



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática

Aplicación para el Acceso a
Plataformas de ELearning desde
Dispositivos Móviles
Documentación Técnica



Presentado por Daniel Puente Gabarri
en Universidad de Burgos — 17 de junio de 2017
Tutor: María Belén Vaquerizo García y Bruno
Baruque Zanón

Índice general

Índice general	I
Índice de figuras	III
Índice de tablas	V
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	9
Apéndice B Especificación de Requisitos	14
B.1. Introducción	14
B.2. Product Backlog	14
B.3. Diagrama de casos de uso	19
Apéndice C Especificación de diseño	23
C.1. Introducción	23
C.2. Diseño de datos	23
C.3. Diagrama de paquetes	26
C.4. Diseño procedimental	27
C.5. Diseño arquitectónico	27
Apéndice D Documentación técnica de programación	28
D.1. Introducción	28
D.2. Estructura de directorios	28
D.3. Manual del programador	28
D.4. Compilación, instalación y ejecución del proyecto	48
D.5. Pruebas del sistema	48

<i>ÍNDICE GENERAL</i>	II
-----------------------	----

Apéndice E Documentación de usuario	49
E.1. Introducción	49
E.2. Requisitos de usuarios	49
E.3. Instalación	49
E.4. Manual del usuario	49
Bibliografía	50

Índice de figuras

A.1. Gráfico Burndown Sprint 1 - Inicio del proyecto.	3
A.2. Gráfico Burndown Sprint 2 - Estudio de refactorización del servidor.	4
A.3. Gráfico Burndown Sprint 3 - Diseño del BackEnd.	5
A.4. Gráfico Burndown Sprint 8 - Ampliación del Front End.	8
A.5. Gráfico Burndown Sprint 8 - Ampliación del Front End.	9
B.1. Diagrama de casos de uso	19
C.1. Clase ControlAccesoProfesor	24
C.2. Clase GestionCuestionario	25
C.3. Clase SolucionCuestionario	26
C.4. Diagrama de paquetes del APIRest	26
C.5. Diagrama de paquetes de la aplicación Android	27
D.1. Instalador de XAMMP	29
D.2. Alerta del firewall de Apache	29
D.3. Panel de XAMPP	30
D.4. Alerta del firewall de MySQL	30
D.5. Carpeta comprimida de Moodle	31
D.6. url de Moodle	32
D.7. Ubicación de donde instalar los directorios	32
D.8. Selección de la base de datos	32
D.9. Configuración de Moodle	33
D.10.url de MySql	33
D.11.Cambiamos el formato de la base de datos	34
D.12.Creación de la tabla Moodle con el formato deseado	35
D.13.Configuramos los datos del administrador de Moodle	35
D.14.Configuramos el sitio	36
D.15.Url de Mysql	36

D.16.Añadimos el fichero de la base de datos	37
D.17.Añadimos la carpeta al htdocs	37
D.18.Abrimos la página de inicio de QuickTest	38
D.19.Entramos en QuickTest	39
D.20.Panel de primeros pasos	40
D.21.Ventana emergente de publicar un cuestionario	41
D.22.Inicio de sesión de Moodle	41
D.23.Creamos un curso	42
D.24.Rellenamos los campos del nuevo curso	42
D.25.Creamos los nuevos usuarios	43
D.26.Rellenamos los campos del primer usuario	43
D.27.Rellenamos los campos del segundo usuario	44
D.28.Agregamos una nueva herramienta externa	45
D.29.Configuramos la nueva herramienta externa 1	46
D.30.Configuramos la nueva herramienta externa 2	46
D.31.Campos de la privacidad	46
D.32.Panel de configuración del Web Service de Moodle	47
D.33.Obtención del token del administrador	48

