



Generalitat de Catalunya
Departament
d'Ensenyament



INS Puig Castellar

Departament: Informàtica

Cicle Formatiu: CFGS Desenvolupament d'aplicacions multiplataforma

Mòdul Professional: MP12 Projecte

MEMORIA DEL PROYECTO

Alumne/s: Alejandro Pereira y Angel de Caldas

Professor: X

Data:

29/05/2020

MyPuig



Aquesta obra està subjecta a una llicència de
[Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

1 PART I. Gestión del proyecto	4
1.1 Introducción	4
1.2 Objetivos	4
1.3 Entorno del proyecto	5
1.3.1 Contexto y Justificación	5
1.3.2 Stakeholders	5
1.4 Alcance	6
1.4.1 Situación actual	6
1.4.2 Metas y posibles obstáculos	6
1.4.3 Metodología, validación y herramientas de seguimiento	6
1.5 Presupuesto	8
1.6 Planificación temporal	10
1.6.1 Planificación inicial	10
1.6.1.1 Puntos de control	10
1.6.1.2 Cambios respecto planificación inicial	10
1.6.2.3 Consecuencias de los cambios respecto a los objetivos	11
1.6.2 Planificación final	12
1.7 Leyes y normativa	12
2 PART II. Ejecución del proyecto	13
2.1 Analisis	13
2.1.1 Requisitos funcionales	13
2.1.2 Requisitos no funcionales	14
2.2 Diseño	14
2.2.1 Arquitectura	14
2.2.2 Seguridad	17
2.2.3 Persistencia	18
2.2.3.1 Authentication	18
2.2.3.2 Database	20
2.2.4 Interficie	22
2.2.4.1 Wireframe	22
2.2.4.2 Interacciones	23
2.2.4.3 Funcionalidades	27
2.2.5 Tecnología	30
2.3 Desarrollo	31
2.3.1 API REST de Moodle	33
2.4 Pruebas	35
2.4.1 Usabilidad	35
2.5 Lanzamiento	36

2.5.1 Tareas para la distribución	36
2.5.2 Publicación	36
2.6 Conclusions	37
2.7 Webgrafia	38
Annexos	40
1. Activar Servicios Web Moodle	40
2. Certbot	42
3. Freenom	43
4. Curl ejemplos y posibles obstáculos	44
5. Authorized keys y Keypair en AWS	46
6. Test de Usabilidad	46

1 PART I. Gestión del proyecto

1.1 Introducción

Resumen del proyecto:

MyPuig trata sobre una aplicación de teléfono destinada a la búsqueda de trabajo, como una bolsa. El objetivo es que cada una de las noticias sean extraídas a través de la api de moodle y que se muestran en nuestra aplicación, proporcionando funcionalidades de marcar como favoritos, ponerse en contacto con la persona que ha publicado la noticia, un sistema de filtrado de las noticias y a partir de esas noticias creadas tener la oportunidad de elegir si queremos que esas notificaciones nos lleguen al teléfono o al correo.

La idea de la aplicación es poder facilitar la visualización de las noticias a todas aquellas personas que estén interesadas en la búsqueda de trabajo o que simplemente estén interesadas en encontrar prácticas, un poco también la idea es que todas estas personas según sus intereses puedan elegir qué noticia es más adecuada para lo que están buscando.

El proyecto se llevará a cabo aplicando una adaptación de la metodología SCRUM, más adelante en el apartado [1.6 Planificación temporal](#) se especifica como se ha aplicado esta metodología dentro del proyecto.

Como objetivo final, se busca desarrollar un inicio en el que las noticias son extraídas de un foro de Moodle, apartar un doble registro en el que en el punto [2.2.4.2 Funcionalidades](#) especificamos de que tratara. Todo esto será extraído de la API REST de Moodle y se trabajará en el lenguaje de programación Java.

Palabras clave (entre 4 y 8):

moodle, trabajo, borsa, mypuig

1.2 Objetivos

- Desarrollar la funcionalidad de favoritos.
- Desarrollar la funcionalidad de ayuda con despleables.
- Instalar Moodle.
- Acceder de manera remota al Moodle.
- Aprender a utilizar EC2 de AWS Student
- Utilizar la API REST de Moodle para sacar los post de un foro.
- Utilizar la API REST de Moodle para sacar datos del usuario.

1.3 Entorno del proyecto

1.3.1 Contexto y Justificación

El contexto del proyecto surge a partir de la falta de interés por parte de los alumnos de los centros educativos, de cara a la inserción en el mundo laboral y la necesidad de ver de una manera mucho más accesible las ofertas laborales. El objetivo de esto es conseguir que los alumnos se consiga interesar más por las entradas insertadas diariamente por los profesores y por ello promover el conseguir trabajo gracias a los centros.

Al ser un algo que ya tiene moodle, el objetivo es todas cosas de las que carece poder añadirlas, consiguiendo así que mucha más gente relacionada con el centro se interese por conseguir trabajo. La idea es que la aplicación y la web del centro funcionen de manera paralelas, es decir que los profesores que son los encargados de utilizar Moodle para añadir las entradas al post y dependiendo de estas los usuarios que utilicen la aplicación podrán visualizar las diferentes ofertas. La idea de esto es muy parecida a la del funcionamiento del Moodle, el objetivo es que los profesores de manera externa a la aplicación puedan gestionar qué tipos de noticias y cuales quieren que vean los alumnos, haciendo así que los profesores tengan el control del foro, y de cara a los alumnos añadir interacciones adicionales que en Moodle no permite, haciendo así que el usuario pueda controlar qué ofertas son de su interés o cuales prefiere filtrar.

En conclusión el objetivo es que todos los centros en los que utilicen foros en aplicaciones Web con Moodle para introducir ofertas de trabajo puedan utilizar esta aplicación para poder captar el interés de los alumnos en la búsqueda de trabajo.

1.3.2 Stakeholders

Al estar relacionada la aplicación con una necesidad generada por los alumnos y profesores de los centros educativos, esto quiere decir que cualquier persona que participe de manera interna en el proyecto, accionistas, directivos, trabajadores; formarán parte de los stakeholders primarios, es decir serán los que tengan la mayor influencia en desarrollo del proyecto debido a la relación con la empresa que se encargará de la aplicación. Además todas aquellos institutos en los que de manera interna influyen en la aplicación de manera menos formal, cliente, alumnos, profesores, formarán parte de unos Stakeholders muchos más secundarios o externos en los que su opinión influirá, pero no de una manera tan imperativa.



1.4 Alcance

1.4.1 Situación actual

La situación actual de la bolsa de trabajo es que funciona como un foro de moodle en el que diferentes profesores de los centros escriben noticias de ofertas de trabajo. El objetivo de esto es facilitar la ayuda del alumnado de Formación Profesional, en la búsqueda de trabajo. El problema de este sistema es la poca accesibilidad que comporta y el poco interés por parte del alumnado, partiendo desde este punto de partida en la que estos foros no tienen demasiado éxito, el objetivo es despertar todo ese interés de los alumnos para la búsqueda de trabajo sea mucho más fácil.

Por eso *MyPuig* pretende ofrecer una aplicación que funcione como plus a todos los institutos que quieran fomentar el empleo y la búsqueda de trabajo gracias a la Formación Profesional.

1.4.2 Metas y posibles obstáculos

El objetivo de este proyecto es poder presentar una aplicación con una base de funcionalidades, esto quiere decir que tendrá una serie de requisitos en los que se clasifican en 3 según la importancia de ellos. Por ello el objetivo de esto es que según vayan apareciendo resultados y se vayan presentando una serie de obstáculos ir desarrollando esos requisitos mínimos.

Por ello la idea es cumplir los requisitos establecidos en el análisis de ellos y a partir de esa base y de los resultados conseguidos comenzar a trabajar.

Además se ha decidido que la aplicación que tendrá un tiempo de vida bastante amplio, con esto se quiere transmitir que la aplicación estará en constante desarrollo y se harán en futuro funcionalidades en las que pueden ser más costosas como el Chat o los Filtros, por ello se pretende conseguir que la app tenga un constante soporte y que haya feedback y colaboración por parte del usuarios finales.

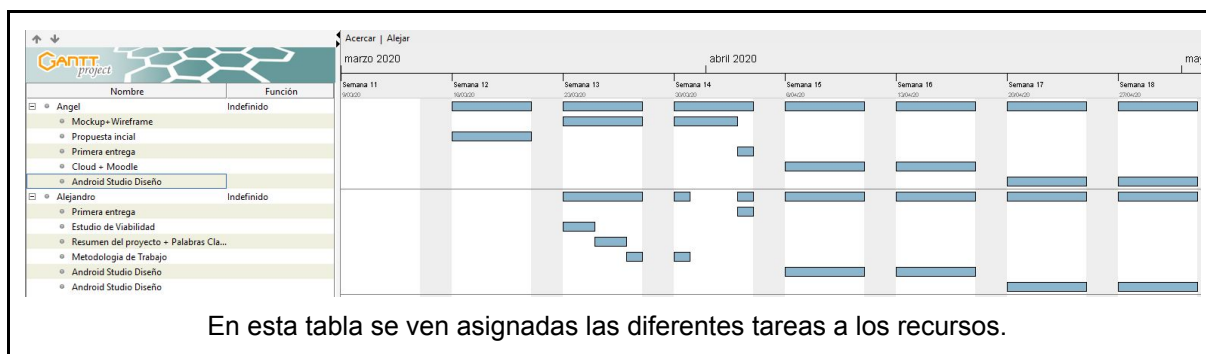
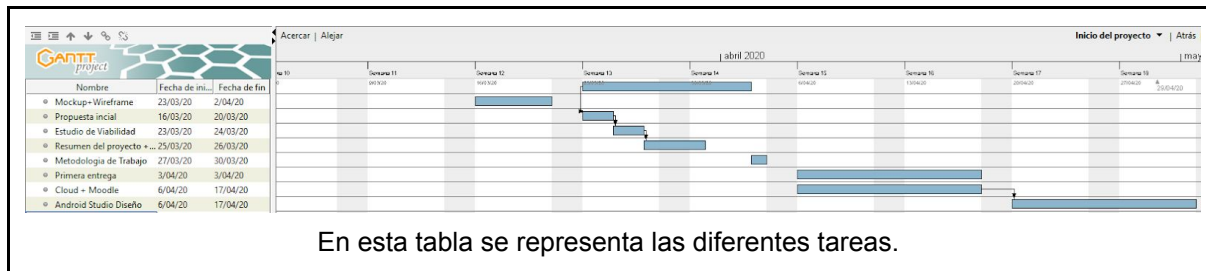
1.4.3 Metodología, validación y herramientas de seguimiento

En una primera fase o fase inicial se comienza a utilizar los diagramas de gantt facilitando así fases como la del análisis o el diseño, esto lo que permite es hacer cierto estudio de la viabilidad demostrando así que es factible comenzar con el diseño o no, además también es muy útil para comenzar con una idea inicial de las tareas de diseño y requisitos que la componen.

El objetivo de este diagrama es tener una primera idea de las tareas que compondrán el proyecto, por ello es necesario centrar la atención en crear un

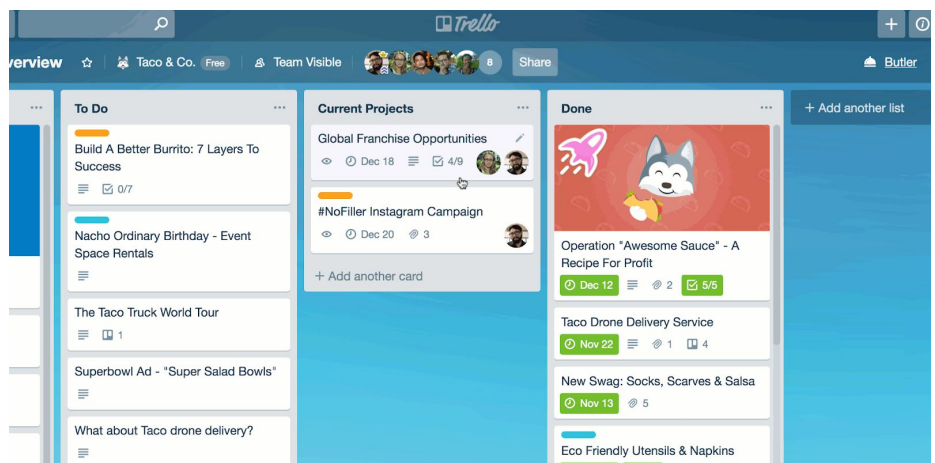
diagrama perfecto porque no será la principal metodología que se utilizara durante el proyecto y según se vayan presentando diferentes dificultades el diagrama simplemente actuaría como una guía.

Por ello hemos utilizado GanttProject para asignar y crear las diferentes tareas a los recursos (humanos) disponibles en el proyecto:



Una vez superada esa primera fase del proyecto, el objetivo es utilizar una metodología ágil. Por ello, el objetivo es utilizar SCRUM haciendo así reuniones diarias en las que poner en común información y saber la evolución de los diferentes integrantes del proyecto, por lo tanto para facilitar la utilización de esta metodología hemos utilizado una herramienta web llamada Trello.

Trello se basa en el método Kanban, este sirve para gestionar las tareas permitiendo organizar el trabajo en grupo de forma colaborativa, utilizando tableros compuestos por diferentes tareas en formas de columnas en las que se especifica el estado de cada una de las tareas y facilitando la visualización de la evolución del proyecto.



