



**UNIVERSIDAD DE CASTILLA-LA MANCHA**  
**ESCUELA SUPERIOR DE INFORMÁTICA**

**GRADO EN INGENIERÍA INFORMÁTICA**

**TECNOLOGÍA ESPECÍFICA DE**  
**TECNOLOGÍAS DE LA INFORMACIÓN**

**TRABAJO FIN DE GRADO**

**Plantilla para Memoria de**  
**Trabajo Fin de Grado**

**Diego Andérica Richard**

Mayo, 2017



PLANTILLA PARA MEMORIA DE  
TRABAJO FIN DE GRADO





**UNIVERSIDAD DE CASTILLA-LA MANCHA**

**ESCUELA SUPERIOR DE INFORMÁTICA**

**Departamento de Tecnologías y Sistemas de Información**

**TECNOLOGÍA ESPECÍFICA DE  
TECNOLOGÍAS DE LA INFORMACIÓN**

**TRABAJO FIN DE GRADO**

**Plantilla para Memoria de  
Trabajo Fin de Grado**

Autor: Diego Andérica Richard

Director: Luis Rodríguez Benítez

Director: Luis Jiménez Linares

Mayo, 2017



**Diego Andérica Richard**

Ciudad Real – Spain

*E-mail:* Diego.Anderica@alu.uclm.es

© 2017 Diego Andérica Richard

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Se permite la copia, distribución y/o modificación de este documento bajo los términos de la Licencia de Documentación Libre GNU, versión 1.3 o cualquier versión posterior publicada por la *Free Software Foundation*; sin secciones invariantes. Una copia de esta licencia esta incluida en el apéndice titulado «GNU Free Documentation License».

Muchos de los nombres usados por las compañías para diferenciar sus productos y servicios son reclamados como marcas registradas. Allí donde estos nombres aparezcan en este documento, y cuando el autor haya sido informado de esas marcas registradas, los nombres estarán escritos en mayúsculas o como nombres propios.





**TRIBUNAL:**

**Presidente:**

**Vocal:**

**Secretario:**

**FECHA DE DEFENSA:**

**CALIFICACIÓN:**

**PRESIDENTE**

**VOCAL**

**SECRETARIO**

Fdo.:

Fdo.:

Fdo.:



# Resumen

El presente documento es un ejemplo de memoria del Trabajo de Fin Grado según el formato y criterios de la Escuela Superior de Informática de Ciudad Real. La intención es que este texto sirva además como una serie de consejos sobre tipografía,  $\text{\LaTeX}$ , redacción y estructura de la memoria que podrían resultar de ayuda. Por este motivo, se aconseja al lector consultar también el código fuente de este documento.

Este documento utiliza la clase  $\text{\LaTeX}$  *esi-tfg*, disponible como paquete Debian/Ubuntu, consulta:

<https://bitbucket.org/arco.group/esi-tfg>.

Si encuentra cualquier error o tiene alguna sugerencia, por favor, utilice el *issue tracker* del proyecto *esi-tfg* en:

<https://bitbucket.org/arco.group/esi-tfg/issues>

El resumen debería estar formado por dos o tres párrafos resaltando lo más destacable del documento. No es una introducción al problema, es decir, debería incluir los logros más importantes del proyecto. Suele ser más sencillo escribirlo cuando la memoria está prácticamente terminada. Debería caber en esta página (es decir, esta cara).



# Abstract

English version of the previous page.



# Agradecimientos

Escribe aquí algunos chascarrillos simpáticos. Haz buen uso de todos tus recursos literarios porque probablemente será la única página que lean tus amigos y familiares. Debería caber en esta página (esta cara de la hoja).

Diego<sup>1</sup>

---

<sup>1</sup>Sí, los agradecimientos se firman





*A alguien querido y/o respetado*



# Índice general

<b>Resumen</b>	<b>V</b>
<b>Abstract</b>	<b>VII</b>
<b>Agradecimientos</b>	<b>IX</b>
<b>Índice general</b>	<b>XIII</b>
<b>Índice de tablas</b>	<b>XVII</b>
<b>Índice de figuras</b>	<b>XIX</b>
<b>Índice de listados</b>	<b>XXI</b>
<b>Listado de acrónimos</b>	<b>XXIII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Estructura de la memoria . . . . .	4
<b>2. Objetivos</b>	<b>5</b>
2.1. Objetivo general . . . . .	5
2.2. Objetivos específicos . . . . .	5
2.2.1. Objetivo I: Implementar un marco de gestión de usuarios vinculado al contexto educativo . . . . .	6
2.2.2. Objetivo II: Proporcionar un entorno de ejecución multiplataforma .	6
2.2.3. Objetivo III: Implementar un mecanismo de monitorización activa del tipo y contenido de los mensajes . . . . .	6
2.2.4. Objetivo IV: Integración de la aplicación con Google Calendar . . .	6
2.2.5. Objetivo V: Implementar mecanismos que permitan comunicaciones personales profesor-tutores del alumno y viceversa . . . . .	7
<b>3. Antecedentes</b>	<b>9</b>
3.1. WhatsApp . . . . .	9

## 0. ÍNDICE GENERAL

3.1.1.	Funcionalidades a nivel de Usuario . . . . .	10
3.1.2.	Funcionalidades para los Desarrolladores . . . . .	10
3.2.	Telegram . . . . .	11
3.2.1.	Funcionalidades a nivel de Usuario . . . . .	11
3.2.2.	Funcionalidades para los Desarrolladores . . . . .	12
3.3.	Slack . . . . .	12
3.3.1.	Funcionalidades para los Usuarios . . . . .	13
3.3.2.	Funcionalidades para los Desarrolladores . . . . .	13
3.4.	Skype . . . . .	14
3.4.1.	Funcionalidades para los Usuarios . . . . .	14
3.4.2.	Funcionalidades para los Desarrolladores . . . . .	15
3.5.	Signal . . . . .	15
3.5.1.	Funcionalidades para los Usuarios . . . . .	16
3.5.2.	Funcionalidades para los Desarrolladores . . . . .	16
3.6.	Wickr . . . . .	16
3.6.1.	Funcionalidades para los Usuarios . . . . .	17
3.6.2.	Funcionalidades para los Desarrolladores . . . . .	17
3.6.3.	Funcionalidades para los Desarrolladores . . . . .	18
3.7.	Comparación de Alternativas . . . . .	18
3.8.	Aplicaciones Especializadas . . . . .	19
3.8.1.	miColegioApp . . . . .	19
3.8.2.	BabyNotez . . . . .	19
3.8.3.	TokApp School . . . . .	20
<b>4.</b>	<b>Metodología</b>	<b>21</b>
4.1.	Metodología de gestión de proyectos: Scrum . . . . .	21
4.1.1.	Teoría de Scrum . . . . .	21
4.1.2.	El equipo de Scrum . . . . .	22
4.1.3.	Eventos de Scrum . . . . .	22
4.1.4.	Artefactos de Scrum . . . . .	24
4.2.	Metodología de Desarrollo de Software: Iterativo e Incremental . . . . .	25
4.3.	Recursos Hardware y Software . . . . .	25
4.3.1.	Recursos Hardware . . . . .	26
4.3.2.	Recursos Software . . . . .	26

<b>5. Resultados</b>	<b>29</b>
5.1. Planificación inicial . . . . .	29
5.1.1. Pila de Producto . . . . .	29
5.2. Sprint 1: Diseño de una plataforma Web para la gestión de los usuarios . . .	29
5.2.1. Planificación del Sprint . . . . .	30
5.2.2. Resultados del Sprint . . . . .	30
5.3. Ejemplo Historia de Usuario . . . . .	32
<b>Un anexo de ejemplo</b>	<b>35</b>
<b>Referencias</b>	<b>37</b>



## Índice de tablas

3.1. WhatsApp vs Telegram . . . . .	18
5.1. Historia de Usuario 1 . . . . .	30
5.2. Historia de Usuario 2 . . . . .	32





# Índice de figuras

1.1. Logo de Papás 2.0 . . . . .	1
1.2. Captura conversación madres . . . . .	3
3.1. Logo de WhatsApp . . . . .	9
3.2. Logo de Telegram . . . . .	11
3.3. Logo de Slack . . . . .	12
3.4. Logo de Skype . . . . .	14
3.5. <i>Tweet</i> de Snowden sobre Signal . . . . .	15
3.6. Logo de Signal . . . . .	15
3.7. Logo de Wickr . . . . .	17
3.8. Logo de miColegioApp . . . . .	19
3.9. Logo de BabyNotez . . . . .	19
3.10. Logo de TokApp School . . . . .	20
4.1. Burn-down chart . . . . .	25
4.2. Información de Windows . . . . .	26
4.3. Tablero Kanban en GitHub . . . . .	27



## Índice de listados



## Listado de acrónimos

<b>MB</b>	MegaByte
<b>API</b>	Application Programming Interface
<b>IP</b>	Internet Protocol
<b>VoIP</b>	Voice over IP
<b>GB</b>	GigaByte
<b>P2P</b>	Peer to Peer
<b>SMS</b>	Short Message Service
<b>NSA</b>	National Security Agency
<b>JCCM</b>	Junta de Comunidades de Castilla-La Mancha
<b>GHz</b>	Gigahercios
<b>CSV</b>	Comma Separated Values



## Capítulo 1

# Introducción

**A**CTUALMENTE existen numerosos avances en lo que a las «nuevas tecnologías» se refiere y cada vez tenemos más dispositivos con los que comunicarnos y estar en contacto con amigos, familiares o conocidos en todo momento. Estas nuevas formas de comunicación están presentes en todos los ámbitos y han modificado la forma de relacionarse. Uno de los ámbitos en los que ha sucedido esto es el educativo. Por ejemplo, antes se recurría al reparto de una circular para anunciar ciertos eventos a los alumnos para que se la entregaran a los padres o tutores o escribir una nota en la agenda del alumno para que la entregase firmada al día siguiente si no había hecho los deberes asignados o había ocurrido algún percance. Posteriormente, se podría usar el correo electrónico para comunicarse directamente con los padres, asegurándose de la correcta recepción del comunicado, aunque pudiera ocurrir que éstos no lo revisaran en tiempo y forma.

