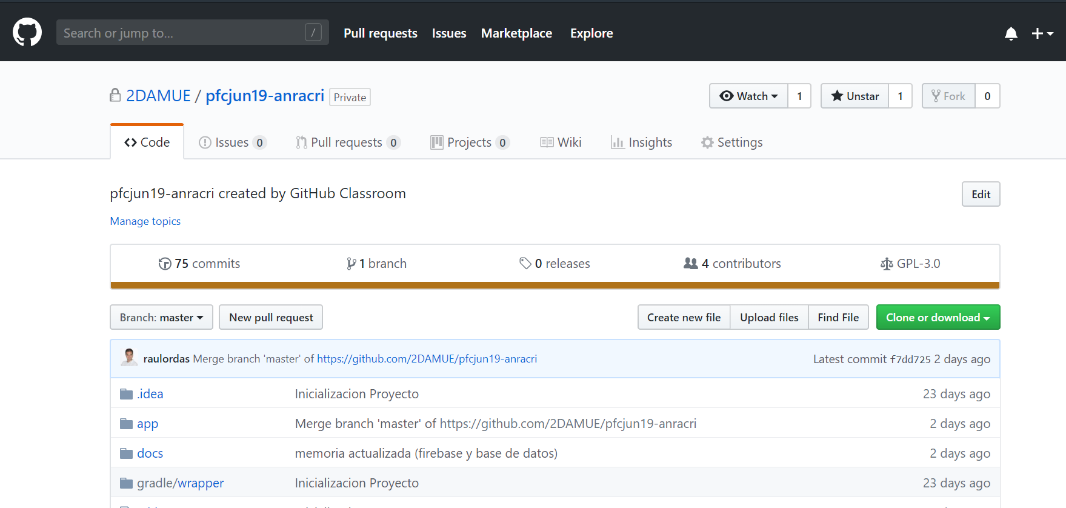
**Slack:**

Slack es la plataforma mediante la que nos comunicamos con

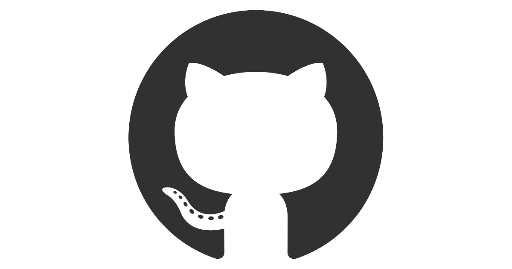
nuestro tutor, Ernesto, para detallar los avances del proyecto

y ultimar las decisiones del desarrollo.



**GitHub**

La plataforma mediante la cual compartimos el código es GitHub



* 1. Descripción del trabajo realizado

Se deben describir aquí los pasos que se han realizado para llevar a cabo los objetivos del trabajo. Estos pasos deben incluir también todos aquellos cálculos y diseños previos que ha sido necesario realizar, preparación del material y herramientas..

En este apartado se incluirá –en el caso de un desarrollo de aplicación- un estudio de los borradores o **mockups** con los que se inició el diseño de la misma.

Es importante incluir **impresiones de pantalla** para facilitar la explicación del proceso realizado, además de utilizar **tablas**, **gráficos** etc cuando sea necesario.



**Figura 1: Vista de la herramienta Slack**

Importante que todas las imágenes tengan un pie como el de la Figura 1.

En memorias sobre desarrollos es importante incluir algunos aspectos interesantes del código (clases, métodos, librerías, ...) un ejemplo de cómo debería verse (siempre un detalle en Anexo el resto):

|  |
| --- |
| public class HelloWorld {  public static void main(String[] args) {  System.out.println("Hello, World");  }  } |

**Listado 1: Hello World Java**

* 1. Resultados y validación

En esta última parte se deben presentar los resultados obtenidos del trabajo. Si el trabajo consistía en hacer un programa se comentará su funcionamiento, lo que tarda en ejecutarse, lo bien que va, los problemas que tiene.

Sobre todo también hay que presentar los resultados de las simulaciones o ejecuciones que se hayan realizado. Si se ha hecho una red es normal probar que funciona, en este caso se comentarán las condiciones en las que se ha probado, la normativa que se ha utilizado, por qué creemos que funciona bien. Esto mismo es aplicable a la instalación de un servicio de red o diseño de una base de datos.

Normalmente el diseño de la red, la configuración del servicio o la documentación de la base de datos se adjuntan en un apéndice al final, salvo que sea realmente breve y se desee comentar aquí mismo. Ver los apéndices al final de esta guía para ver cómo presentar correctamente los listados.

Cuando los resultados contengan variables numéricas o estadísticas es necesario hacer una análisis de los errores de los valores obtenidos. Este análisis debe incluir tanto la exactitud como la precisión de las medidas obtenidas.