La idea principal para comenzar la a implementar el método Kanban es que todas aquellas tareas que se están realizando se tienen que priorizar antes de asignar nuevas tareas, por ello durante el proyecto uno de los objetivos principales es garantizar el inicio y finalización de la tareas antes de asignar nuevas.

En conclusión, el utilizar tantas metodologías permite sacar el máximo provecho a cada una de ellas haciendo así que todo lo mejor de cada una de ellas se utilice para una mejor evolución y desarrollo del proyecto, por ello se trata de aprovechar la flexibilidad y la entrada de las tareas, seguimiento de ellas, priorización con el método Kanban, para la supervisión del equipo de trabajo SCRUM aprovechando las reuniones diarias para el seguimiento y resolución de las diferentes dificultades presentadas durante la fase de desarrollo y Gantt para poder estudiar y medir la viabilidad del proyecto.

1.5 Presupuesto

Aún teniendo en cuenta que el presupuesto no es real y es una simple simulación, se ha calculado como si de un hecho real se tratase. El presupuesto inicial se aproxima a los 23.846€ aunque este presupuesto no es el definitivo ya que en el caso de que se realizarán actualizaciones debidos a cambios en el proyecto, este presupuesto variaría.

Todo esto es debido a que aun teniendo el margen de cálculo hay costes que son variables. Se pueden tener un idea inicial, pero dependiendo de las diferentes decisiones tomadas durante el proyecto, factores como el tiempo y los recursos harían variar ese presupuesto inicial.

El objetivo es que durante el proyecto, teniendo en cuenta un presupuesto flexible que pueda ir variando, aun dependiendo de diferentes decisiones el presupuesto final sea muy cercano a la cifra estimada en el inicial.

Los elementos a tener en cuenta para poder calcular el presupuesto serían los siguientes:

- **Recursos Humanos:** Salario de cada uno de los roles implicados en el proyecto:
 - Diseñador
 - Desarrollador
 - Tester
 - Analista
- **Material:** Hardware y Software
 - Equipo de trabajo
 - Licencias

- Servidor virtual en la nube
- Firebase

Nombre	Duración (días)	Coste
Aplicación	120	9.799,86 €
Analisis	10	940,86 €
Diseño	45	2.790 €
Desarrollo	65	3.569 €
Material y Licencias		2.500€
Proyecto	120	9.799,86 €

Presupuesto inicial del proyecto.

El salario de los diferentes roles que forman parte del proyecto han sido asignados comparando entre diferentes ofertas de **InfoJobs**.

Teniendo en cuenta factores, como la inexperiencia de los desarrolladores y partiendo de que los integrantes del proyecto tienen un perfil junior, se les han asignado estos salarios:

	Analista	Diseñador	Desarrollador
Anual	19.000€	22.320€	21.000€
Mensual	1538€	1860€	1750€
Día	93€	63€	23€

Además teniendo en cuenta factores, como por ejemplo el tiempo y problemas particulares del proyecto, para tener controlado el dinero gastado en la máquina en que tener el moodle se hizo esta pequeña tabla para ver qué tipo de las instancias posibles, era la más óptima:

Instancia	vCPU	ECU	GiB		\$/hora
t2.nano	1	Variable	0,5 GiB	Solo EBS	0,0058
t2.micro	1	Variable	1 GiB	Solo EBS	0,0116
t2.small	1	Variable	2 GiB	Solo EBS	0,023
t2.medium*	2	Variable	4 GiB	Solo EBS	0,0464
t2.large	2	Variable	8 GiB	Solo EBS	0,0928
t2.xlarge	4	Variable	16 GiB	Solo EBS	0,1856

t2.2xlarge	8	Variable	32 GiB	Solo EBS	0,3712
------------	---	----------	--------	----------	--------

1.6 Planificación temporal

1.6.1 Planificación inicial

1.6.1.1 Puntos de control

En cuanto a los puntos de control han sido bastante marcados, ya que el proyecto consta de tres entregas, estas están divididas en un término de tiempo de un mes para ver el progreso o evolución. Por ello se han impuesto una serie de objetivos entre los tres puntos de control:

- **Punto de control 1:** En este primer punto de control se ha decidido tener en cuenta las diferentes tareas que tiene que hacer cada una de las personas que realizarán el proyecto, a la misma hacer un estudio de viabilidad para ver si todas las funcionalidades propuestas son posibles de realizar para la entrega final o el último punto de control.

En cuanto a aspectos de la memoria, hacer un resumen introductorio ([1.1 Introducción](#)) y hacer el diagrama de gantt para facilitar el estudio de viabilidad y la reparto de tareas.

- **Punto de control 2:** En este segundo control los objetivos impuestos son, tener la memoria bastante avanzada y en cuanto a temas de diseño de la aplicación es tener las interacciones y funcionalidades simuladas para poder realizar un test, para poder medir todos esos atributos, funcionalidades o interacciones si son lo suficiente intuitivos y eficientes.

Por ello se realiza un test de usabilidad para poder medir todas esas métricas y atributos relacionados con las interacciones y funcionalidades que utiliza el usuario son lo suficiente para mejorar todos esos detalles o errores durante el desarrollo.

- **Punto de control 3:**

En este tercer y último control se ha decidido terminar de pulir ciertos aspectos de diseño gracias al test de usabilidad realizado en el anterior punto, además de implementar el código que conecta al usuario con Moodle, es decir la conexión mediante un token para poder obtener los datos finales (nombre, correo, etc).

También debe estar relacionado firebase con la app de manera que se almacene los datos del usuario y las noticias, esto permitirá al usuario final acceder a dichas noticias si se han marcado como favoritas o si se han utilizado los filtros.

1.6.1.2 Cambios respecto planificación inicial

Uno de los primeros pasos es analizar dónde estará el Moodle en el que se extraerán las noticias consultadas la API REST. Una de las primeras ideas era tener todo en una máquina virtual, pero debido a la incomodidad por temas de IP, DNS, etc, se propuso utilizar una plataforma en la que poner nuestro Moodle en la nube y poder acceder desde cualquier lugar. Al principio la idea era analizar las diferentes posibilidades. Entre ellas el Cloud de Google y AWS Student, finalmente y una vez comparadas las dos el objetivo era decidir cuál era la más adecuada al proyecto, ya que al tratarse de un proyecto orientado a DAM el objetivo es no restar importancia al Desarrollo. Por lo tanto instalar Moodle y configurar los diferentes puertos que iba a utilizar el proyecto, en AWS.

Una vez con la instancia o servidor creado uno de los primeros problemas presentados durante el proyecto es que la API REST no acepta ni páginas HTTP, ni tampoco paginas sin un dominio (Enlace en el [anexo](#)), por lo tanto cambiar la página a HTTPS y poner un dominio era uno de los primeros requisitos para comenzar a Desarrollar.

En cuanto a términos relacionados con el Desarrollo uno de los primeros objetivos era terminar la interfaz para comenzar las pruebas de Usabilidad. En estas pruebas de Usabilidad el objetivo es recoger información de todos esos aspectos que no resultan satisfactorios al usuario, por lo tanto iconos, colores, letras, tamaños son cambios que son lógicos contemplar a la hora de realizar este tipo de pruebas.

Otro de los cambios y más importantes era el trabajar con la API REST y al dar tanta importancia al tema principal de la aplicación que son los Post del forum, el objetivo es comenzar a programar código de las diferentes consultas realizadas a la API, el cambio respecto a esto es que debido al poco conocimiento del equipo del desarrollo sobre el funcionamiento y la formación de las consultas que se deben realizar, se pasó de desarrollar a utilizar la herramienta curl y obtener éxito, acto seguido comenzar con la programación (Enlace en el [anexo](#)).

Y por último analizar qué funcionalidades se quedaban fuera de nuestro alcance debido a factores como el tiempo, etc. Por lo tanto las funcionalidades de Filtros, Chat y Alertas se quedan para un desarrollo futuro en que la aplicación tenga una base sólida de funcionalidades para que el usuario pueda comenzar a interactuar con la app. Por ello estas funcionalidades a partir del lanzamiento de la app se comenzarán a desarrollar y serán lanzadas como actualizaciones para captar cada vez más público.

1.6.2.3 Consecuencias de los cambios respecto a los objetivos

Los objetivos propuestos están relacionados con la API REST y su utilización. La finalidad de esto es centrar la atención en el punto fuerte del proyecto y en el caso de terminar antes de los límites establecidos, pensar en el futuro y en nuevos objetivos.

La finalidad de pensar en el futuro es que funcionalidades que aún no están desarrolladas se puedan llevar a cabo, si los puntos control y los plazos de estos permiten se podrían desarrollar funcionalidades como la de comunicar al profesor que ha hecho el post de la noticia con el alumno interesado en la oferta de trabajo, la de poder filtrar las diferentes noticias por diferentes parámetros como por ejemplo etiquetas y el poder guardar todas esos criterios de búsqueda en posibles alertas en las que poder notificar al usuario de nuevos posts.

1.6.2 Planificación final

En los últimos puntos de control el objetivo de esta planificación, es analizar los puntos fuertes de la aplicación y intentar mejorar esas funcionalidades y aspectos del diseño.

Una de las primeras tareas es la de analizar si los objetivos principales se han cumplido satisfactoriamente y a partir de ahí mejorar todos esos pequeños errores.

Por eso en funcionalidades como OnBoarding, Ayuda añadir más contenido para que cobren más sentido, por ello en el OnBoarding y en la Ayuda el objetivo es añadir más información para facilitar el uso al usuario.

1.7 Leyes y normativa

Este proyecto se desarrolla siguiendo las directrices de la ley de protección de datos LOPD. Se ha utilizado esta ley debido a que la aplicación utilizará un token con el cual se puede acceder a toda la información del perfil de Moodle y se pretende proteger y garantizar la intimidad personal de cada usuario.

La ley de protección de datos es considerada un derecho que deben de tener todas las personas, y es el control que tienen los demás sobre los datos personales. En nuestro proyecto buscamos que ese control de datos, evite el tratamiento de datos personales y el no comprometer y afectar a los usuarios, respetando sus derechos y libertades.

Por ello, la ley obliga ya sea persona, empresa o organismo a que en el caso de disponer de datos de carácter personal a cumplir una serie de requisitos y aplicar una serie de medidas seguras en función del tipo de datos, por ello se tiene que tener en cuenta que hay obligaciones legales:

- Dar de alta los ficheros en la Agencia Española de Protección de datos.
- Elaborar y mantener actualizado el Documento de Seguridad.
- Obtener la legitimidad de los afectados.

Y en el caso de infringir alguna sanción, dependiendo de la gravedad de la falta. En el caso de cometer una falta leve, como por ejemplo la omisión de la declaración de un fichero rondaría los 60.000€ y en el caso de una cesión de datos sin consentimiento 600.000€.

Por ello hay páginas en las que te facilitan una ayuda y en las que es muchísimo más fácil garantizar al usuario el cumplimiento de esta ley, por ello se ha decidido utilizar [Secitor](#) (herramienta online).

2 PART II. Execució del projecte

2.1 Analisis

2.1.1 Requisitos funcionales

Las funciones del sistema relacionadas con el registro y la visualización de las ofertas de trabajo añadidas en el foro de Moodle estarán relacionadas con la API REST.

El objetivo de es que funcionalidades como el registro puedan permitir al usuario más accesibilidad al iniciar sesión o registrarse con su cuenta de Google, esto permitirá que cuando se inicie sesión por primera vez con su cuenta de Moodle el UID generado de su cuenta se quede almacenado en el Database de Firebase. El objetivo de esta funcionalidad es poder permitir al usuario iniciar sesión una única vez para no tener que estar iniciando sesión en la cuenta de Moodle, cada vez que quiera ver la Bolsa de trabajo.

Después la funcionalidad de consultar a la API para la visualizar las ofertas de trabajo, en las que serán mostradas a partir del token que se generará al iniciar sesión (cuenta de Moodle), y serán mostrados en una lista en las que se podrán marcar como favoritos.

La funcionalidad de favoritos funcionara de la misma manera que la del home, con la única diferencia que cada uno de los usuarios tendrá que tener guardado los favoritos que ha marcado respectivamente, en cada una de las ofertas o posts. Esto significa que cada usuario tendrá guardado en el Database las noticias marcadas como favoritos y ha partir de estas noticias almacenadas serán mostradas en esta funcionalidad.

Cada uno de los usuarios tendrá la oportunidad de poder añadir una serie de criterios de búsqueda en la visualización de noticias, además podrá guardar esos filtros en una funcionalidad en la que podrá acceder a esos criterios de búsqueda que previamente había guardado.

En la sección de Chat el objetivo es conectar el alumno con el profesor, haciendo que la persona interesada pueda hablar de una forma mucho más directa con la persona que ha hecho esa entrada al foro.

Y por último es que el usuario pueda acceder y entender de forma fácil todas las secciones gracias a una funcionalidad conjunta que es la de ayuda y la de

Onboarding al comenzar a utilizar la aplicación, prestando así una ayuda inicial al usuario.