Existen multitud de aplicaciones «generalistas» de mensajería instantánea que se describirán con detalle a lo largo de este trabajo. La comunidad autónoma de Castilla-La Mancha posee la plataforma educativa llamada «Papás 2.0» (Figura 1.1), que se encuentra más enfocada al sector educativo. Se trata de una plataforma educativa perteneciente a la Consejería de Educación, Cultura y Deportes de la Junta de Comunidades de Castilla-La Mancha (JCCM) que facilita la gestión administrativa y establece una vía de comunicación entre los centros educativos y las familias, ofreciendo información relevante a los padres [JCC17]. Además, permite llevar un seguimiento sobre las tareas, trabajos, controles, exámenes, faltas de asistencia y fechas de entrega [JCC10].



Figura 1.1: Logo de Papás 2.0

## 1. INTRODUCCIÓN

Papás 2.0 ofrece una serie de características que pueden resultar de gran valor para los padres, siendo éstas las más destacadas:

- Visualizar los profesores que dan clase a los hijos con sus datos y posibilidad de escribir un mensaje directamente a cualquiera de ellos.
- Consultar las citas concertadas con los profesores, junto con la fecha, hora y motivo de la visita.
- Consultar el horario escolar.
- Consultar las faltas de asistencia, con la posibilidad de ser notificado vía SMS o correo electrónico. Del mismo modo, se podrá registrar una notificación cuando se sepa que el hijo va a faltar a ciertas horas.
- Consultar trabajos y tareas de cada hijo, así como ver las fechas de los exámenes y sus notas del curso y su trayectoria escolar.
- Envío y recepción de mensajes mediante grupos, pudiendo adjuntar archivos de tamaño no superior a 1 MegaByte (MB) y con un máximo de 3 MB en total.

A pesar de que esta aplicación es de gran utilidad por las funcionalidades y facilidades que ofrece tanto a padres como a personal docente, se trata de una plataforma más cercana al correo electrónico que a una aplicación de mensajería instantánea, donde las personas sí que se comunican en tiempo real. En este último caso es donde toman especial relevancia los *smartphones*, que se suelen llevar consigo y desde los que se puede acceder a casi cualquier servicio por estar permanentemente conectados a Internet. Por tanto, normalmente no se espera a tener un ordenador cerca para comunicarse con otras personas, sino que se hace uso directamente del teléfono móvil y de sus aplicaciones. Teniendo esto en cuenta, los profesores podrían comunicarse en tiempo real con los padres y éstos pueden aportar algún tipo de *feedback* en un corto periodo de tiempo.

Aunque existen diversas soluciones para la comunicación entre los padres y el centro en el que se encuentren sus hijos y aportan numerosos beneficios a la comunidad educativa, muchas veces se pueden producir malentendidos cuando se hace uso de la mensajería instantánea, entre otros problemas produciéndose, además, situaciones poco deseables entre todos los miembros. Esto se debe a que se crean grupos de chat en los que el centro tiene poco o nada que ver, no se encuentran convenientemente controlados y, con el tiempo, puede ocurrir que la finalidad original se acabe desvirtuando.



Un ejemplo de esa desvirtuación es este caso, donde un grupo de madres se negaba a llevar a sus hijos al colegio debido a que a la clase de éstos acudía un niño que sufría síndrome de Asperger. Sucedió en Buenos Aires, Argentina, y pronto se dio a conocer en Internet. En las capturas de la conversación (Figura 1.2) se puede ver cómo las madres se alegran de que este chico fuera cambiado de clase. «Una buenísima noticia», según indicaba una de las integrantes [Van17].



Figura 1.2: Captura conversación madres

Más allá de los problemas que puedan surgir entre los padres, los grupos pueden llegar incluso a dañar a los propios hijos. Esto puede suceder puesto que hay padres que comparten el trabajo realizado en casa para que otros niños o padres puedan beneficiarse de ello. Esta práctica puede repercutir en un mal aprendizaje del niño y, por tanto, en una disminución de su rendimiento escolar. Incluso pueden llegar a compartir fotos de los regalos colocados debajo del árbol en la época navideña [Ali17]. Por todo esto se debe establecer de manera firme y consensuada la figura del administrador del grupo, que será quien se encargue del cumplimiento y gestión de las normas para que la relación y saber estar de los padres no se quede sólo en el trato presencial sino que se extrapole a las nuevas soluciones digitales.

Un último caso es el de una madre que mandó una carta a una redactora. Sucedió en Argentina y en ella explica cómo, a raíz de abandonar el grupo de madres, éstas la tratan de una manera diferente debido a que en dicho grupo se hablaba demasiado, llegando a ser «insoportables», decía la madre. Lo que pensaba se confirmó al necesitar a una madre para que recogiera a su hija, asegurando que le devolvería el favor la próxima vez. Esta madre leyó e ignoró el mensaje y, posteriormente, alegó que no se había dado cuenta [Con17].

## 1. INTRODUCCIÓN

Las bondades de la mensajería instantánea son muy amplias y bien conocidas aunque, como en todo, se debe tener medida en su uso y cierto control, sobre todo si se trata de temas más sensibles como la educación o los hijos. Por eso, en este trabajo fin de grado se plantea el desarrollo de una aplicación enfocada a este sector que evite, en la medida de lo posible, problemas como los descritos anteriormente.

### 1.1 Estructura de la memoria

A continuación se detalla la estructura de este trabajo fin de grado, que se encuentra dividido en varios capítulos:

#### **Capítulo 1: Introducción**

En este primer capítulo se realiza una breve introducción al ámbito en el que se desarrolla este trabajo, junto con la presentación de algunas aplicaciones que se usan en la actualidad, así como algunos conflictos que pueden derivar del mal uso de éstas.

#### **Capítulo 2: Objetivos**

Se exponen tanto el objetivo principal del trabajo como los objetivos específicos que se deberán cumplir para lograr la consecución del mismo.

#### **Capítulo 3: Antecedentes**

Este capítulo está destinado al estudio de las soluciones que se pueden encontrar en el mercado en cuanto a mensajería instantánea, realizando una comparación final de las características de las dos aplicaciones más utilizadas.

#### **Capítulo 4: Metodología**

En este capítulo se detallan las metodologías seguidas a lo largo del desarrollo. Se incluyen la metodología de gestión de proyectos y la metodología de desarrollo de software.

#### **Capítulo 5: Resultados**

Por último, en este capítulo final se reflejan los resultados obtenidos como producto de seguir una planificación teniendo en cuenta los objetivos expuestos en el capítulo anterior.

## Capítulo 2

# Objetivos

**E**N este capítulo se expondrán tanto el objetivo principal como los diferentes objetivos específicos del Trabajo Fin de Grado, así como las posibles limitaciones o condicionantes que se pudieran derivar de los mismos.

### 2.1 Objetivo general

El principal objetivo de este trabajo es implementar una herramienta de mensajería instantánea para la comunicación entre el profesorado y los padres de los alumnos específica para el contexto educativo. Se apoyará en un *backend* proporcionado por la empresa Google, llamado *Firebase*. Para su desarrollo se utilizará el Entorno de Desarrollo Integrado *Android Studio*. La motivación por la que surge esta idea reside en la necesidad de disponer de una aplicación adaptada y acotada a las necesidades primordiales en la gestión de la comunicación con los padres por parte del centro educativo. Actualmente, la mayor parte de las alternativas que existen son más generalistas, orientándose a cualquier tipo de público. Por tanto, éstas incluyen funciones cuya utilidad en el sector profesional de la docencia sería, en determinados casos, cuestionable. Adicionalmente, en la actualidad, los grupos de mensajería pueden generar diversos problemas de convivencia. Por todo lo anteriormente mencionado, se considera de interés realizar una aplicación que trate de minimizar la ocurrencia de situaciones no deseadas.

### 2.2 Objetivos específicos

En esta sección se detallarán los objetivos específicos a completar para poder así cumplir el objetivo general.

## 2. OBJETIVOS

### 2.2.1 Objetivo I: Implementar un marco de gestión de usuarios vinculado al contexto educativo

Se pretende implementar un mecanismo de gestión de usuarios que sea lo más eficiente y sencillo de utilizar. Por ejemplo, existe la posibilidad de que el personal del centro realice importación de archivos .csv para la creación de los grupos de chat. De igual manera, se deberán elegir y fijar diferentes roles, que serán asignados a las personas que utilicen la aplicación: el rol de moderador que estará destinado, principalmente, a los tutores, profesores o personal del centro que use la aplicación y que será el encargado de validar finalmente el envío de mensajes y el rol de usuario normal, destinado a los padres que se encuentren registrados.