Índice de tablas

A.1. Coste Hardware	10
A.2. Coste Software 1	10
A.3. Coste Software 2	11
A.4. Coste de aprendizaje	11
A.5. Otros costes	12
A.6. Costes totales	12
B.1. Historia de usuario 1	14
B.2. Historia de usuario 7	15
B.3. Historia de usuario 8	15
B.4. Historia de usuario 2	15
B.5. Historia de usuario 3	16
B.6. Historia de usuario 4	16
B.7. Historia de usuario 5	17
B.8. Historia de usuario 6	17
B.9. Historia de usuario 9	18
B.10. Historia de usuario 10	18
B.11. Historia de usuario 11	18
B.12. Caso de uso 1: Iniciar sesión	19
B.13. Caso de uso 2: Mostrar cuestionarios	20
B.14. Caso de uso 3: Mostrar cuestionarios resueltos	20
B.15. Caso de uso 4: Resolver cuestionario	21
B.16. Caso de uso 5: Olvidar campos	21
B.17. Caso de uso 6: Recordar campos	22

Plan de Proyecto Software

A.1. Introducción

A continuación, en esta sección se encuentra toda aquella información relacionada con la gestión del proyecto. Esta información no solamente va a estar relacionada con el desarrollo del producto pedido, sino que englobará otros aspectos relevantes como la definición de los riesgos o costes del plan proyecto. Como se ha comentado anteriormente en la sección 4.3.3, la metodología de trabajo utilizada para este proyecto es SCRUM. Para llevar cabo al desarrollo del proyecto se ha dividido el desarrollo del mismo en Sprints o Iteraciones de dos semanas. Cabe destacar que algún Sprint varia en esta duración establecida debido a la disponibilidad del equipo para poder llevar a cabo la reunión al finalizar cada iteración.

A.2. Planificación temporal

Como se comentaba, se ha decidido dividir el desarrollo del proyecto en Sprints de dos semanas de duración. Gracias a esta duración se podrá compaginar de una manera más eficiente el desarrollo del proyecto de final de grado junto con el resto de asignaturas de este cuatrimestre. En este cuatrimestre el número de asignaturas cursadas son tres junto con el trabajo de fin de grado. Además, se ha estipulado que el número de horas de dedicación por iteración sea de alrededor de cuarenta, este es el principal motivo por el cual se ha decidido que la duración de cada iteración sea de quince días para poder compaginar el desarrollo del proyecto junto con el resto de asignaturas.

Sprint 1. Inicio del proyecto (7 Feb 2017 a 16 Feb 2017)

Este día tuvo lugar la primera reunión para la explicación sobre el producto a desarrollar. Una vez explicado el mismo, se decidió aceptar este proyecto y

se comenzó con la definición de los principales pilares a superar para construir el producto. Una vez definidas estas bases se decidió realizar en este Sprint las siguientes tareas.

- **Cargar código fuente:** este primer commit fue realizado por los tutores para incluir el proyecto de partida.
- **Objetivos del proyecto:** esta tarea ha sido realizada por el alumno sobre la documentación a entregar. En dicho apartado del documento se definirán las bases principales del proyecto junto con sus objetivos más destacados.
- **Instalar proyecto de partida:** en tarea fue realizada por el alumno, para ello se ha realizado una lectura de la documentación del proyecto de partida y su posterior instalación para conocer al máximo la lógica a implementar.
- **Generar el Product Backlog:** esta tarea fue realizada por el alumno en la memoria. En esta sección se llevó a cabo la definición de los principales requisitos funcionales las cuales derivarán en las historias de usuario a realizar en los diferentes Sprint.
- **Aprendizaje PHP:** esta tarea fue la más costosa de esta iteración ya que se desconocía cualquier conocimiento acorde a este lenguaje. Para ello se realizó una lectura de la documentación proporcionada en la propia web de PHP [7] junto con otras fuentes online como tutoriales sobre PHP desde cero.

A continuación, en la siguiente Ilustración se incluye un gráfico Burndown resumiendo el Sprint, cabe destacar que en esta ocasión solamente se aprecia que todas las tareas fueron finalizadas el mismo día, aunque esto no fue así. Este error nace fruto de un desconocimiento previo de la herramienta, al no incluir en la configuración de la creación de dicho gráfico las tareas que se encuentren dentro del tablero en la sección Done. Posteriormente, una vez localizado y subsanado este error en las siguientes iteraciones este fallo no se produce.



Figura A.1: Gráfico Burndown Sprint 1 - Inicio del proyecto.