2.1.2 Requisitos no funcionales

En cuanto requisitos no funcionales al tener una serie de objetivos impuestos por el cliente, relacionados con la usabilidad, diseño, etc. El objetivo de estos requisitos es cumplir la mayor parte de ellos o en su totalidad, ya que al hacer el análisis de requisitos, el objetivo de esto es que cosas no funcionales hagan que cada vez más alumnos se interesen por la aplicación.

Por lo tanto en aspectos de interfaz o diseño el objetivo es cumplir una serie de estándares relacionados con el color, iconos. El objetivo de esto gracias a los colores y iconos volver una aplicación que gráficamente sea seria pero captar la atención de los usuarios, por ello se han utilizado colores como el azul, blanco y negro. En cuanto a los iconos al utilizar la mayor parte de material de Google lo ideal es utilizar iconos de ellos y no otros diseñados desde cero.

En cuanto a aspecto de usabilidad, el objetivo es que el usuario sin necesidad de llegar a la funcionalidad de Ayuda sea capaz de saber utilizar la mayor parte de la aplicación, y en el caso de no saber algo cómo funciona que la ayuda sea efectiva y no confunda al usuario más de lo que está.

En cuanto a aspectos de seguridad queremos que el usuario se sienta lo más tranquilo posible sin tener que estar preocupado de sus datos personales o sus cuentas, servirán para el beneficio propio de la empresa por ello Firebase prohíbe por ciertas leyes que no se puede compartir la información a la ligera (se especifica en el punto [2.2.2 Seguridad](#)), además internamente el proyecto seguirá las directrices de la ley LOPD y por ello el objetivo de esta aplicación es el seguimiento y cumplimiento de estas leyes será utilizada una aplicación web llamada *Secitor* en la que se facilitará el trabajo (especificado en el punto [1.7 Leyes y normativa](#)).

2.2 Diseño

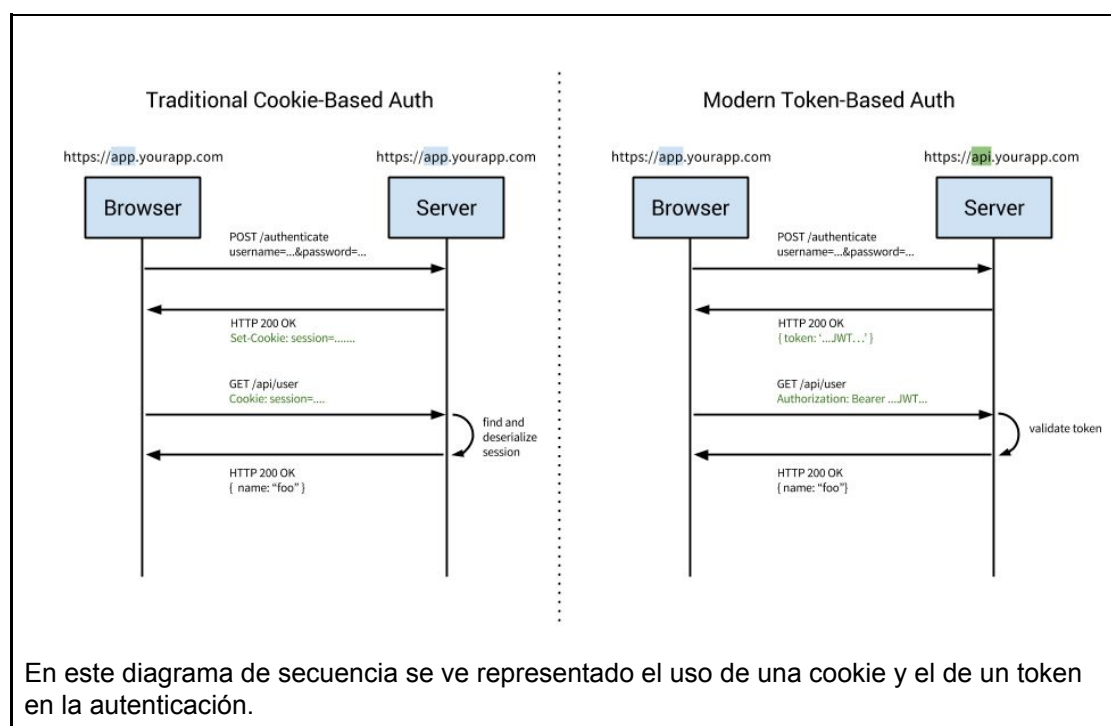
2.2.1 Arquitectura

Para entender bien en qué consiste la arquitectura de la aplicación se debería saber en qué consiste el tema principal de ella, la API REST. Por ello hay tener una serie de aspectos en cuenta:

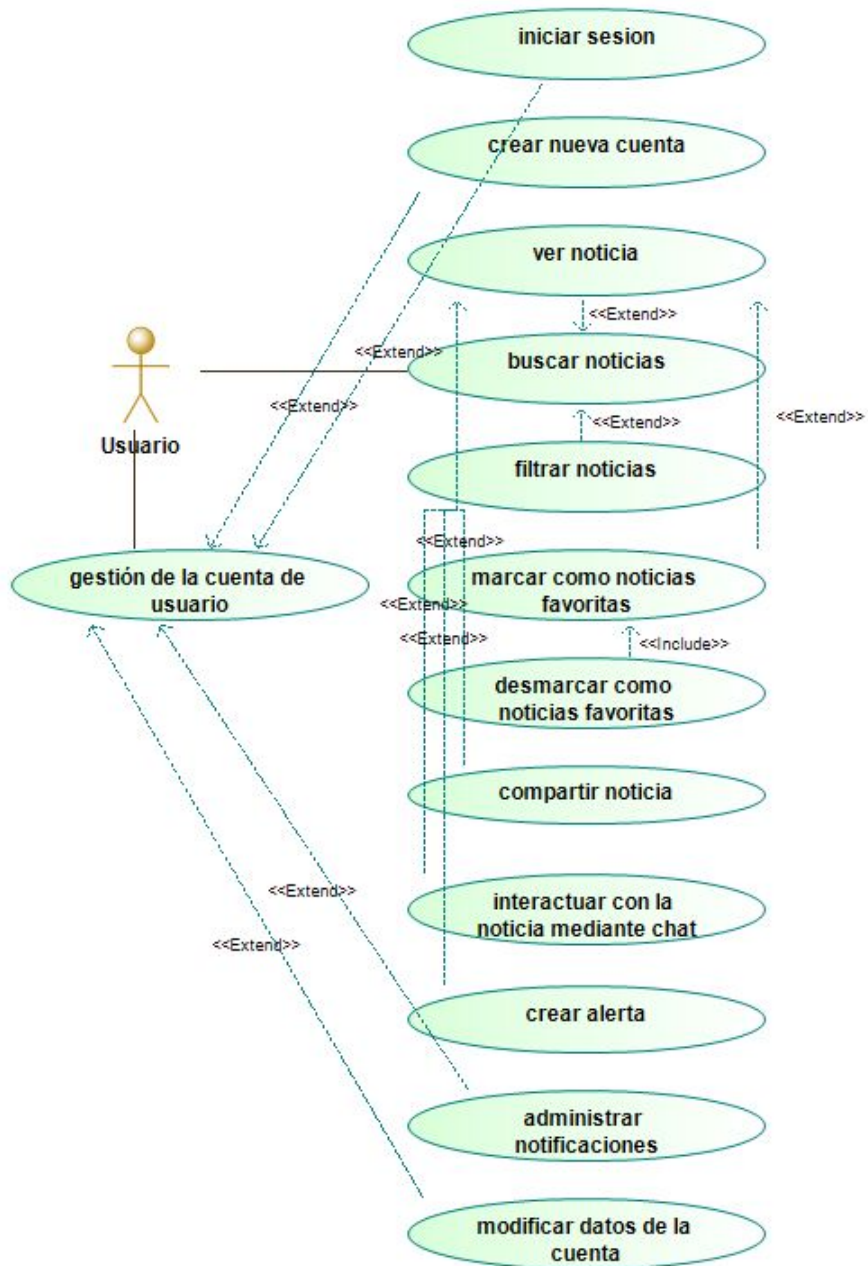
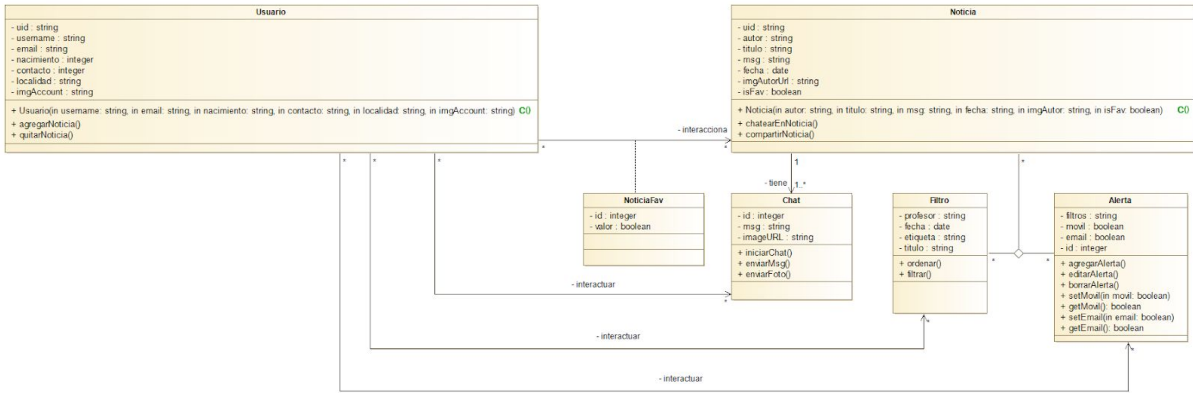
- **La interfaz es uniforme.** Está basada en recursos (id, nombres, apellidos), el servidor es capaz de mandar datos (en el caso de Moodle en XML y JSON) y además es fácil de gestionar los diferentes recursos. Además utiliza una

propia estructura y características para su uso, en el [2.3.1 API REST de Moodle](#) se especifica.

- **Peticiones sin estado.** Esto quiere decir que según la petición que se realice, en el caso de realizarla y que sobren o necesite recursos esta no dará respuesta (permite proporcionar un mayor rendimiento).
- **Métodos de autenticación.** Este consiste en utilizar dos formas de autenticación, basada en cookies o en token, en el proyecto el objetivo es realizar la autenticación con tokens. La ventaja de estos es que permiten una gran accesibilidad al usuario, esto permite que el usuario final pueda realizar acciones sin estar proporcionando credenciales constantemente. Este token normalmente es calculado con un hash de algún dato, en este caso la contraseña y el usuario.



En este diagrama de secuencia se ve representado el uso de una cookie y el de un token en la autenticación.



2.2.2 Seguridad

En cuanto a términos de seguridad tenemos que tener en cuenta dos grandes componentes de nuestro proyecto, el AWS en el que instalaremos la instancia o servidor con el Moodle y además Firebase en el que guardaremos información del usuario, noticias, etc. Por ello se ha clasificado en dos bloques:

Firestore.

Este componente se ha utilizado para la autenticación y el almacenamiento de las noticias o post favoritos en el foro de Moodle. Por ello se debe tener en cuenta que según el servicio que se utiliza en Firestore, estos tienen unas condiciones diferentes.

Por ello para las funcionales que se han utilizado se tiene que tener en cuenta las Condiciones de Seguridad y Procesamiento de Datos (GDPR), este impone obligaciones a los controladores de los datos y a los procesadores, en este caso el cliente actúa como “Controlador de datos” de la información personal sobre el usuario final.

Esto quiere decir que cliente está bajo control sobre los datos que guarda, por lo tanto es responsable de cumplir los derechos de una persona física en términos de datos personales.

Servicio Utilizado	Datos que guarda	Que hace ?	Que retiene ?
Cloud Functions para Firebase (Storage)	- Direcciones IP	Utiliza las direcciones IP para ejecutar funciones, en este caso de almacenaje de datos y es capaz de controlar eventos y funciones de HTTP basadas en las acciones del usuario final.	Almacena las direcciones IP de manera temporal para proporcionar servicios al usuario que la utiliza.
Firestore Authentication	- Contraseña - Direcciones de correo electrónico - Numeros de telefono - Usuarios	Usa los datos para habilitar la autenticación y facilitar la administración de cuentas.	Mantiene las direcciones IP guardadas durante unas semanas. El resto de la información del usuario es almacenada hasta que el cliente elimina el usuario, tras lo

	- Direcciones IP		cual se quitan los datos y quedan activos durante 180 días en el caso de recuperarlos.
--	---------------------	--	--

Además Firebase encripta todos esos datos que se transmiten en HTTPS y datos de clientes aislados. Esto quiere decir que Firebase aplica estas medidas de seguridad para mantener protegidos y reducir el riesgo de acceso de datos. Por ello estos servicios aplican una serie de medidas de seguridad eficaces:

- Firebase restringe el acceso a una selección de empleados que tienen un fin comercial para acceder a los datos personales.
- Firebase registra el acceso de los empleados a los sistemas que contienen datos personales.
- Firebase solo permite que accedan a los datos personales los empleados que lo hacen con el Acceso con Google y la autenticación de dos factores.

Amazon Web Services Educate.

Este componente ha sido utilizado para la instalación de un Server en el que poner un Moodle que actúa como cliente. El servicio está compuesto por muchos sub servicios o funcionalidades, en este caso se ha utilizado EC2. Amazon EC2 utiliza criptografía de clave pública para cifrar y descifrar información de inicio de sesión. La criptografía de clave pública usa una clave pública para encriptar un dato, y luego el destinatario usa la clave privada para descifrar los datos (keypair).