En esta parte de resultados debe incluirse un estudio detallado de los resultados obtenidos junto con sus implicaciones. Un resumen de este estudio se debe incluir en la parte final de conclusiones.

1. **CONCLUSIONES**

Son las conclusiones propiamente dichas del trabajo realizado. Es uno de los capítulos  **importantes.**

Comenzar con un resumen de lo realizado destacando los aspectos más importantes, principales hitos conseguidos, principales problemas encontrados, etc. Se comenta si se han conseguido los objetivos planteados al inicio, si se recomienda el método utilizado o cualquier otro, si el resultado es fiable o no, si son necesarias nuevas pruebas; es decir, se hace un resumen breve de los principales puntos del trabajo realizado y los resultados obtenidos. Este es un buen lugar para destacar la razón de retrasos, o cualquier otro incidente que haya retrasado la realización del PFC.

Así mismo es un buen lugar para evaluar el desvío de la implementación final con respecto al mokcup del diseño original.

* 1. Innovación

En este apartado se pueden destacar aquellos aspectos novedosos que aporta la realización del proyecto. Resulta interesante evaluar el impacto que puede tener cada una de las aportaciones realizadas.

En las aportaciones resulta interesante también comentar la relación de los resultados obtenidos con los trabajos previos que existan sobre el tema.

* 1. Trabajo futuro

En los trabajos de investigación extensos con objetivos ambiciosos conviene enumerar aquellos puntos del trabajo sobre los que se debe hacer énfasis. También se deben destacar aquellas líneas que el trabajo abre y que pueden dar resultados interesantes.

4. BIBLIOGRAFÍA Y WEBGRAFÍA

Al final del informe es necesario añadir la bibliografía general con las referencias que se hayan hecho a lo largo del texto. Un trabajo de investigación y desarrollo sin referencias es un trabajo muy poco documentado y demuestra una falta del estudio de los antecedentes y entorno del trabajo, por lo tanto tiene poca credibilidad como tal.

La bibliografía debe seguir el formato A.P.A. como los siguientes:

* EDUCAMADRID. (2011) www.educa2.madrid.org. Fecha de consulta: 22:17, febrero 14, 2011 de http://www.educa2.madrid.org/educamadrid
* CLOUTIER, J. (1975). L’ére D’emerec ou la comunication audio-scrito-visuelle à L’ heure des self-media. (segunda ed.). Montréal: Les Press de L’ Université de Montréal.
* CALLEJO GALLEGO, J. (2008). El esquema espaciotemporal en la sociedad digital. Madrid. Madrid: UNED.
* O’REILLY, T. (2004, 05) de Octubre. Conferences.oreillynet.com. Fecha de consulta: 09:09, febrero 15, 2011 de http://conferences.oreillynet.com/web2con
* KAPLÚN, M. (1998). Una pedagogía de la comunicación. Madrid: Ediciones de la Torre.
* FLATICON (2013) Consulta de todos los iconos de la aplicación. <https://www.flaticon.com/>
* Manuel Pérez Cardona. (14 OCT 2016). Firebase, qué es y para qué sirve la plataforma de Google. 7 MAY 2019, de IEBS Sitio web: <https://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarroladores-google-seo-sem/>
* Google Developers.(2019) Conoce Android Studio. 08 MAY 2019, de User guide Android Sitio web: <https://developer.android.com/studio/intro/?hl=ES>
* pedrini210. (2016). Carácteristicas y cualidades de Android Studio. 08 MAY 2019, de DESDE LINUX Sitio web: <https://blog.desdelinux.net/caracteristicas-y-cualidades-de-android-studio/>

5. ANEXOS

En los anexos se recoge información técnica subsidiaria al trabajo realizado y que no tiene cabida directa en el esquema presentado hasta ahora. Es el lugar ideal para los listados de código, esquemas de circuitos, demostración de teoremas, etc.

Numerar con letras mayúsculas (pueden tener (sub)secciones: A.1, A.3.1, etc.). Incluir en el anexo todo aquello que ocupe muchas páginas y tenga una estructura repetitiva que pueda hacer tediosa la lectura (resumir en el texto principal y referenciar el anexo). El Manual de Usuario puede incluirse como un anexo o como un documento aparte (comenzando en la página 1 o no). Debería ser una guía de uso donde se explica cómo utilizar cada funcionalidad que ofrezca el sistema, explicando los pasos a dar y apoyándonos masivamente en pantallazos.