Sprint 2. Estudio de refactorización del servidor (16 Feb 2017 a 2 Mar 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 1. En esta iteración se decidió realizar las siguientes tareas:

- Estudio de inclusión de un Framework: esta tarea fue realizada por el alumno y se realizó un estudio de diferentes alternativas como: CakePHP [1], Laravel [4], Symfony [9], Slim [8]. No obstante, la decisión ha sido no utilizar ningún Framework al tratarse de un proyecto pequeño en dimensiones y al añadir mayor complejidad a la resolución del mismo.
- ● Listar los métodos a implementar en el API: esta tarea ha sido realizada por el alumno y en ella se listan las principales lógicas a implementar sobre la base de datos.
- ● Estudio de diferencias entre REST y SOAP: esta tarea también fue realizada por el alumno, para ello se realizó una correcta lectura y posterior documentación en el apartado Soap y Rest. Fruto de este estudio se ha decidido utilizar REST como arquitectura para facilitar el uso y tratamiento de las peticiones desde el programa.
- ● Documentación: Aspectos Teóricos / Técnicas y Herramientas: esta tarea consiste en llevar a cabo una documentación en la memoria sobre estos aspectos en la sección Conceptos teóricos y Técnicas y herramientas.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

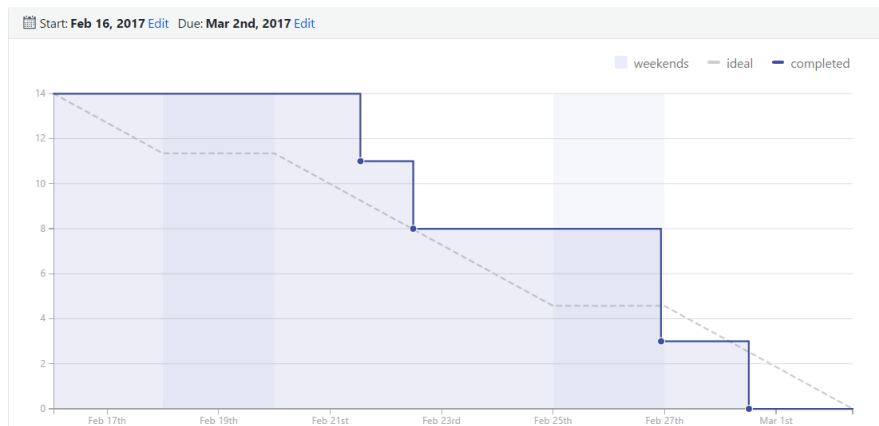


Figura A.2: Gráfico Burndown Sprint 2 - Estudio de refactorización del servidor.

Sprint 3. Diseño del BackEnd (2 Mar 2017 a 15 Mar 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 2. En esta iteración se decidió realizar las siguientes tareas:

- **Implementación de los métodos del API:** esta tarea consiste en realizar la implementación del API REST, para poder comunicarnos con el controlador del proyecto de partida.
- **Diseño arquitectura del BackEnd:** esta tarea consiste en construir los diagramas sobre el BackEnd.
- **Anexo 1 – Planificación del proyecto:** esta tarea consiste en añadir a la documentación este anexo.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.



Figura A.3: Gráfico Burndown Sprint 3 - Diseño del BackEnd.

Sprint 4. Interconexión FrontEnd-BackEnd (15 Mar 2017 a 29 Mar 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 3. En esta iteración se decidió realizar las siguientes tareas:

- **Test unitarios sobre el BackEnd:** en esta tarea se realizó una documentación sobre las pruebas unitarias en PHP. Posteriormente, se realizaron pruebas unitarias sobre las nuevas clases creadas en el APIREST.
- **Leer documentación LTI:** en esta tarea se realizó una lectura sobre el funcionamiento de LTI para su posterior utilización en la aplicación.
- **Prueba conexión Android - Moodle:** esta tarea fue la más complicada del Sprint y posiblemente una de las más costosas del proyecto. Después de mucha lectura se decidió utilizar los Web service de Moodle para obtener la información de la plataforma desde Android.
- **Documentación, Anexo Diseño:** en esta tarea se incluyó en la memoria los diagramas de clases y paquetes del APIREST.
- **Definir ProductBackLog:** en esta tarea se definió el conjunto de historias de usuario a desarrollar.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