Las claves que utiliza Amazon EC2 son claves RSA SSH-2, Amazon almacena la clave pública y el usuario que utilizará la instancia la privada. Esto quiere decir que si cualquier persona se hace poseedor de la clave privada, este puede descifrar toda la información almacenada (EC2 no guarda la clave privada, si se pierde no hay manera de recuperarla).

(Enlace en el [anexo](#) de como funciona las claves)













2.2.3 Persistencia

En la aplicación se ha utilizado Firebase Authentication y Firebase Database (Cloud Firestore), ambas se han usado conjuntamente para conseguir el objetivo propuesto respecto al almacenamiento de datos de moodle y a los usuarios.

2.2.3.1 Authentication

Firebase Authentication ofrece varias formas de inicio de sesión, entre todas las opciones que muestra, en la aplicación se han utilizado por **correo electrónico** y por **google**.

Proveedores de inicio de sesión

Proveedor	Estado
 Correo electrónico/contraseña	Habilitada
 Teléfono	Inhabilitado
 Google	Habilitada
 Play Juegos	Inhabilitado
 Game Center Beta	Inhabilitado
 Facebook	Inhabilitado
 Twitter	Inhabilitado
 GitHub	Inhabilitado
 Yahoo	Inhabilitado
 Microsoft	Inhabilitado
 Apple	Inhabilitado
 Anónimo	Inhabilitado

Además en el Authentication cada vez que se crea una cuenta desde la aplicación o desde la misma Consola de Firebase.

Buscar por dirección de correo electrónico, número de teléfono o UID de usuario Añadir usuario ↻ ⋮

Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
prueba@gial.com		16 may. 2020	16 may. 2020	8r4EFJA7RYgGqLtw1oumkOKNIKN2
angelicod77@gmail.com		16 may. 2020	22 may. 2020	Fqk82NnMzpOyyjAVMxSzVoiG9TI2
apereiraf73@elpuig.xeill.net		15 may. 2020	21 may. 2020	coMkt0OfklUppyprnDZlGY8SO0it2
a@a.com		18 may. 2020	21 may. 2020	i64lqJ9UkAhoxc50vi2DOyr9S8c2
admin@admin.com		15 may. 2020	16 may. 2020	v0nGpcAxbwUns4PnBfNkvjVIKFq1

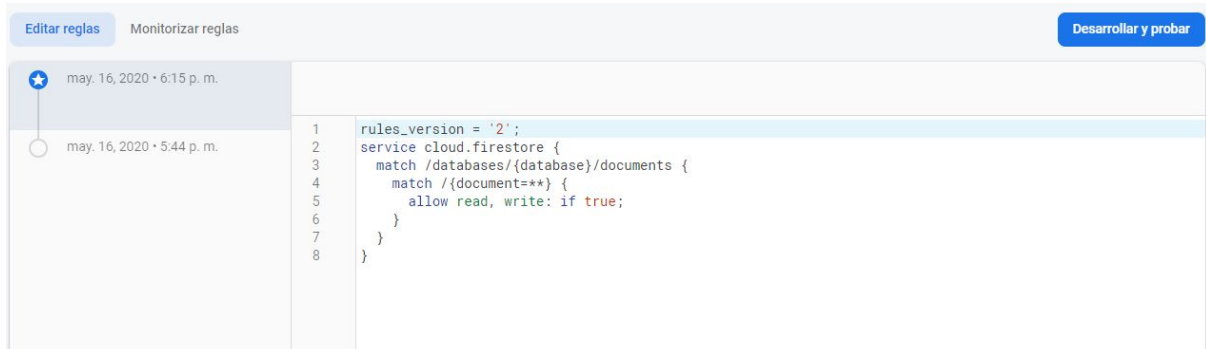
Filas por página: 50 ▼ 1-5 de 5 < >

Como se puede apreciar en la columna derecha aparece el **UID de usuario** que gracias a este UID se podrá relacionar con los tokens obtenidos de la api pudiendo iniciar sesión en moodle sin necesidad de que se solicite este token al moodle en cada inicio de sesión, sino que se guardará en el **Cloud Firestore** para utilizarlo en funcionalidades como Favoritos.

2.2.3.2 Database

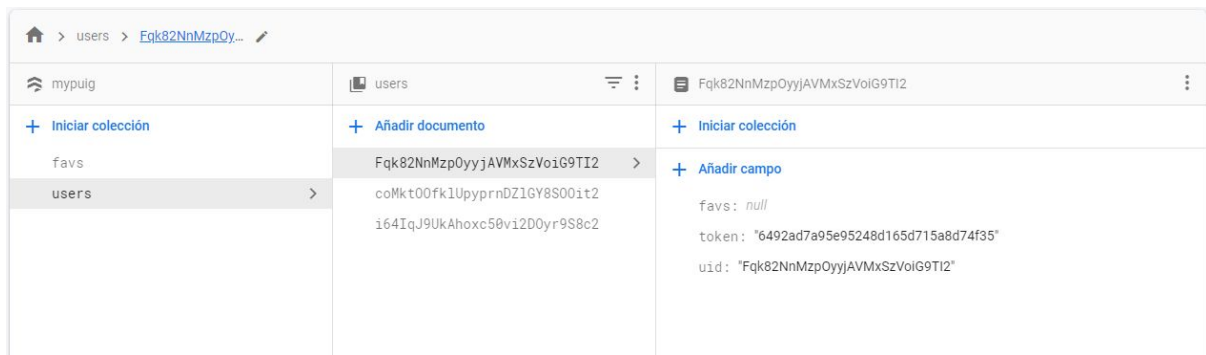
Cloud Firestore es una base de datos NOSQL orientada a los documentos similares a JSON. A diferencia de una base de datos SQL, no hay tablas ni filas. En su lugar, almacenar los datos en documentos, que se organizan en colecciones.

Para configurar la base de datos se necesita especificar y seleccionar las opciones que ofrece, además para que la base de datos funcione hay que cambiar un valor en las reglas, en la línea 5 el **false** debe cambiarse a **true**.

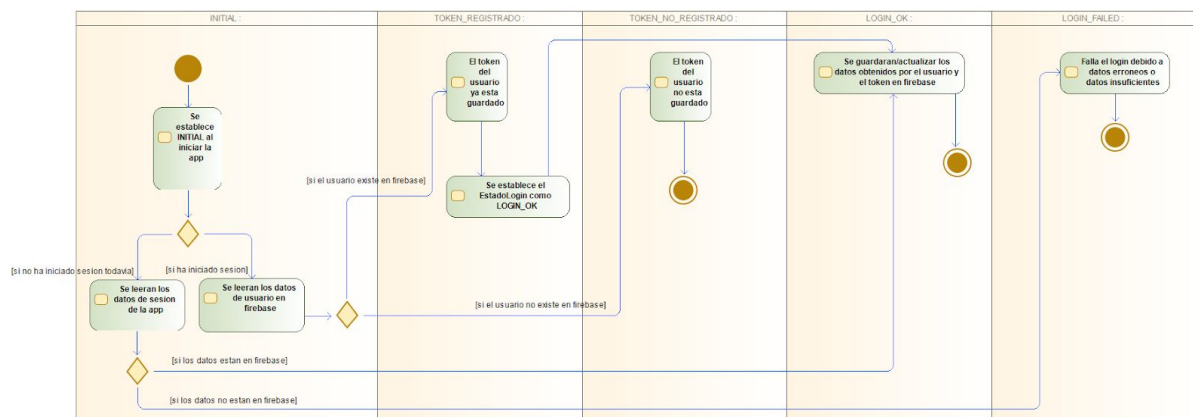
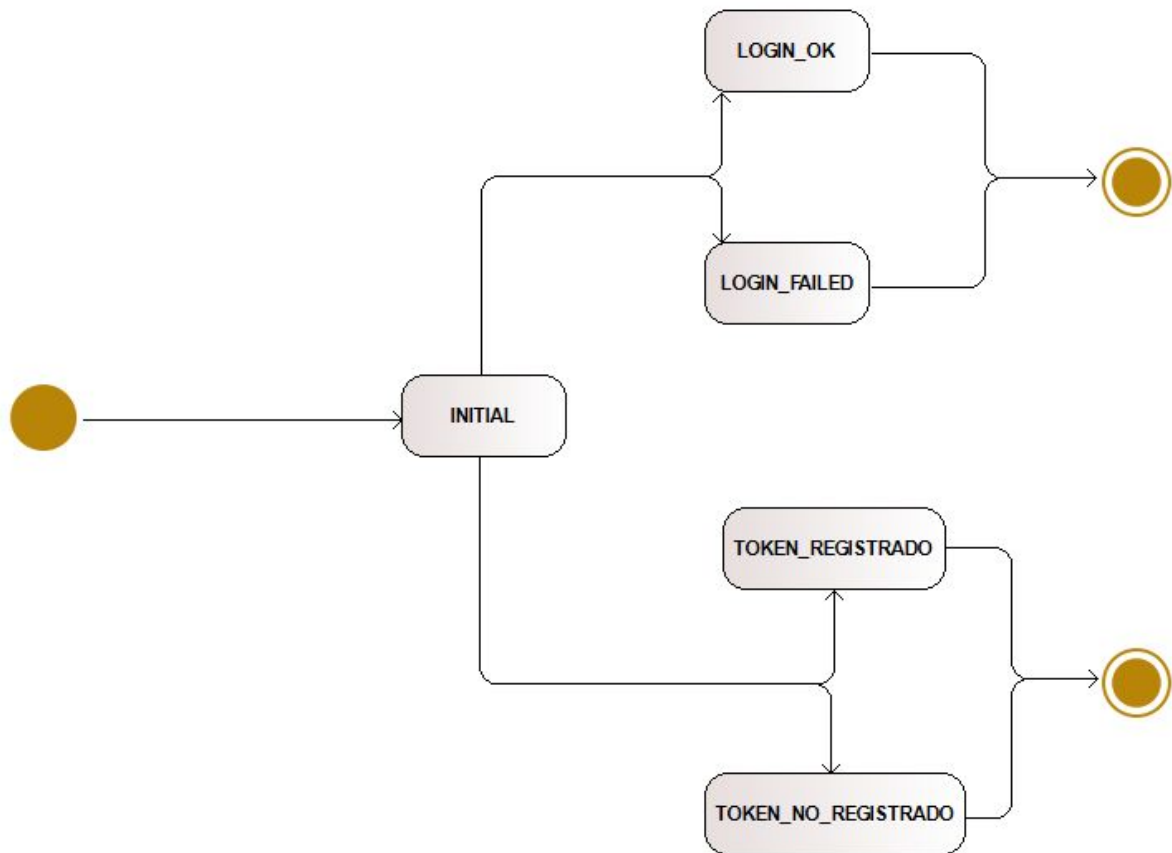


```
1 rules_version = '2';
2 service cloud.firestore {
3   match /databases/{database}/documents {
4     match /{document=**} {
5       allow read, write: if true;
6     }
7   }
8 }
```

Una collection se compone de documentos, cada uno de ellos contiene unos campos o más colecciones(subcolecciones) y esto llevaría tener más documentos.



Anteriormente se ha mencionado los tokens y el uid, bien pues como se puede observar existe una collection users que contendrá una serie de usuarios o que es lo mismo documentos, y dentro el token obtenido mediante el primer inicio de sesión con moodle y el uid de la cuenta usada con el Firebase Authentication.