### 2.2.2 Objetivo II: Proporcionar un entorno de ejecución multiplataforma

Puesto que el personal del centro tendrá las funciones de importar datos, registrar a los usuarios, crear y mantener los grupos de chat, éstas se realizarán mediante el uso de un ordenador personal puesto que, de esta manera, dichas tareas se tornan más sencillas de realizar y de un *frontend* que resulte amigable y fácil de utilizar por parte del profesorado y personal del centro. Por otra parte, el resto de los usuarios de la aplicación accederán a la misma mediante el teléfono móvil o correo electrónico, puesto que los usuarios podrán elegir el método de entrada.

### 2.2.3 Objetivo III: Implementar un mecanismo de monitorización activa del tipo y contenido de los mensajes

Se busca disponer de algún tipo de monitorización para detectar mensajes que puedan ser potencialmente improcedentes dentro del contexto educativo. Esto se puede conseguir mediante el uso de la plataforma *BlueMix* de IBM, desde la que se podrá acceder a «Watson», un sistema de inteligencia artificial con el que se pueden controlar los mensajes que son enviados a través de la aplicación. Una vez enviado el texto a analizar, éste debe ser interpretado por la aplicación, puesto que es devuelto en un formato diferente. Además, se dispondrá de cierta moderación puesto que el profesor responsable de cada grupo tendrá la capacidad de validar los mensajes que el resto de usuarios envíen a dicho grupo.

### 2.2.4 Objetivo IV: Integración de la aplicación con Google Calendar

Este objetivo se centrará en el estudio de cómo compenetrar la aplicación con otros servicios, como *Google Calendar*, de manera que se puedan agregar nuevos eventos de calendario sin que el usuario tenga que cambiar de aplicación manualmente. Ejemplos de estos eventos podrían ser añadir nuevos exámenes, reuniones, tutorías con los profesores, etc.

**2.2.5 Objetivo V: Implementar mecanismos que permitan comunicaciones personales profesor-tutores del alumno y viceversa**

Además de la comunicación mediante los grupos creados por el personal del centro, los usuarios de la aplicación podrán iniciar una conversación con los profesores mediante la creación de un chat privado. Del mismo modo, los profesores podrán crear chats privados con el resto de usuarios.



## Capítulo 3

# Antecedentes

**H**OY en día existen numerosas aplicaciones para comunicarse de manera instantánea con otros usuarios, permitiendo la coordinación y organización de un grupo de trabajo o, simplemente, para hablar con los amigos. Esto último es lo que se conoce habitualmente como *chatear*. En este capítulo se presentará una serie de herramientas para la mensajería instantánea junto con las características más destacadas que ofrecen a los usuarios finales junto con las que ofrecen a los desarrolladores, por ejemplo, si poseen una Application Programming Interface (API), si son multiplataforma o si se tratan de aplicaciones de código abierto.

### 3.1 WhatsApp

WhatsApp (figura 3.1) es una de las aplicaciones de mensajería instantánea más usadas a nivel de usuario, quizá porque fue de las primeras en llegar a los *smartphones* o teléfonos inteligentes. En sus comienzos, la aplicación venía a suplir la carencia que acusaba Jan Koum, uno de los creadores, de poder ver un estado al lado de cada uno de los contactos de la agenda de, por aquel entonces, un recién lanzado iPhone. De esta manera, un usuario podría conocer qué estaba haciendo un contacto mediante un texto corto o «estado». Finalmente, el 24 de febrero de 2009, WhatsApp se creó como empresa y producto y a principios de 2011 ya se encontraba en el top 20 de aplicaciones en la *App Store* de *Apple* en Estados Unidos.



Figura 3.1: Logo de WhatsApp

### 3. ANTECEDENTES

El 19 de febrero de 2014, la empresa de Mark Zuckerberg, *Facebook*, anunció la compra de WhatsApp por 19.000 millones de dólares, aunque no sería hasta octubre del mismo año cuando se llevara a cabo la compra definitiva por 21.800 millones de dólares [Nov14].

#### 3.1.1 Funcionalidades a nivel de Usuario

WhatsApp, como se ha mencionado anteriormente, permite la comunicación en tiempo real con otros usuarios de la aplicación. Además de eso, algunas de sus características principales a nivel de usuario son [Wha17]:

- Grupos de «chat» de hasta 256 personas para compartir mensajes, fotos y vídeos con la posibilidad de darles un nombre, silenciarlos, personalizar notificaciones, etc. Cada uno de estos grupos puede tener uno o varios administradores.
- Llamadas y videollamadas mediante Voice over IP (VoIP).
- Posibilidad de continuar una conversación en un navegador web u ordenador personal mediante *WhatsApp Web* o la aplicación de escritorio. En este caso, el teléfono ha de estar conectado a Internet, puesto que hace de «puente» reenviando los mensajes al ordenador.
- Cifrado de extremo a extremo mediante el protocolo «Signal».
- Posibilidad de enviar archivos de hasta 100 MB.
- Envío de mensajes de voz.

#### 3.1.2 Funcionalidades para los Desarrolladores

Desgraciadamente, WhatsApp no cuenta con una API abierta que pueda ser utilizada por los desarrolladores para crear otras aplicaciones basadas en la original. WhatsApp es multi-plataforma, encontrándose en diferentes sistemas operativos (además de poder usarse desde un navegador web): Android, iOS, Windows Phone, Nokia Symbian S40, BlackBerry, Windows y MacOS. El número de usuarios que pueden participar en un único grupo, por el momento, es de 256. Además, cada cuenta de WhatsApp ha de estar vinculada obligatoriamente a un número de teléfono móvil.



## 3.2 Telegram

Telegram (figura 3.2) llegó con varios años de retraso con respecto a su principal rival, WhatsApp, que se encontraba ya en una posición, en cierto modo, cómoda y consolidada. Se lanzó en el año 2013, aunque no sería traducido al español y llevado a los ordenadores hasta pasado un año, en 2014. Esta es una de las principales consecuencias de que Telegram no acapare una cuota de usuarios tan grande, pues actualmente tiene cien millones de usuarios frente a los mil millones que tiene WhatsApp [Ram17]. No obstante, esta aplicación ofrece muchas características que WhatsApp no tiene, al menos, por ahora.



Figura 3.2: Logo de Telegram

### 3.2.1 Funcionalidades a nivel de Usuario

Telegram posee un rango de características algo más amplio, siendo éstas las más destacadas [Tel17b]:

- Coordinar grupos de «chat» de hasta 10.000 miembros («supergrupos»).
- Existen también los «canales», donde los usuarios pueden unirse e interactuar con él mediante sólo lectura, no pudiendo enviar ningún tipo de mensaje o archivo.
- Los historiales y conversaciones se almacenan en un servidor externo, no dependiendo del teléfono móvil.
- Posibilidad de enviar archivos de hasta 1,5 GB.
- Cifrado de mensajes mediante el protocolo propietario de Telegram: *MTPProto*.
- Borrado de mensajes no sólo en el dispositivo de origen, sino también en el de destino dentro de una conversación.
- Capacidad de hacer las veces de «nube personal» ilimitada, al poder enviarse a un mismo usuario los archivos o mensajes que quiera conservar.
- Llamadas mediante VoIP.
- Existencia de «chats» secretos. Esta modalidad consiste en que se usa un cifrado de extremo a extremo, a diferencia del resto de «chats», que usan cifrado cliente-servidor/servidor-cliente [Tel17a]. Además, estas conversaciones no se guardan en los servidores de Telegram y se puede establecer un contador para el borrado de cada mensaje.

#### 3.2.2 Funcionalidades para los Desarrolladores

- Se trata de un software *Open Source*. Esto quiere decir que el código del cliente se encuentra disponible, aunque no pasa lo mismo con el código del servidor.
- Existen dos tipos de API: una para desarrollar aplicaciones basadas en Telegram y otra para realizar *Bots*, que son como asistentes a los que se pueden realizar consultas.
- Es un servicio multiplataforma, disponible en los siguientes sistemas operativos (además de tener una versión web): Android, iOS, Firefox OS, Windows, MacOS y Linux.

#### 3.3 Slack

Slack (figura 3.3) es una aplicación destinada a los conjuntos de personas que trabajan sobre un mismo tema, es decir, principalmente, a equipos y grupos de trabajo. Ha introducido un concepto algo diferente en el ámbito de la mensajería instantánea, mejorando el conocido correo electrónico, puesto que se pueden mantener conversaciones privadas, crear canales públicos o compartir archivos. Originalmente se trataba de una herramienta interna que se comenzó a usar para el desarrollo de un juego en *flash*, llamado *Glitch*. Más tarde, Stewart Butterfield, su fundador, confesó que fracasaron al realizar el juego. A finales de 2012, Butterfield comunicó en *Twitter* que no continuarían con el juego, aunque la empresa, *Tiny Speck*, seguiría «viva» [Tho15]. Slack se lanzó en febrero de 2014 con alrededor de 15.000 usuarios, para finales de ese año eran 285.000 y la cifra ha ido subiendo hasta los más de tres millones que posee actualmente [PyM17].