Consideraciones importantes para la elaboración de la MEMORIA

Lo más importante de un trabajo es sin duda el trabajo en sí y sus resultados, sin embargo, un buen trabajo mal presentado arrojará dudas sobre lo allí expuesto y sobre la/s persona/s que han realizado el trabajo y la memoria. Por lo tanto resulta conveniente cuidar la presentación del mismo:

* Hay que evitar radicalmente las faltas de ortografía y gramaticales. Así como cuidar la formalidad del texto (no escribir como se habla y no escribir en primera persona “He considerado...” mejor “Se considera...” o “Hay o habría que considerar...”). Un trabajo presentado con faltas de ortografía, aunque sea bueno, dará una impresión lamentable.
* Es mejor utilizar un tipo de letra estándar y un tamaño también estándar. El tamaño de letra debe estar entre 10 y 12 pt., y la fuente puede ser Verdana, Arial o similar. Por lo general la letra Verdana de 12 pt. suele ser la más utilizada. Nunca hay que usar tipos raros o poco legibles y nunca tamaños inferiores a 10 pt. (10 pt. ya es realmente pequeño) ni superiores a 12 pt.
* Los listados de los programas se hacen con una fuente que tenga espaciado fijo (Courier por ejemplo). Estas fuentes suelen ser más grandes que las normales por lo que puede ser interesante cambiar también el tamaño de letra y dejarlo entre 8 y 10 pt.
* El interlineado debe ser simple y nunca menor de una línea. En algunos casos se puede agrandar algo pero sin sobrepasar la mitad del interlineado simple (1.5 líneas).
* En todos los trabajos resulta interesante incluir un índice con los capítulos, secciones y subsecciones que contenga, así como la página en la que se encuentra.
* La inclusión de imágenes ilustrativas y figuras -sin abusar- ayudan a la claridad y mejoran la presentación de un trabajo.

Directrices particulares para el proyecto

* La presentación es muy importante precisamente porque hace brillar su contenido cuando es valioso.
* En la portada añadir el título del proyecto y los integrantes.
* No dejar hojas completas en blanco como separación, más allá de las que puedan aparecer en este documento.
* Abrir una nueva página al comenzar una nueva sección, salvo para evitar que un título quede suelto. Hacer un salto de página para empezar una nueva sección.
* La extensión del proyecto debe ser **entre 40 y 60 páginas** incluyendo todo el trabajo realizado, las conclusiones, etc., sin extenderse en detalles superfluos.

Directrices particulares para los listados de programas

En los proyectos es bastante común la realización de memorias sobre la creación de algún programa, da igual el lenguaje, que haga alguna cosa. En estos casos las directrices dadas en los puntos anteriores son de obligada aplicación, aunque resulta conveniente resaltar los siguientes aspectos:

* El listado del programa debe estar comentado siempre. Si un programa no tiene comentarios casi es mejor no incluirlo.
* Un listado del programa, por muy bien comentado que esté, no es por sí solo la memoria del proyecto, de hecho debería ser una de las partes menos importantes.
* Si se incluye el listado completo del programa se pondrá en los anexos al final de la memoria. Sólo en el caso de que el programa sea muy pequeño (apenas una rutina) se puede poner en la parte de desarrollo o resultados.
* Si el listado es muy grande (más de 3 ó 4 páginas) es mejor no incluirlo o ponerlo aparte en un cd. También se puede intentar reducir el tamaño de letra, los espaciados, etc.
* En la parte de desarrollo o resultados se describe el funcionamiento del programa, por qué se ha hecho así, se explicará si funciona bien o no, bajo qué casos, etc. En estos apartados se pueden incluir trozos del listado del programa, pero sólo aquellas partes significativas para la explicación. Si el trozo que se incluye ocupa una página o más entonces no es aconsejable ponerlo.
* Los listados, tanto si son fragmentos como si es el listado global, se deben hacer con un tipo de letra de tamaño fijo (por ejemplo Consolas). Por otro lado, el tamaño debe ser lo menor posible para que ocupe poco espacio.
* Es recomendables insertar los listados de código en una tabla e insertarse un título (Referencias -> Insertar Título), si hay muchos listados puede ser interesante añadir un Índice de listados.

Algunas reglas mecanográficas

Estas reglas son casi tan importantes como las ortográficas y su incumplimiento produce una mala impresión del trabajo que se está realizando. Se listan a continuación algunas de las más importantes o utilizadas:

* Los signos de puntuación como los puntos, comas, dos puntos, etc., van siempre unidos a la palabra que preceda; es decir, nunca hay que poner un espacio delante de cualquiera de estos signos.
* En cuanto a los signos que se abren y cierran como las comillas, los paréntesis, las llaves, etc., el signo que abre debe ir unido a la palabra que sigue, y el signo que cierra debe ir unido a la palabra precedente; es decir, detrás de un signo que abre nunca va un espacio al igual que delante de un símbolo de cierre donde tampoco se pone espacio.
* Los títulos de las secciones o subsecciones no deben ir sueltos. Si no hay espacio en la página para empezar una sección se debe empezar en página nueva.