Como se puede observar en el siguiente código, esta es la forma en la que se guarda el token y el UID, mediante el **currentUser** (usuario logueado con Authentication) se puede obtener los datos del usuario actual que ha iniciado sesión, guardando así el UID y el token de moodle obtenido de la api. Con el *db.collection* se creará la collection users y con el *.document* se especifica el nombre de ese documento, que será el UID del usuario y son el set se guarda el usuario creado anteriormente.

```

public void addDataUser(){
    currentUser = FirebaseAuth.getInstance().getCurrentUser();

    // Create a new userFavs with a first and last name
    User user = new User(currentUser.getId(), token.getValue());

    Add a new document with a generated ID
    db.collection( collectionPath: "users" ) CollectionReference
        .document(currentUser.getId()) DocumentReference
        .set(user) Task<Void>
        .addOnSuccessListener(aVoid -> Log.w(TAG, msg: "TOKEN ->" + user.token)) Task<Void>
        .addOnFailureListener(e -> Log.w(TAG, msg: "Error adding document", e));
}

```

2.2.4 Interfície

2.2.4.1 Wireframe

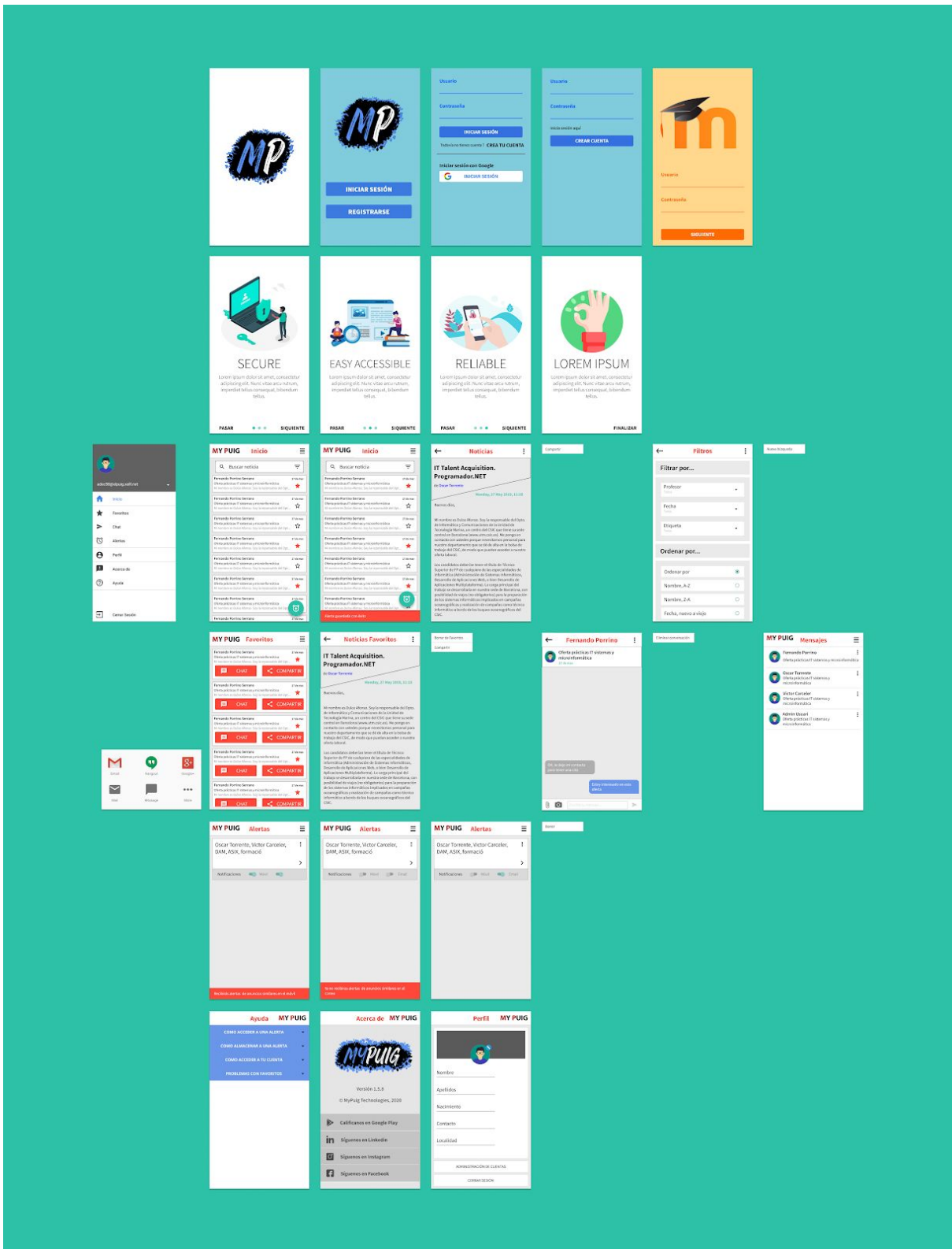
La herramienta utilizada para la creación del mockup es Figma. Una aplicación web en la que simplemente tienes que registrarte y ya la puedes utilizar de manera gratuita, trabaja por frames en los que se pueden representar las diferentes pantallas de la aplicación.

Es muy interesante ya que cuenta con herramientas como componentes y vectores además con una aplicación externa a la aplicación web llamada figma mirror. En este mirror te permite hacer selección de un frame que quieras ver y te muestra en el teléfono de manera exacta, teniendo en cuenta colores y dimensiones.

Además figma cuenta con una funcionalidad muy interesante de wireframes en el que es más fácil mostrar las diferentes funciones que lo componen, información que será mostrada, la relación de las diferentes pantallas a través de escenarios y que reglas utiliza la aplicación para mostrarlos.

En el diseño, se ha utilizado colores más orientados a la búsqueda de trabajo, por lo tanto colores coloridos no se utilizan y hay colores corporativos debido al público al que va orientado, por ello el objetivo es utilizar colores como por ejemplo el azul, gris, blanco. En cuanto a dimensiones se han utilizado las de un teléfono con una medida estándar, teniendo cuenta que todo el mundo utiliza un teléfono de 5 pulgadas y además se ha optado por un diseño únicamente en vertical y responsive, se ha tomado esta decisión debido a la poca utilización del teléfono en horizontal y a la poca comodidad que comporta, por ello se ha decidido de tener una única distribución de pantalla.

(Enlace al [wireframe](#))



2.2.4.2 Interacciones

Tap/Pulsar. Esta interacción está prácticamente en todo las pantallas, todo lo que sean iconos en los que se puedan interactuar y botones se podrán hacer un tap. En la sección de registrarse, a la hora de poner la contraseña y el usuario de los dos

registros. En el inicio se puede interactuar con a barra de búsqueda, el botón de favoritos, el botón de añadir una alerta, el de filtros y el tap al querer visualizar la noticia, una vez dentro de la noticia podemos hacer dos cosas, volver atrás o darle al icono de los dos puntos en los que nos saldrá una pequeña pestaña en la que nos permitirá compartir la noticia.

Una vez dentro en favoritos podremos hacer lo mismo que en la de inicio con una pequeña variación en la que nos saldrán dos botones en cada una de las noticias en la que nos permitirá ponernos en contacto con la persona que ha añadido esa noticia y compartirla de una manera más fácil.

Y en alertas se podrá ver los criterios de búsqueda almacenados por el usuario y se podrán hacer tres cosas, controlar las notificaciones con dos botones switch, acceder a noticias con esos criterios de búsqueda guardados y al interactuar con el botón de los tres puntos aparecerá una pequeña pestaña en la que nos permitirá borrar esa alerta. En el Perfil para poder cambiar la foto de perfil, los datos personales o para administrar las cuentas y cerrar sesión se tiene que hacer tap.



Swipe/Pasar. En la funcionalidad del Onboarding en el que se visualiza una pequeña imagen con un texto y una descripción debajo se puede hacer swipe hacia la izquierda o la derecha dependiendo de si el usuario quiere llegar hasta el final, para poder comenzar a utilizar la aplicación o si quiere volver a la pantalla anterior.

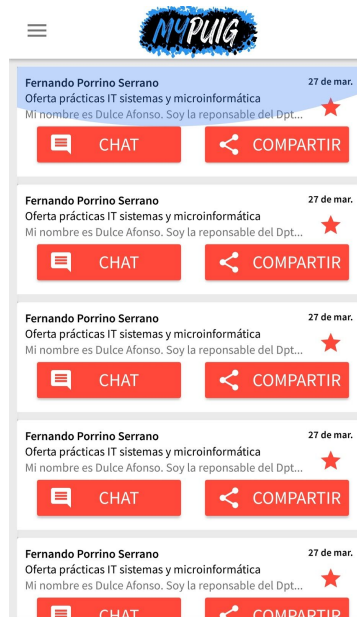


SECURE

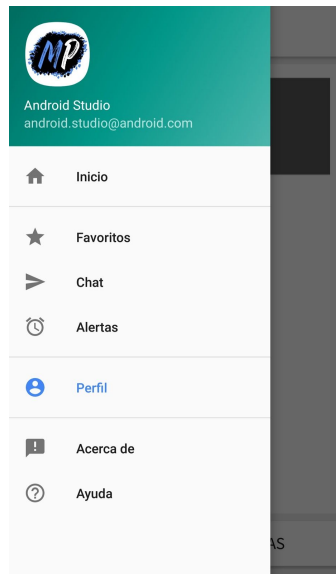
Lorem ipsum dolor
sit amet, consectetur
adipiscing elit. Nunc vitae
arcu rutrum, imperdiet
tellus consequat, bibendum
tellus.



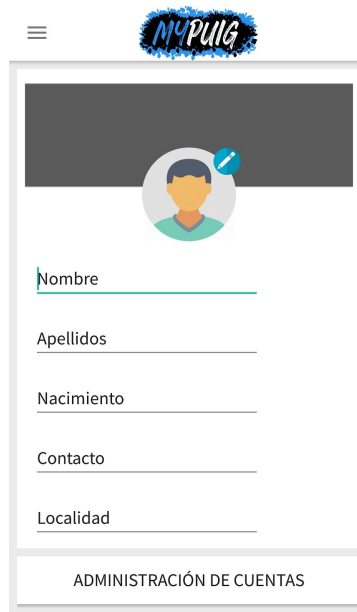
Scroll/Desplazar. Todo lo que sean listas de noticias, búsquedas, etc, se basará en modelo de contenedor RecyclerView. Esto significa que cada uno de ellos utilizarán un scroll vertical en secciones como la lista de noticias (Inicio), la lista de noticias marcadas como favoritos (Favoritos), los criterios de búsqueda guardados por el usuario (Alertas). La única pantalla que tendrá un scroll y no es un recyclerview es la de ayuda, en la que se visualizará diferentes secciones y subsecciones en las que se mostrarán las preguntas más usuales generadas los usuarios.



Drag/Arrastrar. Al hacer tap en el menú podemos observar cómo aparece un menú de izquierda a derecha, al tocar fuera desaparece pero al hacer drag de derecha a izquierda este también desaparece para seguir visualizando la pantalla en la que se haya abierto o hayamos interactuado con el menú.



Escribir. Para introducir los datos de registro del usuario o para iniciar sesión, ya sea poniendo el nombre de usuario o la contraseña, se tiene que introducir datos. Además en el inicio cuenta con una barra de búsqueda en la que se puede buscar por palabras todas los títulos relacionados con esa palabra introducida con texto en el buscador. Y por último la gestión de cuentas, en lugares como la edición de perfil se puede escribir para editar los datos personales del usuario.

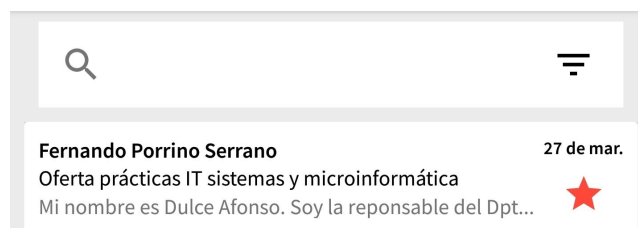


2.2.4.3 Funcionalidades

- Favoritos

En la sección de favoritos se pueden visualizar todas las noticias que el usuario ha decidido añadir dependiendo de su interés. El tener las noticias favoritas en una sección diferente añade al usuario una comodidad adicional, estas noticias se quedan guardadas y por lo tanto hace que no solo sea una simple sección de favoritos, sino que también que funcione como un leer más tarde. El usuario guarda algo y en el caso de ser de su interés guardarlo o en su efecto eliminarse.

Esta opción también permite que las noticias de interes puedas compartirlas de una forma mucho más fácil y además puedas contactar con la persona que haya colgado la noticia en caso de alguna duda o problema.

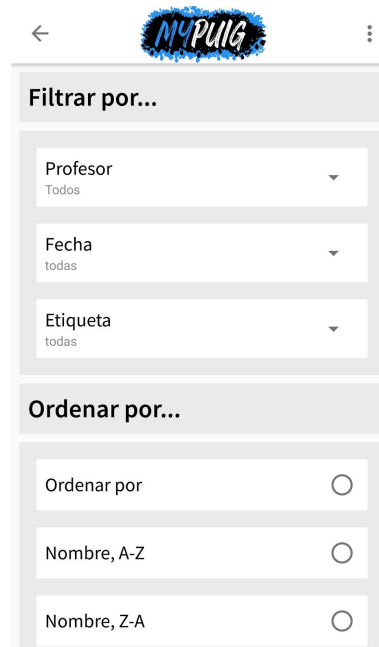


- Filtrar

En esta funcionalidad tenemos la oportunidad de dos cosas, filtrar y además ordenar por, el interés de esta sección es que en el momento que el usuario accede a los filtros y hace uso de ellos un botón se activa en inicio, proporcionando así al usuario la oportunidad de guardar unos criterios de búsqueda. Lo interesante es que además de poder guardar estos criterios de búsqueda, el usuario puede permitir si

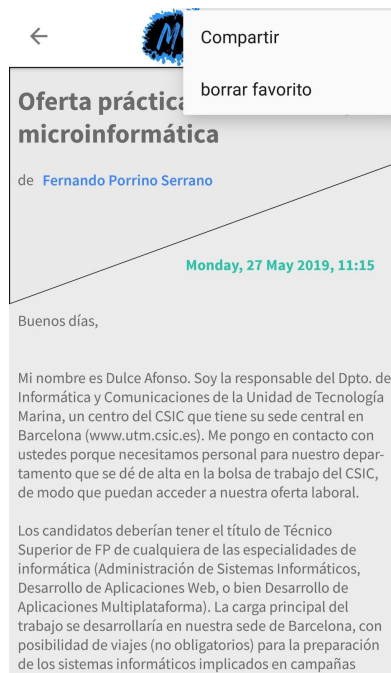
quiere que le lleguen notificaciones o no de noticias nuevas con los criterios guardados.

En cuanto a filtrar, el objetivo es que el usuario pueda hacerlo sobre diferentes temas relacionados con el curso, como por ejemplo el profesor que las ha colgado, las etiquetas que tienen esa entrada al foro y la fecha en la que se hizo esa entrada.