Figura 3.3: Logo de Slack

Esta aplicación integra de manera eficiente otros servicios como Google Drive, GitHub, DropBox o Google Hangouts, entre otros. La comunicación se realiza mediante «canales» a los que los usuarios pueden entrar para hablar con otros, compartir archivos y documentos o enlaces.

Posee varias formas de tarificación:

- **Free**. Para equipos pequeños y de uso por tiempo ilimitado. Como restricciones tiene que sólo se pueden buscar mensajes entre los 10.000 últimos, integración con un máximo de 10 aplicaciones de terceros o videollamadas de uno a uno. **Precio:** gratuito.
- **Standard**. Ofrece más ventajas como integración ilimitada con aplicaciones de terceros, videoconferencias de hasta 15 personas o 10 GB por miembro de equipo. **Precio:** 6.25€ por usuario, al mes, si se contrata anualmente.

- **Plus.** Ofrece todas las características disponibles de Slack, así como una disponibilidad del servicio muy alta o 20 GB. **Precio:** 11.75€ por usuario, al mes, si se contrata anualmente.

### 3.3.1 Funcionalidades para los Usuarios

Las características de Slack están más enfocadas al uso corporativo, siendo estas las más destacadas [Sla17]:

- Creación de canales para conversar entre personas del mismo grupo de trabajo, siendo un concepto similar al de «grupo» en WhatsApp o Telegram, pudiendo ser canales públicos o privados. Los canales públicos son canales para proyectos, grupos y temas visibles para toda la organización. Los mensajes en estos canales se archivan y se pueden recuperar mediante búsquedas. En cuanto a los canales privados, están destinados a temas más sensibles, sólo se puede entrar a estos canales mediante invitación y los mensajes únicamente son visibles para los miembros.
- Llamadas y videollamadas integradas con posibilidad de compartir la pantalla.
- Mensajes directos.
- Posibilidad de crear «cuentas de invitado», para dar acceso a la misma información.
- Posibilidad de enviar archivos a un canal, a los que los demás usuarios pueden hacer comentarios y hacer búsquedas avanzadas para encontrarlos.
- Recomendación de canales.
- Integración con otros servicios.

### 3.3.2 Funcionalidades para los Desarrolladores

Al igual que sucede con Telegram, Slack también cuenta con una API. No obstante, se trata de una aplicación que no es *Open Source*. Slack puede instalarse en diversas plataformas y sistemas operativos: Android, iOS, Windows Phone, Windows, MacOS y Linux.

## 3.4 Skype

Skype (figura 3.4) es una de las aplicaciones de mensajería, llamadas y videollamadas más conocidas en el entorno del ordenador personal. El 29 de agosto de 2003 se lanzaba la primera beta, tratándose de un cliente Peer to Peer (P2P) gratuito cuyo código fuente no era abierto. Más tarde, en 2005, fue adquirida por eBay por 2.600 millones de dólares [Vel13].

Posteriormente, en mayo de 2011, se anuncia la compra de Skype por parte de Microsoft por la cifra de 8.500 millones de dólares y, en noviembre de 2012, sería esta aplicación la que sustituiría al conocido *Messenger* [Ram13].



Figura 3.4: Logo de Skype

### 3.4.1 Funcionalidades para los Usuarios

Skype tiene unas características más enfocadas a las llamadas, como [Sky17a]:

- Llamadas y videollamadas gratuitas, individuales y grupales, entre usuarios de Skype.
- Llamadas a teléfonos fijos y móviles con cierto coste.
- Los contactos de cierto usuario llaman a un teléfono, recibiendo el destinatario la llamada en Skype. Esto es lo que se conoce como «Número de Skype».
- Desvío de llamadas a cualquier teléfono.
- *Skype To Go*: posibilidad de llamar a números internacionales desde cualquier teléfono con un coste añadido.
- Posibilidad de usar un «chat» de mensajería instantánea, individual y grupal, donde, además de enviar texto, se pueden enviar archivos de cualquier tamaño, Short Message Service (SMS) o mensajes de voz.
- Capacidad para poder compartir pantalla de manera individual y grupal.
- *Skype Translator*: traducción de llamadas, videollamadas y mensajes instantáneos en tiempo real.

### 3.4.2 Funcionalidades para los Desarrolladores

En la página web de Skype únicamente se ofrece una limitada cantidad de posibilidades [Sky17b]:

- Creación de *bots*, con los que se puede interactuar mediante «chat», voz o vídeo.
- Posibilidad de integrar videollamadas y «chat» en una página web.
- Pagos integrados.
- Integrar aplicaciones, como *YouTube* o *Giphy*, para mandar vídeos e imágenes en movimiento sin tener que abandonar la aplicación principal (*Add-ins*).

## 3.5 Signal

Signal (figura 3.6) es la aplicación de mensajería conocida por el llamado «Caso Snowden», en el que Edward Snowden filtró los casos de espionaje de la National Security Agency (NSA). El propio Snowden fue el que recomendó esta aplicación mediante un *tweet* (ver figura 3.5) en su cuenta de *Twitter* debido a la gran seguridad y privacidad que ofrecía su método de encriptación, cifrando de punto a punto las conversaciones. Más tarde, WhatsApp adoptaría este método de cifrado, incluyéndolo por defecto en todas las conversaciones de su aplicación.



Figura 3.5: *Tweet* de Snowden sobre Signal

A diferencia de WhatsApp, Signal no guarda ningún «metadato» en los servidores, como el tiempo de conversación, quién habla a quién o cuándo lo hace.



Figura 3.6: Logo de Signal

#### 3.5.1 Funcionalidades para los Usuarios

Signal posee unas características algo más reducidas que las anteriores alternativas, pero esto se debe a que está enfocada a la seguridad y privacidad, como se ha descrito anteriormente [Sig17].

- Conversaciones individuales y grupales, en las que se puede enviar texto, mensajes de voz, vídeo, documentos e imágenes.
- Llamadas de voz y de vídeo.
- Encriptación de mensajes punto a punto.
- Posibilidad de programar un contador para hacer desaparecer los mensajes enviados.

#### 3.5.2 Funcionalidades para los Desarrolladores

Al contrario que sucede con WhatsApp, Signal es de código abierto, por lo que el código se encuentra disponible en su página de *GitHub* para que pueda ser revisado por la comunidad. Las características más destacadas son:

- Ofrece el protocolo de la API.
- Se encuentra en los siguientes sistemas operativos: Android, iOS mediante aplicación nativa y Windows, MacOS, Linux y ChromeOS mediante una aplicación para el navegador web *Google Chrome*.

### 3.6 Wickr

Wickr fue lanzada en junio de 2012 originalmente sólo para iOS, el sistema operativo de los dispositivos de *Apple*. No obstante, posteriormente fueron apareciendo más versiones. Últimamente ha tenido una mayor repercusión debido a que aparece en una serie de televisión: *Mr. Robot* [Eli16]. Al igual que Signal, su principal premisa es la seguridad y privacidad de las conversaciones de los usuarios, siendo su diseño bastante austero. No es necesario registrar el número de teléfono o una dirección de correo electrónico para comunicarse, al menos, en *Wickr Me*. Además, no se guarda «metadato» alguno. Posee diferentes modalidades de tarificación diferenciándose principalmente en el peso máximo de los archivos, el tiempo de expiración de los mensajes y controles administrativos [Wic17]:

- **Wickr Me. Precio:** gratuito.
  - Tiempo máximo de desaparición de mensajes de seis días.
  - Tamaño máximo de los archivos de 10 MB.
  - No dispone de *Secure Rooms* (equipos y proyectos de hasta 50 usuarios), controles administrativos o «chat» de voz y vídeo.

- **Wickr Plus. Precio:** 15\$ por usuario al mes, unos 13€.
  - Tiempo máximo de desaparición de mensajes de seis días.
  - Tamaño máximo de los archivos de 1 GB.
  - Dispone de algunos controles administrativos.
- **Wickr Enterprise. Precio:** se debe contactar con el departamento de ventas.
  - Tiempo máximo de desaparición de mensajes de un año.
  - Tamaño máximo de los archivos de 5GB.
  - Dispone de todos los controles administrativos.
- **Wickr Pro. Precio:** 25\$ por usuario al mes, unos 21€.
  - Tiempo máximo de desaparición de mensajes de un año.
  - Tamaño máximo de los archivos de 5GB.
  - Dispone de algunos controles administrativos (equilibrio entre *Wickr Plus* y *Wickr Enterprise*).



Figura 3.7: Logo de Wickr

### 3.6.1 Funcionalidades para los Usuarios

Las principales características que Wickr ofrece a sus usuarios son:

- Encriptación punto a punto.
- Autodestrucción de mensajes.
- Posibilidad de colaboración dentro de un equipo de trabajo.
- Envío de archivos de hasta 5 GB, tal y como se ha detallado en las modalidades de tarificación.

### 3.6.2 Funcionalidades para los Desarrolladores

Desafortunadamente, Wickr no tiene disponible más que una implementación en C del protocolo de envío de mensajes en su página de *GitHub*. Wickr se encuentra disponible en los siguientes sistemas operativos: Android, iOS, Windows, MacOS y Ubuntu.

### 3.6.3 Funcionalidades para los Desarrolladores

Papás 2.0 no ofrece su código, API o algún otro recurso que pueda ser utilizado por los desarrolladores.