Sprint 5. Recuperación de datos desde el BackEnd (29 Mar 2017 a 20 Abr 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 4. En esta iteración se decidió realizar las siguientes tareas:

- **Implementar conexión LTI desde Android:** en esta tarea se realizó la obtención del token del usuario que ha iniciado sesión en la aplicación.
- **MockUp del la App Android:** en esta tarea se realizó una construcción genérica de la app.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

Sprint 6. Recuperación de Datos desde BackEnd (II) (20 Abr 2017 a 4 May 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 5. En esta iteración se decidió realizar las siguientes tareas:

- **Diseño de la arquitectura del FrontEnd:** en esta tarea se incluyó en la memoria el diagrama de clases y paquetes de la aplicación Android.
- **Obtención datos, herramienta externa:** en esta tarea se realizó un estudio sobre los diferentes métodos del Web service de Moodle necesarios para la obtención de los cuestionarios de QuickTest.
- **HU1 - Iniciar sesión:** en esta tarea se proporcionó un diseño al login más cercano al producto final.
- **HU2 - Cuestionarios a resolver:** en esta tarea se obtuvieron los cuestionarios de QuickTest asociados al usuario logeado.
- **Añadir aspectos relevantes hasta el momento en el manual del programador:** en esta tarea se completó esta sección en la memoria.
- **Casos de Uso:** en esta tarea se completó esta sección en la memoria.
- **Obtener datos desde QuickTest:** en esta se incluyó en el APIREST la funcionalidad de obtener un cuestionario por identificador, ya que esta funcionalidad no se encontraba desarrollada en el controlador,
- **Incluir nuevas bibliotecas en Manual Programador:** en esta tarea se añadieron una explicación sobre las librerías Gson y Volley.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

Sprint 7. Grabado de datos desde el BackEnd (4 May 2017 a 18 May 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 6. En esta iteración se decidió realizar las siguientes tareas:

- **HU3 - Resolver un cuestionario:** en esta tarea realizó el desarrollo de esta historia de usuario. Destacar que solamente se permitió que el alumno resolviera un cuestionario sin la posibilidad de los comodines.
- **HU4 - Finalizar un cuestionario:** en esta tarea se permitió que el alumno pueda finalizar un cuestionario desde la aplicación.
- **HU7 - Podrá cerrar sesión:** en esta tarea se realizó el desarrollo de esta historia de usuario en la aplicación.
- **Completar Aspectos Relevantes (Memoria):** en esta tarea se completo esta sección en la memoria.
- **Envío de respuestas desde la app a Quicktest:** en esta tarea se permitió que al finalizar un cuestionario, estos datos sean enviados al APIREST para su almacenamiento.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

Sprint 8. Ampliación del Front End. (18 May 2017 a 8 Jun 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 7. En esta iteración se decidió realizar las siguientes tareas:

- **HU5 - Calificación obtenida:** en esta tarea se incluyó en la aplicación una sección donde el usuario pudiera ver la calificación obtenida en un cuestionario ya resuelto.
- **HU6 - Revisar un cuestionario:** en esta tarea se incluyó en la aplicación una sección donde el usuario pudiera ver una tabla con información sobre el cuestionario.

- **Implementación de Funcionalidad de Comodines:** en esta tarea se completó la HU3 al permitir al alumno utilizar comodines.

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

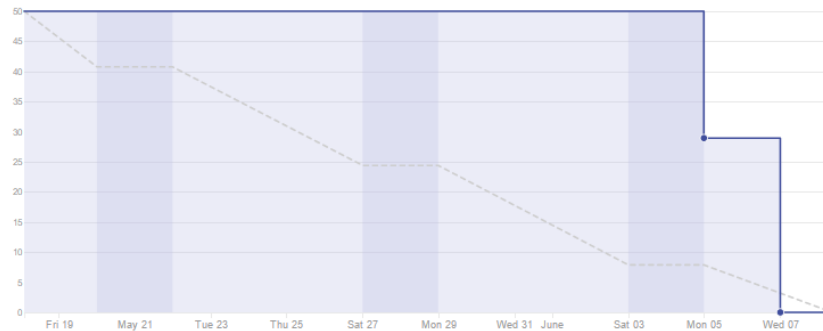


Figura A.4: Gráfico Burndown Sprint 8 - Ampliación del Front End.