- **Compartir**

En el objetivo de esta funcionalidad es que dependiendo de si el usuario tiene la noticia el favoritos o simplemente visualizando la noticia, la facilidad de compartir la noticia varía. Esto tiene un porque, cuando guardas una noticia ya tienes el botón de compartir (1 tap), sin embargo visualizando la noticia se tendría que dar al botón y dar a compartir (3 taps), el objetivo de esto es que el usuario guarde las noticias que le interesan y las comparta de manera más sencilla.



- Ayuda

En esta sección se podrán ver una serie de subapartados en las que aparecerán las diferentes funcionalidades de la aplicación y en forma de desplegable saldrán una serie de preguntas a las que serán resueltas. El objetivo de estas preguntas es proporcionar una ayuda mucho más clara, en el caso de que el usuario tenga algún tipo de duda.

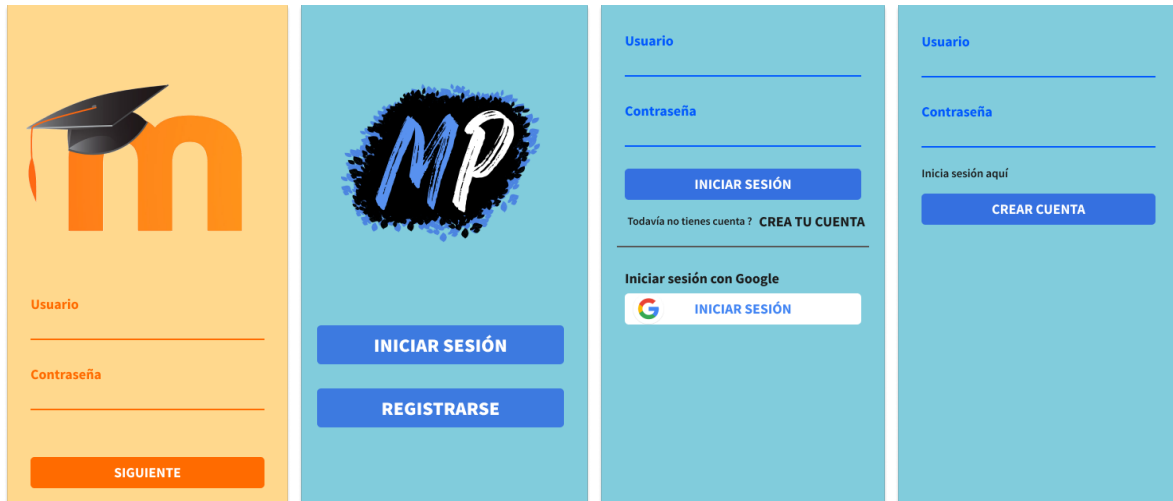
Además esta aplicación cuenta con Onboarding en el que antes de utilizar la aplicación, hace un pequeño tour en el que con unas pequeñas imágenes y descripciones dan una vista previa de cómo será nuestra app.

- Página de registro

Esta página de registro contará como con una doble funcionalidad. La primera consistirá en registrarse con un usuario de Moodle y dependiendo de si ha iniciado satisfactoriamente la sesión, en caso de no hacerlo te redirigirá a la página web de Moodle (en el caso de olvidar la contraseña, etc.). Al terminar de iniciar sesión de la primera, pasar a otra página de registro que irá relacionada con la aplicación de teléfono en la que se podrá registrarse o iniciar sesión, dependiendo de la necesidad del usuario.

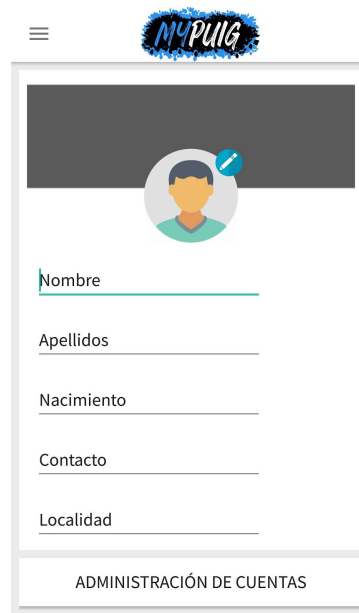
El objetivo de las dos páginas de registros es que el usuario que se registra con su cuenta de Moodle, esté inscrito en el curso del que se sacaran las noticias del foro, el objetivo de esta página de registro es que el usuario se tenga que registrar una vez y no tenga que volver a acceder más. La otra funcionará más bien como la

página principal en la que un usuario se puede registrar para comenzar a utilizar la aplicación.



- Gestión de cuentas

En el perfil se pueden hacer 4 cosas, editar tu información personal, cambiar tu foto de perfil, administrar las cuentas con las que se han iniciado sesión y cerrar sesión. También desde el menú se puede acceder a las diferentes cuentas que tenemos asociadas a la aplicación y en el caso de que cerrar sesión, volver a la segunda página de registro.



2.2.5 Tecnología

La tecnología utilizada en este proyecto ha sido Android debido a que es mucho más fácil controlar los Patrones de UI y al no disponer de un iOS, todo esto es porque en elementos como menús, listas, encabezados y un elemento muy importante en la estructura de la aplicación que es el toolbar superior en el que

navegamos por las diferentes pantallas en iOS cambiaría de forma completa ya que las flechas se representan de una forma diferente y la navegación funciona de otra manera.

En cuanto a utilizar una app móvil, esta decisión es porque al tomar la decisión de hacer una app web nos quitaría de varias ventajas que la app móvil, una de ellas es trabajar con los IDE con los que más experiencia tenemos y otro el no poder acceder recursos del teléfono como la cámara, gps, etc, si que es verdad que al principio si que es una idea bastante atractiva ya que al desarrollar una web app nos permitiría utilizar un consumo más bajo de recursos a comparación de una app.

Al final se ha decido utilizar una aplicación debido a una mayor experiencia del equipo de desarrollo en los IDE específicos y a la facilidad de tener en cuenta un único patrón de UI, por ello la tecnología implicada en el proyecto ha sido dividida en dos bloques:

- La tecnología utilizada como soporte han sido AWS, para la instalación del Moodle y poder acceder de forma remota a el; Github integrado dentro de IntelliJ y Android Studio, para compartir código; Figma para el diseño de la interfaz de la aplicación y la utilización de IntelliJ junto con Android Studio para el desarrollo de la aplicación, en cuanto a detalles de desarrollo se ven especificados en el punto [2.3 Desarrollo](#).
- La tecnología que se ha desarrollado ha sido en Java, todo es porque queríamos hacer una aplicación para Android y esto suponía utilizar un lenguaje de programación en específico, en este caso se ha decido Java porque es multiplataforma y gracias a la gran cantidad de contenido Open Source que hay en plataformas como Github, etc.
Y por último una de las grandes razones por la que se ha elegido es por su orientación a objetos, esto permite al desarrollador adquirir características como la facilidad del desarrollo, reutilización del código y mayor calidad del código.

2.3 Desarrollo

Todas las tareas relacionadas con el desarrollo serán llevadas a cabo sobre un servidor en AWS. Esta plataforma uno de los servicios que proporciona es el de Ec2 un servicio web en el que simularemos el ambiente del Moodle del Puig, en el que añadiremos un curso y varias entradas a un foro creado en ese mismo curso, además crearemos diferentes usuarios para realizar diferentes pruebas.

Utilizando esta herramienta evitamos diferentes problemas, uno de ellos el tener que ir cambiando de ip, AWS proporciona una funcionalidad muy interesante llamada elastic IP, semejante a las IP estáticas.

Otro de los problemas sería el trabajo en grupo y por lo tanto el tener que utilizar máquinas virtuales, esto podría generar diferentes versiones de la máquina virtual y por lo tanto varias incompatibilidades entre los usuarios creados en el Moodle y relacionados con el token que cada uno de ellos genera, todo esto nos ayudará a agilizar el proceso del desarrollo de la aplicación y por lo tanto optimizando de una forma más efectiva nuestro tiempo.

AWS es una plataforma con diferentes servicios, entre ellos nube, base de datos, mensajería, servicios móviles, inteligencia artificial, etc. El servicio que más se ha utilizado en el proyecto es el de EC2 (Amazon Elastic Compute Cloud), este trata sobre un servicio web que proporciona la oportunidad de crear diferentes instancias en las que poder ejecutar aplicaciones en la infraestructura de Amazon Web Services, antes de la instalación proporciona una serie de componentes útiles, entre ellos las Elastic Ip una ipv4 asociada a la cuenta del AWS, a la que se puede acceder desde cualquier lugar con internet, otro de los aspectos que se deben tener en cuenta son los Security Groups, sirve como firewall virtual en el que controlar el tráfico entrante y saliente, este concepto es algo que tener en cuenta para poder utilizar cosas como el SSH o poder acceder al moodle ya sea por HTTP o HTTPS. También hemos estado utilizando el entorno de desarrollo para aplicaciones Android Studio, basado en IntelliJ IDEA y gratuito. Este viene con funciones muy interesantes y útiles en las que facilitan el desarrollo y progreso de la aplicación, como por ejemplo:

- Soporte para Gradle
- Refactorización
- Soporte para Google Cloud Platform, en los que permite integrar Firebase (entre otros).
- Renderizado a tiempo real.
- Consejos de optimización y ayuda para la traducción.
- Un dispositivo virtual de Android en el que ejecutar y probar aplicaciones.

Además, para realizar pruebas se ha estado utilizando el IDE de IntelliJ IDEA antes de implementar las diferentes pruebas en el proyecto con Android Studio.

Así pues, durante la fase de desarrollo del proyecto, según ha ido evolucionando y por lo tanto se ha ido consiguiendo resultados hemos procedido a subir el código a Github. Se ha decidido utilizar esta herramienta, ya que es la que más familiar le

parece al equipo de desarrollo y porque los dos IDE que se han utilizado durante el proyecto tienen integrados por defecto, este sistema de control.

2.3.1 API REST de Moodle

Una API REST consiste en una interfaz para conectar diferentes sistemas basados en el protocolo HTTP pudiendo de esta manera generar y obtener datos, devolviendo esos datos en formatos como por ejemplo JSON y XML.

Para ello se ha utilizado la herramienta de terminal curl, esta permite transferir datos desde o hacia un servidor utilizando una variedad muy amplia de protocolos, entre ellos HTTP o HTTPS. (Enlace en el [anexo](#))

Moodle es una plataforma web para la gestión de cursos, en el se permite gestionar diferentes elementos del ámbito como son alumnos o profesores, una de sus funciones no es sólo gestionar cursos y todo lo relacionado con ello, sino que también crear foros.

Además este proporciona una serie de servicios, entre ellos los de API REST. Para ello debemos tener en cuenta una serie de funciones y estructuras según el uso que le quiera dar. Su estructura está compuesta por:

- wsfuction = el método que se va utilizar.
- fullcomponent = es como moodle identifica los diferentes componentes por ejemplo mod_forum.
- methodname = se podría decir que muchos methodname forman un fullcomponent y cada uno de estos tienen funcionalidades diferentes pero relacionadas por el mismo componente, por ejemplo de mod_forum tiene mod_forum_get_forums_by_courses, mod_forum_can_add_discussion, etc.
- verb = @GET, @CREATE, @DELETE, @UPDATE
- noun = objetos relacionados con moodle como por ejemplo usuarios, cursos, etc.

```
curl
"https://moodle.elpuig.xeill.net/webservice/rest/server.php?wstoken=xxxxx&moodlewsrest
format=json&forumid=xxxxx&wsfunction=mod_forum_get_forum_discussions_pagina
ted"
nouns= wstoken, forumid
wsfunction= mod_forum_get_forum_discussions_paginated
*moodlewsrestformat= sirve para especificar el formato de visualización de datos
```

Además para poder utilizar la API REST debemos disponer de un token, este es generado por el usuario una vez ha iniciado sesión con su nombre y contraseña, pero para que esto funcione se deben de activar varias funcionalidades en Moodle para que este token se genere, sin él sería imposible trabajar con la API.

Además para utilizar y almacenar la respuesta que hacemos con el @GET debemos utilizar unas clases en específico y unos nombres de variables exactos. Por ello se debe comprender el formato de la respuesta y cómo la vamos a almacenar:

```
@GET("webservice/rest/server.php?moodlewsrestformat=json&wsfunction=mod_forum_get_forum_discussions_paginated")
Call<Discussions> discussions(@Query("wstoken") String wstoken, @Query("forumid") String forumid);
```

Como se puede ver en esta consulta se intenta sacar con la función mod_forum_get_forum_discussions_paginated, esta ya nos da una pista de cómo será la respuesta. Para poder realizarla se está utilizando el token (wstoken) y el id del foro (forumid).

```
public class Discussion {
    public String id;
    public String name;
    public String message;
    public String userpictureurl;
    public String userfullname;
    public long created;
    public long timemodified;
    public boolean isFav;
}

public class Discussions {
    public List<Discussion> discussions;

    @Override
    public String toString() {
        return "Discussions{" +
            "discussions=" + discussions +
            '}';
    }
}
```

Como se puede ver la respuesta que nos da esta consulta es una lista de discusiones compuestas por variables idénticas a la respuesta, es decir si la respuesta o internamente en el moodle hay una variable llamada message (que hace referencia al contenido de un post) esa variable declarada en nuestro código Java debe llamarse igual.

```
public LiveData<List<Discussion>> getNoticias = Transformations.switchMap(token, new Function<String, LiveData<List<Discussion>>>() {
    @Override
    public LiveData<List<Discussion>> apply(String input) {
        api.discussions(input, Connection.getForumid()).enqueue(new Callback<Discussions>() {
            @Override
            public void onResponse(Call<Discussions> call, Response<Discussions> response) {
                Log.e("ABC", response.message());
                if(response.body() != null){
                    for(Discussion d: response.body().discussions) Log.e("ABC", d.toString());
                    listaNoticias.postValue(response.body().discussions);
                } else {
                    Log.e("ABC", response.message());
                }
            }

            @Override
            public void onFailure(Call<Discussions> call, Throwable t) {
                Log.e("ABC", "msg: "getDiscussions connection failed");
            }
        });
        return listaNoticias;
    }
});
```

Con este método se realiza el almacenamiento de la respuesta que se ha hecho en la conexión con la API y por lo tanto guardamos esa lista de discusiones y dos métodos para controlar si la conexión se ha realizado de forma efectiva.