## 3.7 Comparación de Alternativas

Finalmente, en la tabla 3.1, se van a comparar las dos alternativas más populares mostrándose sus funcionalidades más destacadas.

	WhatsApp	Telegram
«Chats» grupales	Hasta 256 personas	Hasta 10.000 personas
Llamadas integradas	Sí	Sí
Videollamadas integradas	Sí	No
Posee aplicación de escritorio	Sí (requiere de teléfono móvil)	Sí (conversaciones en la nube)
Cifrado	Signal (de extremo a extremo por defecto en todas las conversaciones)	De extremo a extremo en chats secretos, servidor-cliente en el resto de conversaciones
Envío de archivos multimedia	Sí	Sí
Envío de documentos y otros archivos	Sí, con límite de 100 MB	Sí, hasta 1,5 GB y sin extensión específica.
Mensajes de voz	Sí	Sí
Canales	No	Sí
Respaldo de «chats» en la nube	Sí (en la versión de escritorio dependen del móvil)	Sí (en los servidores de Telegram)
«Chats» secretos	No	Sí
Creación de bots	No	Sí
Open Source	No	Sí
Existencia de una API	No	Sí
Multiplataforma	Sí	Sí

Tabla 3.1: WhatsApp vs Telegram



## 3.8 Aplicaciones Especializadas

En esta sección se presentan ciertas aplicaciones que, a diferencia de las mencionadas anteriormente, han sido específicamente desarrolladas para el sector educativo.

### 3.8.1 miColegioApp

Esta aplicación ofrece un canal de comunicación directo, oficial, inmediato y seguro entre centros escolares y familias. Para los colegios tiene una serie de ventajas como enviar circulares, mensajes personalizados, fotos o documentos pdf; conocer estadísticas en tiempo real sobre la lectura de notificaciones o la aplicación de filtros en el envío: por alumno, grupo, curso, actividad, etc. Para los padres tiene otras funcionalidades como confirmar citas o firmar autorizaciones. Está disponible para dispositivos Android, iOS y cuenta con una versión web [cre].



Figura 3.8: Logo de miColegioApp

### 3.8.2 BabyNotez

«BabyNotez» 3.9 está pensada para los centros de educación infantil. Su finalidad principal es la de sustituir, además de añadir características adicionales a la clásica agenda contando, por ejemplo, con una mensajería instantánea bidireccional. Cada día los profesores redactarán un pequeño informe que los padres recibirán con una notificación sobre cómo ha acontecido el día del alumno. Se trata de una aplicación gratuita para las familias, pero con un coste no determinado para el centro por la activación de la intranet, puesto que cada uno requiere de una pequeña red interna que gestiona la empresa responsable de la aplicación, necesaria para el funcionamiento de ésta. Se encuentra disponible para iOS y Android [Edu16].



Figura 3.9: Logo de BabyNotez

#### 3.8.3 TokApp School

Esta plataforma (Figura 3.10) proporciona una plataforma de gestión online desde la que el centro realiza todas las tareas de tipo organizacional, como dar de alta y gestionar alumnos y clases. A los padres y alumnos permite enviar mensajes y notificaciones, además de comunicarse con el centro y otros estudiantes. Al profesorado le evita el envío de circulares o documentos en papel puesto que la aplicación permite también el envío de archivos adjuntos. Es gratuita para los padres pero para los centros tiene un coste que no está especificado en su página web [Edu16].



Figura 3.10: Logo de TokApp School

# Metodología

EN este capítulo se van a exponer brevemente las características principales de la metodología elegida para el desarrollo de este trabajo fin de grado. En este caso se ha decidido escoger *Scrum*, una metodología ágil para la gestión de proyectos.

### 4.1 Metodología de gestión de proyectos: Scrum

Scrum [Sut17] es un marco de trabajo de procesos que se usa para la gestión del desarrollo de productos dentro del cual se pueden emplear diferentes procesos y técnicas. Posee equipos autogestionados con sus roles, eventos, artefactos y reglas asociadas. Esta metodología se basa en dividir el proyecto en diferentes fases, de manera que una fase no puede comenzar mientras la anterior no haya terminado. Dos de los términos más importantes en Scrum son el Sprint y la Pila de Producto (*Product Backlog*). El primero es un bloque de tiempo donde se crea un entregable del producto, mientras que el segundo es el conjunto de todo lo que podría ser necesario en el producto.

#### 4.1.1 Teoría de Scrum

Scrum se basa en la teoría de control de procesos empírica, lo que asegura que el conocimiento procede de la experiencia de la toma de decisiones basadas en lo que ya se conoce, empleando un enfoque iterativo e incremental. Sus tres pilares fundamentales son [Sut17]:

- **Transparencia.** Los aspectos significativos del proceso han de ser visibles para los responsables del resultado.
- **Inspección.** Los usuarios de Scrum deben inspeccionar con frecuencia los artefactos y el progreso para detectar variaciones indeseadas. Estas inspecciones no deben interferir en el trabajo.
- **Adaptación.** Si un inspector determina que uno o más aspectos de un proceso se desvían de ciertos límites y que el resultado será inaceptable, se procederá a un reajuste que deberá realizarse tan pronto como sea posible para evitar una posible desviación.

### 4.1.2 El equipo de Scrum

Cada equipo de Scrum [Sut17] se compone del **Dueño del Producto** (*Product Owner*), el **Equipo de Desarrollo** (*Development Team*) y un «**Maestro de Scrum**» (*Scrum Master*). Además, los equipos son autoorganizados y multifuncionales.

#### **Dueño del Producto** (*Product Owner*)

Es el responsable de maximizar el valor del producto desde el punto de vista del negocio, puesto que es el intermediario entre el cliente y el equipo. Es decir, el Dueño del Producto es el representante del cliente. Además, tiene la responsabilidad de controlar la Pila de Producto, control que se realiza mediante diferentes tareas como la de fijar sus ítems, ordenarlos, optimizar el valor del trabajo del equipo de desarrollo o asegurar que cada ítem se encuentra correctamente descrito.

#### **Equipo de Desarrollo** (*Development Team*)

Este equipo está formado por profesionales que entregan un incremento del producto terminado que se puede poner en producción al final de cada Sprint. El equipo de desarrollo tiene las siguientes características:

- **Son autoorganizados.** Nadie indica al equipo cómo convertir la Pila de Producto en incrementos.
- **Son multifuncionales.** Cuentan con las habilidades necesarias para crear un incremento.
- **No se reconocen títulos individuales.** El grupo es más importante que el individuo, es decir, el peso del trabajo que se realiza recae en todo el equipo.
- **No se reconocen subequipos.** Puede considerarse una derivación del punto anterior.
- **Cada miembro debe tener habilidades especializadas y áreas en las que enfocarse.**  
A pesar de esto, la responsabilidad siempre recaerá en el equipo como conjunto.

#### **«Maestro de Scrum»** (*Scrum Master*)

El *Scrum Master* [GGdlHdD17] es el responsable de que Scrum se entienda y se adopte y de que el equipo sea productivo. Principalmente, es un «facilitador». Trabaja muy cerca del dueño del producto y del equipo, protegiéndolo de interferencias externas, eliminando impedimentos y procurando que fluya la comunicación y la colaboración.

### 4.1.3 Eventos de Scrum

Existen eventos [Sut17] predefinidos cuyo fin es crear regularidad en el trabajo y minimizar la necesidad de realizar reuniones no definidas. Cada evento es un bloque de tiempo al que se le asocia una duración máxima.

## El Sprint

Es un bloque de tiempo cuya duración es, como máximo, de un mes donde se crea un incremento del producto. Un nuevo Sprint comienza inmediatamente después de la conclusión del anterior y una de sus principales características es que, en cada uno, se solapan todas las etapas de la creación de un producto. Es decir, en cada iteración se realiza la planificación, análisis, creación y comprobación del entregable. El Sprint se divide en una serie de etapas, que son la **Planificación del Sprint** (*Sprint Planning*), el **Scrum Diario** (*Daily Scrums*) el **Trabajo de Desarrollo** (*Development Work*), la **Revisión del Sprint** (*Sprint Review*) y la **Retrospectiva del Sprint** (*Sprint Retrospective*).

### Planificación del Sprint (*Sprint Planning*)

Esta tarea se realiza al comienzo de cada Sprint, cuya finalidad es la de programar el trabajo que se realizará durante el mismo [GGdIHdD17]. Antes, el dueño del producto revisa que la Pila de Producto se corresponde con las historias de usuario que le gustaría ver incluidas en la siguiente iteración, junto con su correcta descripción y priorización. La reunión debe terminar con unos objetivos completados: una lista de historias o *Sprint Backlog* (conjunto de historias de usuario y tareas en las que se dividen); un propósito para el Sprint que sugiere el dueño del producto; el compromiso del equipo de realizar las historias; la estimación del equipo del esfuerzo necesario para realizar cada historia; y que todos entiendan el contenido y el alcance de todas las historias.

### Scrum diario (*Daily Meeting*)

El Scrum diario [Sut17] es una reunión de corta duración para la sincronización de las actividades y la creación del plan de actividades por parte del equipo de desarrollo.

### Revisión del Sprint (*Sprint Review*)

Esta tarea se realiza al final de cada Sprint para inspeccionar el incremento y adaptar la Pila de Producto si fuera necesario. Se trata de una reunión informal en la que participan el equipo Scrum y los interesados.