Sprint 9. Finalizar FrontEnd y test. (8 Jun 2017 a 19 Jun 2017)

Al comienzo de esta iteración se realizó la revisión de las tareas a realizar en el Sprint anterior y una vez revisadas se dio por finalizado el Sprint 8. En esta iteración se decidió realizar las siguientes tareas:

-

En la siguiente Ilustración podemos observar el gráfico Burndown correspondiente a esta iteración.

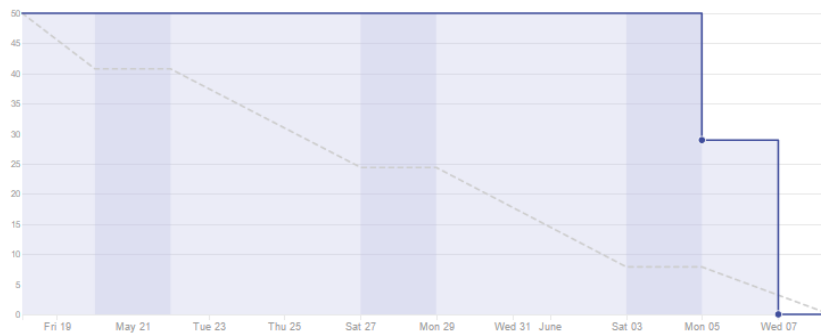


Figura A.5: Gráfico Burndown Sprint 8 - Ampliación del Front End.

A.3. Estudio de viabilidad

Uno de los objetivos principales de cualquier proyecto por no clasificarlo como el más importante es conocer si este proyecto es viable. Para ello se debe realizar un estudio sobre la viabilidad del mismo. Es por esto que se deberá realizar este estudio sobre dos enfoques: *económico y legal*. Y gracias a estos sabremos si el proyecto es rentable y viable.

Viabilidad económica

Relacionado con lo anteriormente expuesto en esta sección se va realizar un estudio sobre la viabilidad del proyecto en el enfoque económico. Para ello se va realizar un análisis sobre los diferentes costes que entran en juego y son necesarios para el desarrollo del mismo junto con el mantenimiento una vez desarrollado.

Coste Hardware

Para el desarrollo del proyecto y posterior corrección de ciertos problemas que puedan surgir será necesario:

- **Un ordenador potente en prestaciones**, ya que el IDE de Android Studio es bastante pesado en tareas principalmente de emulación.
- **Un dispositivo móvil**, es decir un Smartphone con S.O Android.

En este caso se utilizarán las herramientas propias del alumno encargado de realizar el proyecto, que son las siguientes:

- **Asus GL552VW-DM142T**: coste actual 1199 €sin S.O.

- **OnePlus 2:** coste actual 319 €.

Ambos componentes son bastante potentes y por lo tanto no ocasionarán ninguna problemática de que el hardware se quede anticuado en prestaciones en posteriores años. Para redondear los posteriores cálculos estimaremos que el tiempo de vida de ambos componentes será de 5 años, es decir 60 meses. Aunque seguramente su tiempo de vida podría alargarse aún más. Además, al realizarse el proyecto en un cuatrimestre la duración de la utilización de dichos componentes será de 4 meses.

Tiempo de amortización	60 meses
Coste de los componentes	1199,00 € + 319,00 € = 1518,00 €
Coste de amortización/mes	1518,00 € / 60 meses = 25,3 €/mes
Coste final	25,3 €/mes * 4 meses = 101,2 €

Tabla A.1: Coste Hardware

Coste Software

Para la realización del proyecto y su posterior mantenimiento será necesario lo siguiente:

Software	Licencia	Coste
Microsoft Windows 10	Home	135,00 €
Android Studio	Licencia Apache 2.0 [11]	0,00 €
XAMPP	GNU [14]	0,00 €
GitHub	X	X
TortoiseSVN	GNU General Public License [13]	0,00 €
Moodle	GNU GPL [12]	0,00 €
PhpStorm	Estudiante	0,00 €
SonarQube	LGPL	0,00 €
Advanced REST Client	X	X
Microsoft Office	Hogar	99,00 €
		234,00 €

Tabla A.2: Coste Software 1

En este caso la licencia de Microsoft Office es para un único año de duración, pero la licencia del S.O no tiene caducidad por lo que solamente tendremos en cuenta una duración anual.