Con el Transformation lo que se hace es observar un Live Data para poder modificar otro, la finalidad de esto es debido a que al hacer el connection hace primero el login y no el sacar la lista de discussions, esto lo que provoca es que una vez hayamos abierto la aplicación no se visualicen las noticias.

(Enlace en el [Anexo 1](#), [Anexo 2](#), [Anexo 3](#))

2.4 Pruebas

Al terminar esa primera fase de diseño en la que ya poder tener algo en lo que se pueda presentar, una de las primeras fases es decidir si todo lo claro que le parece al equipo de desarrollo elementos como iconos, colores de la aplicación o la distribución de los diferentes elementos es adecuada.

Por ello se ha decido realizar una prueba de usabilidad en la que se miden diferentes métricas y atributos para tener en cuenta modificaciones y mejoras de cara a app.

Durante y después de la fase de desarrollo de la aplicación se han realizado una serie de pruebas para medir diferentes parámetros, en los que se puede facilitar y mejorar el trabajo realizado por los desarrolladores durante la realización del proyecto.

2.4.1 Usabilidad

El objetivo de este test es medir diferentes métricas y atributos a lo largo de la prueba, como por ejemplo efectividad, satisfacción, eficiencia o números de errores realizados al interactuar con algo. Entre esas pruebas quiero ver a cuantas pantallas es capaz de interactuar el usuario en un determinado tiempo y que el usuario vaya generando feedback según realiza las diferentes navegaciones por las pantallas, también en cuanto a resultado final me gustaría ver una curva de aprendizaje de cara al usuario, para ver si funcionalidades como la de Ayuda y Onboarding tienen la efectividad esperada.

La conclusión de este test, es que en cuanto a aspectos visuales ha gustado de manera funcional y visual. En cuanto a detalles mejorables en la aplicación serían, algunos colores en algunas secciones, hacer más visible el viewholder de la funcionalidad de Favoritos, hacer más intuitivo el agregar diferentes alertas. En cuanto a efectividad de los objetivos propuestos, se puede decir que ha sido bastante notable ya que en el tiempo esperado se han realizado todas las tareas con la mayor efectividad posible.

Pero se considera volver a hacer un test de usabilidad, para medir si estos detalles o tareas que no hemos podido cumplir una vez la aplicación sea más funcional, en

cuanto aspectos de programación y sea más intuitiva. Por ello pensamos que volver hacer un test de usabilidad con la aplicación más avanzada acabaría de pulir esos detalles para ver si es lo suficientemente usable.

(Enlace en el [anexo](#))

(Enlace del [test](#))

2.5 Lanzamiento

2.5.1 Tareas para la distribución

Una vez terminadas las pruebas de Usabilidad y haber pulido todos los aspectos que no resultaban satisfactorios por parte de los Testers. Una vez terminada esa primera fase de Test en la que ir probando todos los puntos débiles y mejorarlos, seguidamente el objetivo es comenzar con la distribución sin tener todas las funcionalidades del proyecto disponibles, solo las de la planificación inicial como mínimo.

Esto hará que en el caso de tener éxito con la app poder seguir desarrollando más y más funcionalidades para ir incrementando el público y la gente que utilizara la aplicación, además realizar pequeñas difusiones en los centros educativos para que cada vez más institutos estén interesados en fomentar la búsqueda de trabajo por parte de los alumnos de Formación Personal. Por ello deberán mantenerse tanto los servidores de alojamiento web y del Firebase Database para el funcionamiento correcto de la app.

En el caso de superar las expectativas exponencialmente se desarrollaría a partir de la misma app nuevos módulos en las que no solo los alumnos de Formación Personal puedan utilizarla, sino que dividir la aplicación en diferentes categorías según las necesidades del usuario final y sus intereses haciendo la aplicación más completa y abarcando a un público y número de personas mucho más grande.

2.5.2 Publicación

Para la publicación de una app o aplicación a Play Store se tiene que tener una serie de requisitos antes de la publicación. Por ello se tiene que crear una cuenta de desarrollador en Google Play para poder utilizar el Play Console.

Acto seguido al tener el Console se tiene que crear la aplicación y especificar una serie de parámetros de la aplicación como por ejemplo la clasificación de contenido o configurar los precios y la distribución.

Una vez creada la aplicación tenemos que tener en cuenta que nuestra aplicación aún no está publicada, por ello se debe saber que Play Console gestiona las versiones de las aplicaciones según su estado:

- **Borrador:** el archivo APK aun no se ha publicado para los usuarios.
- **Activo:** archivos APK están publicándose.
- **Archivado:** los archivos APK que estaban activos han dejado de publicarse para los usuarios.

Teniendo en cuenta estas versiones además debemos tener en cuenta que disponemos de un tamaño limitado, esto significa que dependiendo de la versión de la app y la API podremos subir un límite de MB de nuestra APK.

Una vez hemos terminado debemos tener en cuenta que debemos firmar nuestra APK y que está APK deberá tener una serie de prerequisites mínimos.

Y para finalizar simplemente configurar la ficha de Play Store en la que irán parámetros como detalles del producto, recursos, idiomas, traducciones, categorías, contacto, política de privacidad, etc.

2.6 Conclusions

La verdad que este trabajo nos ha hecho aprender muchas cosas, entre ellas a utilizar una API REST, las ventajas y desventajas que conlleva utilizarla y a utilizar Firebase de una forma mucho más completa, hasta llegar al punto de plantearnos lo poco a que nos sabe Firestore y todo los problemas que nos ha llevado almacenar parámetros como los checkbox de los favoritos y que se pierda la información de estos en la aplicación, la dificultad que conlleva comprender la manera en la que se guardan los datos y cómo hacerlo.

Además de que ha sido un gran reto de realizar este proyecto en un periodo de tiempo escaso entre el proceso de investigación entre utilizar los diferentes componentes y el desarrollo, pero creemos que puede llegar a ser beneficioso tanto para nosotros como para los centros educativos que utilizan moodle y estén dispuestos a utilizar esta app, ya que es algo nuevo que moodle por defecto no ofrece y es un plus que podríamos ofrecer.

Los objetivos principales si que se han conseguido de la manera propuesta, sí que es verdad que hay en funcionalidades como en favoritos tal vez nos ha supuesto un grado de dificultad mayor que el de aprender a utilizar la API REST, por ello nos ha supuesto tener que utilizar más tiempo de lo debido y dejar a medio camino funcionalidades como Filtros, Chat, Alertas.

Nuestra planificación inicial era hacer los filtros, favoritos y api rest; el suponer que API REST era el gran tema principal nos ha supuesto un gran trabajo de investigación ya que desde el principio se presentaban bastantes problemas de

ASIX, entre ellos el configurar el Moodle y poner el marcha el Web Services, los dominios y el certificado.

Por ello se ha decidido que funcionalidades y detalles que se quedan a medio hacer o simplemente no están hechos, partiendo de la base que hay unas funcionalidades principales que son Favoritos y la visualización de los foros comenzar con el Lanzamiento. Esto significa que hay una visión de futuro la aplicación seguirá teniendo soporte y desarrollándose para ir mejorando y desarrollando funcionalidades como el Chat entre ellas.

La verdad que nos hubiera gustado desarrollar de manera funcional una o dos funcionalidades más de la aplicación. Además poder extraer datos y profundizar mucho más en el mod_forum, por ello consideramos que poder trabajar en un tema tan interesante que es API de Moodle nos ha hecho aprender aspectos en general de las API y la dificultad que conlleva utilizarlas por las respuestas poco claras que devuelven las consultas.

2.7 Webgrafia

Adictos al trabajo, Autentia, Bases de Datos de Firebase con Android, Disponible en: <https://www.adictosaltrabajo.com/2019/03/19/bases-de-datos-de-firebase-con-android/>

Juanda, Gitbooks.io, Arquitectura y funcionamiento de una API REST, Disponible en: <https://juanda.gitbooks.io/webapps/content/api/arquitectura-api-rest.html>

Moodle, Moodle, Web service API functions and core web service functions, Disponible en: https://docs.moodle.org/dev/Web_service_API_functions

Moodle, Moodle, Creating a web service client, Disponible en: https://docs.moodle.org/dev/Creating_a_web_service_client

Vital Innova, eBusiness Growth, Servicios web en Moodle 2.0, Disponible en: <https://www.vitalinnova.com/servicios-web-en-moodle-2-0/>

Gerard Falco, IES Puig Castellar, AutoTest, Disponible en: <https://github.com/gerardfp/AutoTest>

Firebase, Firebase, Firebase Console, Disponible en: <https://console.firebase.google.com/u/0/?hl=ca>

Firebase, Firebase, Firebase Security Rules, Disponible en:

<https://firebase.google.com/docs/rules>

Firebase, Firebase, Get started with Cloud Firestore Security Rules, Disponible en:

<https://firebase.google.com/docs/firestore/security/get-started>

Firebase, Firebase, Understand Firebase Security Rules for Cloud Storage,

Disponible en: <https://firebase.google.com/docs/storage/security>

Moodle, Damyon Wiese, Mod_forum_get_forum(), Disponible en:

<https://tracker.moodle.org/browse/MDL-37247>

Moodle, Damyon Wiese, Mod_forum_get_forum(), Disponible en:

<https://tracker.moodle.org/browse/MDL-37247?attachmentViewMode=list>

InfoJobs, Infojobs, Ofertas de trabajo y justificación del presupuesto, Disponible

en: <https://www.infojobs.net/cambrils/disenador-grafico/of-iefecb554b9447e84f7921c5c8e0aea?applicationOrigin=search-new%7Cversion~react&page=1&sortBy=RELEVANCE>

InfoJobs, Infojobs, Ofertas de trabajo y justificación del presupuesto, Disponible en:

<https://www.infojobs.net/madrid/analista-datos-junior-ingles-alto/of-if0a9abfd814f82b1a523b294918257?applicationOrigin=search-new%7Cversion~react&page=1&sortBy=RELEVANCE>

InfoJobs, Infojobs, Ofertas de trabajo y justificación del presupuesto, Disponible en:

<https://www.infojobs.net/sant-cugat-del-valles/programador-informatico-junior/of-ia10f0ecac4cc69a1b1b14ae648c58?applicationOrigin=search-new%7Cversion~react&page=1&sortBy=RELEVANCE>

Annexos

1. Activar Servicios Web Moodle

Para poder los servicios de API REST de Moodle debemos tener en cuenta una serie de requisitos y cosas que debemos de hacer para ello debemos tener una cuenta de administrador y dirigirnos a:

- Site administration > Plugins > Web services > Overview
- En esta pantalla se mostrarán todos los requisitos que tenemos que tener en cuenta para poner en marcha los Web Services.

Step	Status	Description
1. Enable web services	Yes	Web services must be enabled in Advanced features.
2. Enable protocols	rest	At least one protocol should be enabled. For security reasons, only protocols that are to be used should be enabled.
3. Create a specific user		A web services user is required to represent the system controlling Moodle.
4. Check user capability		The user should have appropriate capabilities according to the protocols used, for example webservice/rest:use, webservice/soap:use. To achieve this, create a web services role with protocol capabilities allowed and assign it to the web services user as a system role.
5. Select a service		A service is a set of web service functions. You will allow the user to access to a new service. On the Add service page check 'Enable' and 'Authorised users' options. Select 'No required capability'.
6. Add functions		Select required functions for the newly created service.
7. Select a specific user		Add the web services user as an authorised user.
8. Create a token for a user		Create a token for the web services user.
9. Enable developer documentation	Yes	Detailed web services documentation is available for enabled protocols.
10. Test the service		Simulate external access to the service using the web service test client. Use an enabled protocol with token authentication. WARNING: The functions that you test WILL BE EXECUTED, so be careful what you choose to test!

- El primer paso a seguir es activar los Web Services para poder utilizarlos.

Search results

Advanced features

Administration / Advanced features

Enable web services
[enablewebservices](#)

Default: No

Web services enable other systems to log in to this Moodle and perform operations. For extra security this feature should be disabled unless you are really using it.

Save changes

- En siguiente paso seria activar los protocolos de Web Services que se van a utilizar, por defecto todas opciones salen desactivadas.