### Retrospectiva del Sprint (*Sprint Retrospective*)

Aquí el equipo Scrum se inspecciona a sí mismo para crear un plan de mejoras para el siguiente Sprint. Se realiza después de la revisión y antes de la siguiente planificación.

### 4.1.4 Artefactos de Scrum

Los artefactos representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades de adaptación e inspección y deben estar al alcance de todos los participantes en el proyecto.

#### **Pila de Producto (*Product Backlog*)**

El *Product Backlog* es uno de los elementos fundamentales, siendo una lista de todo lo que podría ser necesario en el producto. Contiene las características, funcionalidades, requisitos, mejoras y correcciones que conforman cambios a realizar sobre el producto para entregas futuras. Su responsable es el dueño del producto, incluyendo su contenido, disponibilidad y ordenación. Esta lista va evolucionando a medida que el producto y el entorno en el que se usará también lo hacen. Esto quiere decir que es dinámica, cambiando constantemente.

#### **Pila de Sprint (*Sprint Backlog*)**

Esta lista [GGdlHdD17] contiene los trabajos a realizar en un Sprint determinado. Contiene las historias de usuario y las tareas identificadas por parte del equipo de desarrollo, que es quien gestiona esta lista. Al igual que la lista de producto, es dinámica y se va modificando durante el Sprint según se trabaja en lo planeado.

#### **Incremento**

El Incremento [Sut17] es la suma de todos los elementos de la lista de producto completados durante un Sprint y el valor de los Sprints anteriores.

#### **Gráficos *burn-down* y *burn-up***

Otros artefactos importantes son las gráficas. Principalmente se usa la gráfica *burn-down*, que consiste en representar el trabajo mediante dos ejes: el vertical representa una escala numérica del trabajo (historias de usuario o «puntos de historia») y el horizontal representa el tiempo. Además, se tiene una línea recta de la evolución «ideal» del trabajo y cada día se va comparando con lo que se ha realizado realmente para llegar finalmente a cero, lo que significaría que el trabajo se ha completado totalmente. Un ejemplo visual se encuentra en la Figura 4.1, donde también se puede observar la línea *velocity*, que representa la cantidad de trabajo terminado en cada iteración mediante «puntos de historia». Por otra parte, los gráficos *burn-up* muestran el trabajo realizado de manera creciente.

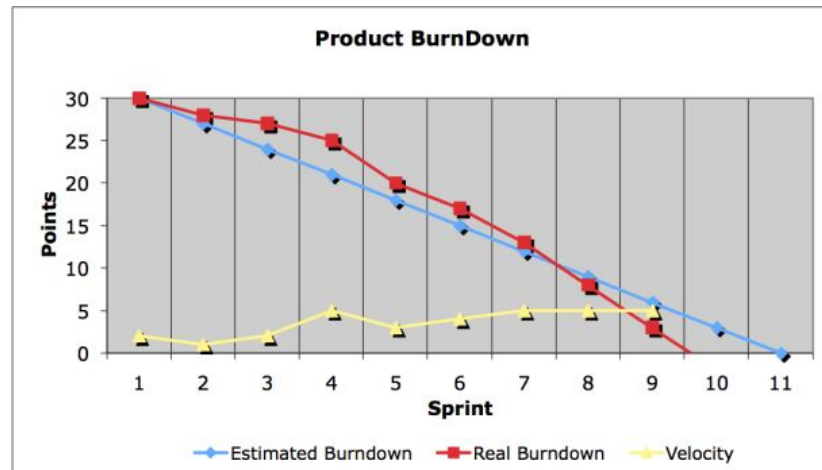


Figura 4.1: Burn-down chart

## 4.2 Metodología de Desarrollo de Software: Iterativo e Incremental

Así como la metodología de gestión de proyectos elegida ha sido Scrum, en cuanto al desarrollo del software se ha elegido el ciclo de vida iterativo e incremental. Este ciclo de vida consiste en desarrollar por partes el producto, integrándolas progresivamente conforme se van completando, agregando más funcionalidad al sistema final. Por otra parte, en cada iteración se revisa y mejora el producto, añadiendo nuevos requisitos o mejorando los ya existentes, repitiendo un proceso de trabajo similar [pro]. En cierto modo, se crean «miniproyectos» entre los que se reparte el trabajo y cada uno representa una iteración que generará un incremento como resultado. Cada uno de estos «miniproyectos» sigue el esquema análisis-diseño-pruebas. Si la iteración finalmente cubre los requisitos elegidos, la siguiente puede comenzar, de otro modo los requisitos han de ser revisados. Esta metodología tiene una serie de ventajas: el riesgo del proyecto se reduce a un incremento, se previene de lanzar software incompleto o de baja calidad, hace que el esfuerzo de los desarrolladores se concentre y sea más eficiente en cada iteración y se puede conseguir una mejor colección de requisitos.

## 4.3 Recursos Hardware y Software

Para el desarrollo de este trabajo no se requiere de hardware especial, al contrario que pudiera suceder con el software, puesto que se utilizan herramientas y equipos que están disponibles para cualquier usuario doméstico. El software, por su parte, resulta más específico para el desarrollo de este trabajo puesto que requiere de ciertos conocimientos más técnicos.

## 4. METODOLOGÍA

### 4.3.1 Recursos Hardware

Principalmente se va a usar un ordenador que ejecuta el sistema operativo Windows (Figura 4.2) para llevar a cabo la consecución del proyecto.

- **Marca y modelo:** Sony VAIO F-Series.
- **Procesador:** Intel® Core™ i7-720QM @ 1.6 Gigahercios (GHz).
- **RAM:** 8 GigaByte (GB).
- **Tarjeta Gráfica:** NVIDIA GeForce GT 330m.
- **Disco Duro:** 500 GB.

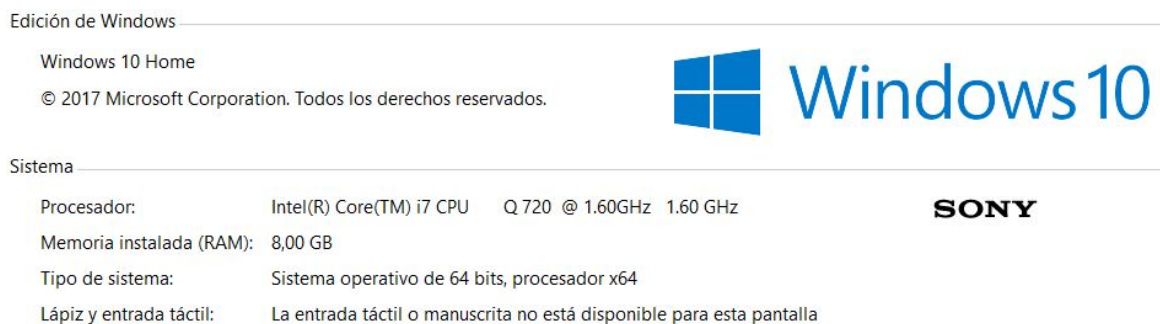


Figura 4.2: Información de Windows

Del mismo modo, se ha usado un *smartphone* para comprobar el correcto funcionamiento de la aplicación.

- **Marca y modelo:** LG Optimus L5 II.
- **Procesador:** MediaTek MT6575 @ 1 GHz.
- **Sistema Operativo:** Android 4.1.2 *Jelly Bean*.
- **RAM:** 1 GB.
- **Memoria interna:** 4 GB.

### 4.3.2 Recursos Software

Puesto que se pretende que la aplicación sea multiplataforma, se han usado diferentes sistemas operativos sobre los que se debe actuar para interactuar con dicha aplicación, así como diferentes programas para desarrollar las aplicaciones Android y Web.

#### Sistemas Operativos

Como sistemas operativos, se han usado Microsoft Windows 10 Home [Mic] en el PC y en cuanto al *smartphone*, se usará Android en su versión 4.1.2 *Jelly Bean* [Gooa].



## Lenguaje de Programación

Puesto que la aplicación está destinada a *smartphones* Android, el lenguaje de programación ha de ser Java. Java es un lenguaje de programación orientado a objetos usado para el desarrollo de aplicaciones cuyos propósitos son muy variados, puesto que también ofrece concurrencia [Ora]. Por otra parte, para la Web de gestión, se programará en HTML y JavaScript usando el entorno de programación *Eclipse Oxygen* [Ec118] .

## GitHub

GitHub es una plataforma que ofrece la posibilidad de crear repositorios para proyectos y así poder trabajar de manera sencilla en colaboración con otras personas, como podrían ser los diferentes integrantes del equipo de Scrum. También dispone de un apartado para cada repositorio llamada *projects* en la que se pueden crear tableros Kanban, que serán útiles durante el desarrollo del trabajo. Kanban [GGdlHdD17] es una palabra de origen japonés que significa signo, señal o tarjeta. Este tablero resulta de gran ayuda puesto que se pueden observar de un rápido vistazo las tareas que quedan por hacer, en las que se está trabajando y las terminadas de una manera visual, organizada y rápida (Figura 4.3).

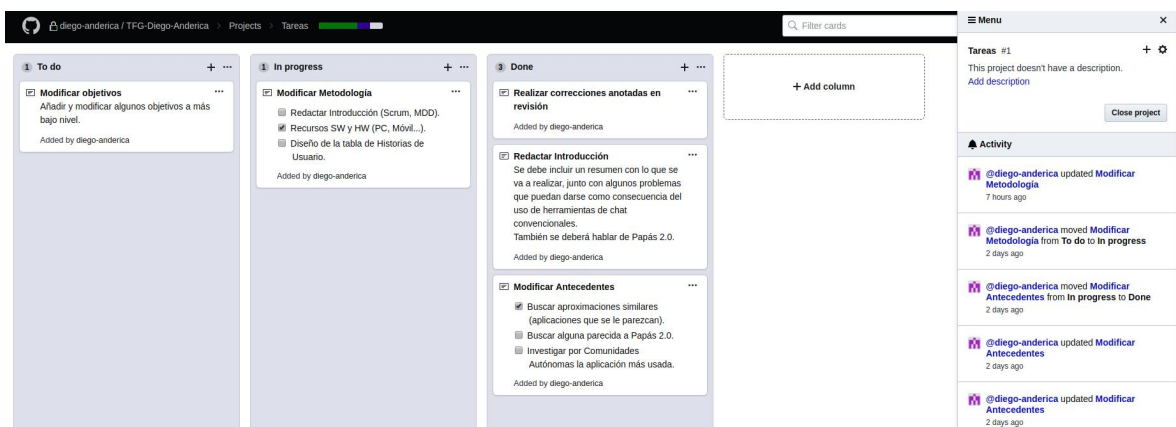


Figura 4.3: Tablero Kanban en GitHub

## Android Studio

Android Studio [And] es el entorno de programación oficial para el desarrollo de aplicaciones en Android, proporcionando las herramientas necesarias para ello. Además, posee integración con *Firebase*, lo que permite conectar las aplicaciones con este servicio para agregar *Analytics*, *Authentication* y *Notifications*, entre otros servicios, que resultarán imprescindibles para el desarrollo de este trabajo.

### Firestore

Firestore [Goob] es, principalmente, un *backend* que facilita las tareas de programación en el lado del servidor, puesto que proporciona acceso fácil a los recursos que ofrece. No sólo ofrece soporte al desarrollo de aplicaciones en Android, sino que también está disponible para su integración en iOS y aplicaciones Web. Algunas de sus funciones son:

- **Cloud Firestore.** Se trata de una base de datos en tiempo real, evolución de *Realtime Database*. Ofrece una base de datos no relacional.
- **Authentication.** Permite autenticar usuarios de forma simple en las aplicaciones de un proyecto. Además del usual método de entrada usando correo y contraseña, permite la autenticación mediante redes sociales y/o número de teléfono.
- **Remote Config.** Permite modificar la aplicación de manera remota en todos los clientes sin necesidad de implementar una nueva versión.

Este servicio posee tres maneras de tarificación:

- **Plan Spark.** Este es el plan más sencillo, con un coste gratuito. Posee ciertas limitaciones, aunque sería suficiente para un centro con un número de usuarios pequeño/medio ya que dichas limitaciones no son muy restrictivas.
- **Plan Flame.** Este plan tiene un coste de 25\$, con lo que se eliminan las restricciones del plan básico y sería apropiado para un centro con un número de usuarios relativamente grande.
- **Plan Blaze.** El último plan no tiene un precio definido puesto que se paga por lo que se vaya a utilizar, por lo que es el más flexible de los tres mencionados.

### IBM Bluemix

IBM Bluemix es una plataforma que permite el acceso a sus utilidades *cloud* de manera sencilla. Ofrece diversos servicios entre los que se encuentra IBM Watson. Este servicio ofrece tecnologías cognitivas para crear aplicaciones inteligentes aportando la posibilidad de analizar y comprender sentimientos o palabras claves a partir de un texto. A la hora de interpretar si un mensaje es adecuado o no para su envío se utilizará el módulo *Natural Language Understanding* [IBM].

### LaTeX

En cuanto a la documentación, se ha usado el lenguaje de generación de documentos  $\text{\LaTeX}$ , junto con la clase *esi-tfg* proporcionada [ARC].  $\text{\LaTeX}$  es un sistema de preparación de documentos de alta calidad tipográfica usado principalmente en documentos técnicos o científicos y permite a los autores centrarse más en el contenido [The].

## Capítulo 5

# Resultados

**D**URANTE el desarrollo de este capítulo se mostrarán los resultados obtenidos de seguir la planificación usando la metodología de desarrollo detallada en el capítulo anterior, por lo que el avance se dividirá en Sprints. Cada uno de ellos tendrá, al menos, una historia de usuario asignada que deberá quedar implementada al final del desarrollo.

Debido a que el equipo de Scrum, en este caso, es reducido, se han tenido que realizar diversos ajustes para adoptar esta metodología al trabajo que se va a desarrollar. Por tanto, el equipo identificado queda de la siguiente manera:

- **Dueño del Producto:** Luis Rodríguez Benítez.
- **Maestro de Scrum:** Luis Jiménez Linares.
- **Equipo de Scrum:** Diego Andérica Richard.

Este capítulo constará de los diferentes Sprints en los que se ha dividido el proyecto, así como su planificación, resultados y reajustes necesarios, si los hubiese. Antes de comenzar con el primer Sprint se ha de tener en cuenta uno de los principales artefactos de la metodología escogida, como es la Pila de Producto.

## 5.1 Planificación inicial

### 5.1.1 Pila de Producto

La Pila de Producto, como se ha explicado con anterioridad, se trata de una lista de todo lo que puede ser necesario en el producto. En este caso, la Pila de Producto es la siguiente:

## 5.2 Sprint 1: Diseño de una plataforma Web para la gestión de los usuarios

Este primer Sprint está enfocado a diseñar e implementar una plataforma Web para la gestión de los usuarios de la aplicación Android. De esta manera, se busca que el hecho de realizar tareas como las de añadir nuevos usuarios, consultar los existentes o modificarlos resulte mucho más sencillo, cómodo y práctico para los potenciales administradores de la plataforma de mensajería instantánea que se pretende desarrollar, puesto que ofrecerá una interfaz más amigable que la que ofrece actualmente la base de datos de segunda generación

## 5. RESULTADOS

de Firebase, Firestore.

### 5.2.1 Planificación del Sprint

Se ha elegido una única historia de usuario para la consecución de este Sprint durante la reunión inicial, representada en la Tabla 5.1.

Historia de Usuario	
Sprint Asignado: 1	
Número de Historia: 1	Usuario/Rol: Administrador
Nombre de la Historia: Gestión de usuarios de la aplicación móvil	
Prioridad: Alta	Duración: 30 horas
Descripción: Desarrollar una plataforma Web para facilitar la gestión de los usuarios de la aplicación móvil.	
Tareas: Diseño inicial de la base de datos Integrar Bootstrap y diseñar apariencia Web. Conectar aplicación Web con Firebase.	Pruebas:

Tabla 5.1: Historia de Usuario 1

### 5.2.2 Resultados del Sprint

#### Tarea de diseño de la base de datos

Firebase Firestore ofrece una base de datos no relacional, lo que significa que no existen relaciones ni tablas como sí existen en las relacionales. Por tanto, en este caso, se tienen colecciones de documentos que, a su vez, pueden tener colecciones de documentos anidadas. En consecuencia, se ha realizado un primer diseño en el que la base de datos tendrá tres colecciones: «Usuarios», para los usuarios de las familias; «UsuariosWeb», que albergará los administradores del sitio Web y «Docentes», que contendrá los datos de los docentes y potenciales administradores de los grupos de chat.

En cuanto a contraseñas, no es necesario guardarlas en la base de datos, a excepción de las de los administradores puesto que, al hacer uso de la funcionalidad de Firebase *Authentication*, éstas se guardan internamente en el proyecto y únicamente son conocidas por la herramienta, asignando además un identificador único de usuario a cada uno.

#### Tareas de implementación de la plataforma Web

En el centro habrá uno o varios administradores de la aplicación que se encargarán de gestionar los usuarios de la aplicación. Por tanto, se contemplan las siguientes acciones: dar de alta, dar de baja, consultar y modificar. La plataforma Web debe proporcionar estas acciones de una manera sencilla, vistosa y *responsive*. Para este propósito se ha decidido usar Bootstrap [Boo], que proporciona un conjunto de herramientas para desarrollo con HTML, CSS y JavaScript. Del mismo modo, se ha requerido incluir Firebase y las referencias a la base de datos de Firestore del proyecto. Por seguridad, el registro de un nuevo administrador en la base de datos se deberá llevar a cabo de manera manual, accediendo al proyecto de Firebase, donde se guardará un nuevo documento con el correo electrónico y la contraseña

cifrada mediante el algoritmo MD5. Del mismo modo, se ha implementado un método para evitar que un usuario acceda a una página conocida dentro del servidor mediante el uso de *Session Storage*. De esta manera, cuando el usuario accede correctamente con su correo y contraseña, el servidor genera y devuelve un número de sesión que se almacena en esta caché del navegador junto con el correo por lo que, si alguno de estos campos se encuentra sin definir en el momento de acceder a una página, se devuelve automáticamente al inicio de sesión. Una vez se ha accedido, el administrador accede a una página principal con cuatro botones y una barra de navegación superior desde donde puede realizar cuatro acciones principales: dar de alta, dar de baja, consultar usuarios y modificar usuarios. En los siguientes apartados se llevará a cabo una explicación de cada uno.