Manage protocols

Active web service protocols

Protocol	Version	Enable	Settings
REST protocol	2018120300	<input checked="" type="checkbox"/>	
SOAP protocol	2018120300	<input type="checkbox"/>	
XML-RPC protocol	2018120300	<input type="checkbox"/>	

For security reasons, only protocols that are in use should be enabled.

Web services documentation Default: No

Enable auto-generation of web services documentation. A user can access to his own documentation on his security keys page [More details](#). It displays the documentation for the enabled protocols only.

[Save changes](#)

- Para poder a comenzar a utilizar la API REST se deben generar usuarios específicos con permisos necesarios para poder utilizar la API REST, por ello nos dirigimos a create specific user, lo creamos y una vez tengamos ese usuario saldrá con todos estos permisos:

My Puig

Dashboard / Site administration / Users / Accounts / Add a new user

Blocks editing on

General Expand all

Username

Choose an authentication method

Suspended account

Generate password and notify user

The password must have at least 8 characters, at least 1 digit(s), at least 1 lower case letter(s), at least 1 upper case letter(s), at least 1 non-alphanumeric character(s) such as *, -, or #

New password

Force password change

First name

Surname

Email address

Email display

City/town

Check permissions in System

Roles for user Administrador 1

- Authenticated user in System

Permissions for user Administrador 1

Filter [Clear](#)

Capability	Allowed
Block: Admin bookmarks	
Add a new admin bookmarks block to Dashboard	Yes
block/admin_bookmarks:myadinstance	
Block: Latest badges	
Add a new Latest badges block to Dashboard	Yes
block/badges:myadinstance	
Block: Calendar	
Add a new calendar block to Dashboard	Yes
block/calendar:month:myadinstance	
Block: Upcoming events	
Add a new upcoming events block to Dashboard	Yes
block/calendar_upcoming:myadinstance	
Block: Comments	
Add a new comments block to Dashboard	Yes
block/comments:myadinstance	
Block: Community finder	
Add a new community finder block to Dashboard	Yes
block/community:myadinstance	
Block: Courses	
Add a new courses block to Dashboard	Yes
block/course_list:myadinstance	
Block: Global search	
Add a new global search block to Dashboard	Yes

- Una vez creado el usuario debemos generar su token, en *Create token*:

Create token

Token

User ! No selection

Search ▼

Service ! Moodle mobile web service ▾

IP restriction

Valid until 26 ▾ May ▾ 2020 ▾ Enable

Save changes Cancel

There are required fields in this form marked ! .

- Una vez creado los token y los usuarios podemos administrar los token y los usuarios en Site administration > Plugins > Web Services > Manage tokens, además se visualizan todas las funcionalidades que tiene un usuario específico para utilizar la API a diferencia de uno normal:

Manage tokens

Any token can be deleted, though you can only view tokens that you created.

Token	First name / Surname	Service	IP restriction	Valid until	Creator	Operation
90e14267817c33afefcef1aab6a222d4	Administrador 1	Moodle mobile web service			Administrador 1	Delete
dee6fe9db13599d527d032f116e6574b	Admin admin	Moodle mobile web service			Administrador 1	Delete

Missing capabilities: moodle/badges:viewotherbadges, moodle/calendar:manageentries, moodle/calendar:manageownentries, moodle/calendar:managegroupentries, moodle/comment:view, moodle/competency:competencyview, moodle/competency:evidencedelete, moodle/competency:competencymanage, moodle/competency:competencygrade, moodle/competency:coursecompetencyview, moodle/competency:usercompetencyview, report/completion:view, moodle/category:viewhiddencategories, moodle/course:update, moodle/course:viewhiddencourses, moodle/course:view, moodle/user:viewdetails, moodle/user:viewwhiddendetails, moodle/course:useremail, moodle/user:update, moodle/site:accessallgroups, moodle/course:viewparticipants, moodle/course:managegroups, moodle/site:deletebannerimage, moodle/user:editnummaccanprofile, moodle/site:editmaccan

2. Certbot

Para poder utilizar la API REST uno de los requisitos mínimos es que el dominio en el que se está utilizando tenga un certificado SSL.

1. El primer paso es conectarse por SSH a nuestro servidor.
2. Añadir Certbot PPA a la lista de repositorios:

```
$ sudo apt-get update
$ sudo apt-get install software-properties-common
$ sudo add-apt-repository universe
$ sudo add-apt-repository ppa:certbot/certbot
$ sudo apt-get update
```

3. Instalar Certbot:

```
$ sudo apt-get install certbot python3-certbot-apache
```

4. Cómo ejecutar el Certbot para instalar los certificados:

```
$ sudo certbot --apache
```

3. Freenom

Freenom es una página web en la que se pueden registrar dominios tanto de forma gratuita como de pago, los dominios gratuitos se pueden obtener durante 12 meses, a partir de ese plazo se deberá pagar. Para poder agregar el certificado SSL a una web, es necesario que esta tenga un dominio al que asignarle dicho certificado, ya que si se intenta agregar dicho certificado a una dirección IP esta no tendrá validez y la API REST no podrá acceder.

1. Para registrar un nuevo dominio se debe introducir el dominio y se comprobará su disponibilidad.
2. Seleccionar el dominio junto a su TLS y se deberán seguir los pasos que solicita.



Get one of these domains. They are free!

mypui .tk			x
mypui .ml	• FREE	EUR 0. ⁰⁰	Get it now!
mypui .ga	• FREE	EUR 0. ⁰⁰	✓ Selected
mypui .cf	• FREE	EUR 0. ⁰⁰	Get it now!
mypui .gq	• FREE	EUR 0. ⁰⁰	Get it now!

3. Se administraría el dominio para poder redirigir la ip publica al dominio recientemente adquirido.

Domain	Registration Date	Expiry date	Status	Type
mypuig.tk	2020-05-15	2021-05-15	ACTIVE	Free
mypuig.ml	2020-05-15	2021-05-15	ACTIVE	Free
mypuig.ga	2020-05-15	2021-05-15	ACTIVE	Free
mypuig.gq	2020-05-15	2021-05-15	ACTIVE	Free
mypuig.cf	2020-05-15	2021-05-15	ACTIVE	Free

- Una vez dentro de la administración del dominio se debe acceder a la pestaña de **Manage Freenom DNS**.

- Y por ultimo lugar se deberá introducir la ip pública como se aprecia en la captura.

4. Curl ejemplos y posibles obstáculos

```
curl
"https://moodle.elpuig.xeill.net/webservice/rest/server.php?wstoken=b7648eb1d5e13e164bfa36fd0d0eb3e0&moodlewsrestformat=json&forumid=440&wsfunction=mod_forum_get_forum_discussions_paginated"
```

```
{
  "discussions": [
    {
      "id": 1608,
      "name": "Oferta pr\u00e1cticas IT sistemas y microinform\u00e1tica",
      "groupid": -1,
      "timemodified": 1583362964,
      "usermodified": 1081,
      "timestart": 0,
      "timeend": 0,
      "discussion": 900,
      "parent": 0,
      "userid": 1081,
      "created": 1583362964,
      "modified": 1583362964,
      "mailed": 1,
      "subject": "Oferta pr\u00e1cticas IT sistemas y microinform\u00e1tica",
      "message": "\n\n<p></p><p>Hola,</p></p></u></u></p><p>Soy Miguel \u00c1ngel G\u00f3mez, responsable de IT y eCommerce de la empresa Diselectric. </p></u></u></p><p>Somos una empresa de distribuci\u00f3n de material el\u00e9ctrico con sede central \r\nubicada en la calle Caracas 5 de Barcelona, y 6 delegaciones \r\nrepartidas por Catalunya. </p></u></u></p><p>Contamos con m\u00e1s de 55 trabajadores, m\u00e1s de 50 equipos inform\u00e1ticos ( entre pc\u00e1 y port\u00e1tiles ), 23 tablets, y 11 servidores Windows Server 2018 ( entre \r\nf\u00edsicos y virtuales ), 5 switches y 7 Firewall de la marca Zyxel, 1 NAS\r\nd\u00e9nde tenemos centrali
```

En esta respuesta se visualiza todas los post que se han realizado en el foro del Moodle especificado. Como se puede ver lo devuelve en forma de array. Además uno de los principales problemas es la visualización de esta respuesta en la aplicación, esto es un inconveniente porque su formato es HTML. Para arreglarlo, en la variable en la que tenemos guardada esa información poner `Html.fromHtml(nombreDeLaVariable)`.

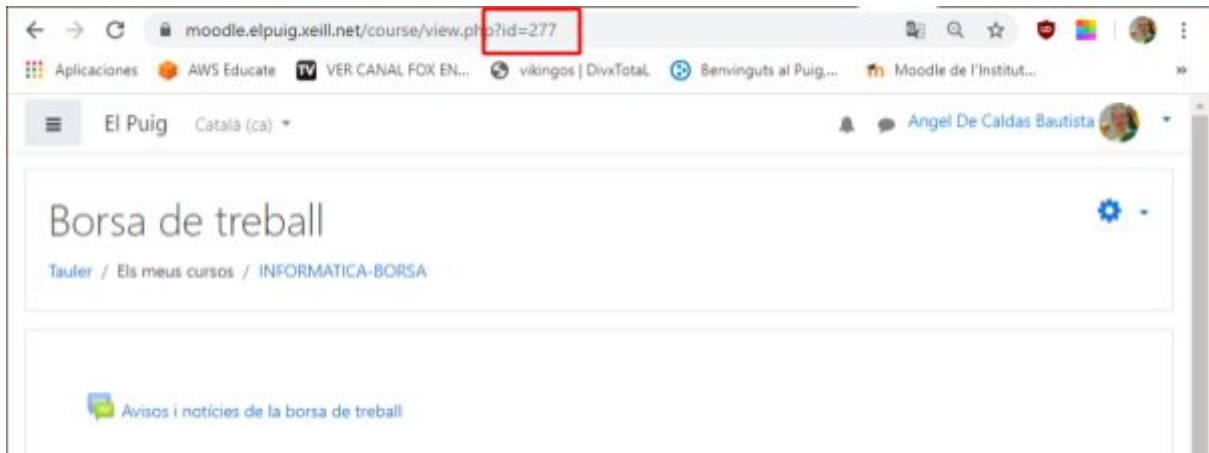
Además la forma de visualizar las diferentes fechas es en el formato es en una unidad de medida bastante difícil de comprender, GMT. Este hace referencia a la zona horaria en la que nos encontramos por lo tanto uno de los problemas que se presentaban eran traducir esta fecha a una entendible, por ello hicimos una clase porque todo las respuestas relacionadas con fechas o horas los valores se representan de esa manera.

```
public static String converter(long time){
    long unix_seconds = time;
    //convert seconds to milliseconds
    Date date = new Date(unix_seconds*1000L);
    // format of the date
    SimpleDateFormat jdf = new SimpleDateFormat("dd-MM-yyyy, HH:mm");
    jdf.setTimeZone(TimeZone.getTimeZone("GMT+1"));
    String java_date = jdf.format(date);
    System.out.println("\n"+java_date+"\n");
    return java_date;
}
```

```
curl
"https://moodle.elpuig.xeill.net/webservice/rest/server.php?wstoken=6492ad7a95e95248d165d715a8d74f35&moodlewsrestformat=json&wsfunction=mod_forum_get_forums_by_courses&courseids[0]=277"
```

```
[{"id":448, "course":277, "type":"news", "name":"Avisos i not\u00e8cies de la borsa de treball", "intro":"Anuncis i not\u00e8cies de la borsa de treball de inform\u00e0tica", "introformat":1, "introfiles":[], "assessed":0, "assess_timestart":0, "assesstimefinish":0, "scale":0, "maxbytes":0, "maxattachments":1, "forcesubscribe":1, "trackingtype":1, "rsstype":0, "rssarticles":0, "timemodified":1558948594, "warnafter":0, "blockafter":0, "blockperiod":0, "completion_discussions":0, "completionreplies":0, "completionposts":0, "cmid":37026, "numdiscussions":19, "cancreatediscussions":false, "lockdiscussionafter":0, "istracked":false}]Angelinodc71@DESKTOP-2Q00757:~$
```

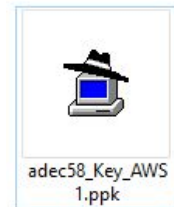
Para poder sacar la lista de discussions se tiene que saber el id de este, para ello se tiene que hacer una consulta en la que con el id del curso (courseids[0]=277) para poder conseguir el id del foro que tiene el curso.



Por ello en link a la hora de acceder a un curso se ve un atributo que pertenece al id del curso.

5. Authorized_keys y Key Pair en AWS

Para poder utilizar SSH de la instancia creada debíamos tener en cuenta que AWS de forma por defecto o de manera automática crea un par de claves en las que se descargan en el portador de la cuenta, esto significa un plus de seguridad pero en el caso de perder ese par de claves haría imposible al usuario acceder al servidor creado, y por lo tanto lo haría inutilizable.



Key pairs (1/1)			
<input type="text" value="Filter key pairs"/>			
<input checked="" type="checkbox"/>	Name	Fingerprint	ID
<input checked="" type="checkbox"/>	adec58_Key_AWS1	6a:8a:e0:e9:0a:91:22:5e:8b:8d:b6:b0:8...	key-02d45c138e24112c1

6. Test de Usabilidad