#### Dar de Alta

En esta primera funcionalidad, el administrador tiene la opción de rellenar un sencillo formulario para introducir un nuevo usuario en la base de datos que se puede rellenar con los datos de un único tutor legal del alumno o, si lo hubiese, también con los del segundo tutor. Los datos que se han especificado para cada tutor son: nombre, primer apellido, segundo apellido, número de teléfono y correo electrónico. Por último, para identificar a este usuario en la base de datos, se le asignará un identificador compuesto de los apellidos de ambos tutores o, en su defecto, del único tutor, añadiendo un número al final. Es decir, si dos familias diferentes poseen los mismos apellidos, se añadirá un número al final de la composición de sus apellidos para el identificador, por lo que quedarán representadas de manera única en la base de datos.

Por otra parte, además de tener el formulario para introducir nuevos usuarios, el administrador también tiene la posibilidad de importar un archivo CSV para realizar esta tarea de una manera más sencilla y automatizada. Este archivo debe tener un formato específico detallado en la primera línea del mismo y que consta, principalmente, de los mismos datos que en el formulario pero separados por comas y sin espacios entre ellos, separando los usuarios por un salto de línea.

#### Dar de Baja

Al entrar en esta página, se dispone de una lista desplegable en la que se cargarán todos los usuarios registrados en la base de datos. Web el diseño se basa en tener una lista desplegable con todos los usuarios de la base de datos. Al seleccionar uno de ellos, automáticamente se cargarán en una tabla los datos del mismo para poder ver el usuario que se desea eliminar. Como medida adicional contra el borrado accidental, se preguntará si realmente se desea borrar el usuario seleccionado.

#### Consultar Usuarios

De manera similar al apartado anterior, se ha implementado una tabla para consultar los datos de los usuarios aunque, en este caso, se dispone de una herramienta básica de búsqueda en la que se puede filtrar por cada uno de los campos existentes en la base de datos. Si se pulsa

## 5. RESULTADOS

el botón de buscar sin escribir nada en el filtro, la tabla se cargará con todos los registros existentes.

### Modificar Usuarios

Esta funcionalidad se ha implementado mediante el uso de una lista desplegable que carga todos los usuarios de la base de datos y un formulario que inicialmente se muestra bloqueado para no escribir de manera no intencionada y modificar los datos. Al seleccionar uno de estos usuarios, su información se carga en el formulario para realizar una primera inspección de la misma y, al pulsar el botón de modificar, se habilitan los campos para la escritura. Cuando el administrador finalice la modificación de la información, pulsará de nuevo un botón, siendo preguntado sobre si quiere realmente llevar a cabo la modificación. En caso positivo, se sobrescribirá la información del usuario seleccionado de la base de datos.

### 5.3 Ejemplo Historia de Usuario

Historia de Usuario	
Sprint Asignado: 1	
Número de Historia: 1	Usuario/Rol: Administrador
Nombre de la Historia: Gestión de usuarios de la aplicación móvil	
Prioridad: Alta	Duración: 30 horas
Descripción: Desarrollar una plataforma Web para facilitar la gestión de los usuarios de la aplicación móvil.	
Tareas: Diseño inicial de la base de datos Integrar Bootstrap y diseñar apariencia Web. Conectar aplicación Web con Firebase.	Pruebas:

Tabla 5.2: Historia de Usuario 2

# ANEXOS





## Un anexo de ejemplo

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.



## Referencias

- [Ali17] Marina Alias. Los grupos de WhatsApp de padres vuelven con alevosía: «Hasta comparten fotos de los deberes hechos», Septiembre 2017. url: [http://www.vozpopuli.com/altavoz/educacion/reglas-escritas-grupos-WhatsApp-colegio-redes-sociales-padres.0\\_1059194995.html](http://www.vozpopuli.com/altavoz/educacion/reglas-escritas-grupos-WhatsApp-colegio-redes-sociales-padres.0_1059194995.html).
- [And] Android. Android Studio. url: <https://developer.android.com/studio/index.html>.
- [ARC] ARCO. esi-tfg. url: [https://bitbucket.org/arco\\_group/esi-tfg](https://bitbucket.org/arco_group/esi-tfg).
- [Boo] Bootstrap. Bootstrap - The most popular HTML, CSS, and JS library in the world. url: <https://getbootstrap.com/>.
- [Con17] Consuelo. «Me fui del grupo de WhatsApp y ahora me tratan con distancia en la vida real», Agosto 2017. url: <http://www.lanacion.com.ar/2050884-me-fui-del-grupo-de-whatsapp-y-ahora-me-tratan-con-distancia-en-la-vida>
- [cre] creaTáctil. miColegioApp. url: <http://micolegioapp.com/wordpress/>.
- [Ecl18] Eclipse Foundation. Eclipse. <https://www.eclipse.org>, Enero 2018.
- [Edu16] Educo. Las 5 mejores apps para colegios, Noviembre 2016. url: <http://www.educo.es/las-5-mejores-apps-para-colegios/>.
- [Eli16] Javier Elio. Wickr Me, la mensajería instantánea ultrasegura que utiliza Mr Robot, Septiembre 2016. url: <https://elandroidelibre.elespanol.com/2016/09/wickr-me.html>.
- [GGdlHdD17] Carmen Lasa Gómez, Alonso Álvarez García, y Rafael de las Heras del Dedo. *Métodos Ágiles: Scrum, Kanban, Lean*. Anaya Multimedia, 2017.
- [Gooa] Google. Android. url: <https://www.android.com/>.
- [Goob] Google. Firebase. url: <https://firebase.google.com/?hl=es-419>.

- [IBM] IBM. Watson. url: <https://www.ibm.com/cloud-computing/bluemix/es/watson>.
- [JCC10] JCCM. *Manual Papás 2.0*, Septiembre 2010. url: <http://www.ies-ramonycajal.com/files/ManualPadresMadres.pdf>.
- [JCC17] JCCM. Papás 2.0, Febrero 2017. url: <http://www.educa.jccm.es/alumnado/es/papas>.
- [Mic] Microsoft. Windows. url: <https://www.microsoft.com/es-es/windows/>.
- [Nov14] Jaime Novoa. La historia de WhatsApp: de la nada absoluta a ser comprados por Facebook, Febrero 2014. url: <https://www.genbeta.com/movil/la-historia-de-whatsapp-de-la-nada-absoluta-a-ser-comprados-por-facebook>.
- [Ora] Oracle. Java. url: <https://www.java.com/es/about/>.
- [pro] proyectosagiles.org. Desarrollo iterativo e incremental. url: <https://proyectosagiles.org/desarrollo-iterativo-incremental/>.
- [PyM17] Caso de Estudio real: cómo Slack se ha convertido en la iniciativa empresarial que más rápido ha crecido de la historia, 2017. url: <https://www.recursosparapymes.com/premium/caso-de-estudio-real-como-slack-se-ha-convertido-en-la-iniciativa-empresarial-que-mas-r>.
- [Ram13] Iván Ramírez. Skype cumple 10 años: historia del programa de llamadas más famoso, Agosto 2013. url: <https://www.softonic.com/articulos/skype-10-aniversario>.
- [Ram17] Iván Ramírez. WhatsApp vs Telegram en 2017: ¿cuál es la mejor aplicación de mensajería?, Junio 2017. url: <https://www.xatakandroid.com/comunicacion-y-mensajeria/whatsapp-vs-telegram-en-2017-cual-es-la-mejor-aplicacion-de-mensajeria>.
- [Sig17] Signal. Características, 2017. url: <https://signal.org>.
- [Sky17a] Skype. Características, 2017. url: <https://www.skype.com/es/features/>.
- [Sky17b] Skype. Skype for Developers, 2017. url: <https://dev.skype.com>.
- [Sla17] Slack. Features, 2017. url: <https://slack.com/features>.
- [Sut17] Ken Schwaber; Jeff Sutherland. *The Scrum Guide*. Scrum, Noviembre 2017. url: <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>.

- [Tel17a] Telegram. Preguntas Frecuentes, 2017. url: <https://telegram.org/faq/es>.
- [Tel17b] Telegram. What can you do with Telegram?, 2017. url: <https://telegram.org>.
- [The] The Latex Project. An introduction to LaTeX. url: <https://www.latex-project.org/about/>.
- [Tho15] Knowlton Thomas. A Brief History of Slack, Septiembre 2015. url: <https://techvibes.com/2015/09/25/a-brief-history-of-slack-2015-09-25>.
- [Van17] Vanguardia. Un grupo de madres celebra que el colegio cambie de clase a un niño con Asperger, Septiembre 2017. url: <http://www.lavanguardia.com/vida/20170903/431027397175/madres-celebran-colegio-eche-nino-asperger.html>.
- [Vel13] J.J. Velasco. Historia de la Tecnología: Skype cumple 10 años, Agosto 2013. url: <https://hipertextual.com/2013/08/skype-decimo-aniversario>.
- [Wha17] WhatsApp. Funciones, 2017. url: <https://www.whatsapp.com/features/>.
- [Wic17] Wickr. Products, 2017. url: <https://www.wickr.com/products>.



Este documento fue editado y tipografiado con  $\text{\LaTeX}$  empleando la clase **esi-tfg** (versión 0.20180412) que se puede encontrar en:  
<https://bitbucket.org/arco.group/esi-tfg>

[respeta esta atribución al autor]