Tiempo de amortización	12 meses
Coste de los componentes	135,00 € + 99,00 € = 234,00 €
Coste de amortización/mes	234,00 € / 12 meses = 19,5 €/mes
Coste final	19,5 €/mes * 4 meses = 78,00 €

Tabla A.3: Coste Software 2

Coste de instalación

Este coste será nulo y no se tendrá en cuenta ya que la instalación de los distintos componentes será realizada por el propio alumno.

Coste de aprendizaje

Este proyecto será desarrollado por el alumno por lo que este coste de realización será un sueldo mensual a dicho alumno por llevar a cabo el diseño, implementación y testeo del mismo. Como se comentaba anteriormente se ha estipulado que el número de horas dedicadas al desarrollo del mismo será de 40 horas cada dos semanas.

Duración	4 meses * 4 semanas = 16 semanas
Salario/hora	10,00 €/hora
Nº horas/semana	20 horas/semana
Coste final	10,00 €/hora * 20 horas/semana * 16 semanas = 3.200,00 €

Tabla A.4: Coste de aprendizaje

Coste de soporte

Este coste sería realizado por un hipotético técnico que se ocuparía de resolver cualquier incidencia en la utilización del producto desarrollado. Al ser una aplicación que, si resulta viable, será publicada en el Play Store no existirá ningún técnico encargado de subsanar directamente los errores a los usuarios, sino que serán los propios operarios de dicha página los que se remitirán cualquier incidencia. Es por esto que coste de soporte será nulo y por tanto no se tendrá en cuenta.

Coste de mantenimiento

En este caso será el propio alumno el encargado de realizar el mantenimiento de la aplicación por lo que esta tarea se encuentra remunerada dentro de su salario. No obstante, cabe destacar que la distribución desde esta App

no incluye costes de almacenamiento de la información resultantes, es decir, las propias instituciones que compren el producto tendrán que realizar un mantenimiento propio de sus servidores.

Otros costes

Además, también hay que tener en cuenta otros costes como los derivados del *material mobiliario, de oficina, documentación, electricidad, Internet*.

Tipos de coste	Coste	Total
Infraestructura	10,00 €/mes	10,00 €/mes * 4 meses = 40,00 €
Documentación	36,10 €[2]	36,10 €
Electricidad	35,00 €	35 €/mes * 4 = 140 €
Internet	X	37,90 €/mes * 4 = 151,60 €

Tabla A.5: Otros costes

Total

Tipos de coste	Total
Coste Hardware	101,2 €
Coste Software	78,00 €
Coste de instalación	0,00 €
Coste de aprendizaje	3.200,00 €
Coste de soporte	0,00 €
Coste de mantenimiento	367,70 €
Otros coste	367,70 €
	3.746,90 €

Tabla A.6: Costes totales

Análisis coste-beneficio

Como ya se ha explicado, este proyecto nace de la base fundamental de otro proyecto. Este proyecto fue desarrollado con una base y un fin muy bien definido y fijado, la herramienta QuickTest es compatible con cualquier plataforma que cumpla el estándar LTI. Fruto de esta idea se desarrolló este proyecto como un complemento de resolución de cuestionarios para Moodle. Es por esto que este análisis de coste-beneficio se va realizar para Moodle. Moodle cuenta actualmente con más de 100 millones de usuarios en todo el mundo, lo que la

convierte en la plataforma digital más utilizada en todo mundo por la comunidad educativa [5] [6]. Además, se encuentra registrada en 234 países, en la que se encuentra España en el segundo puesto con 7.152 inscripciones. Como el principal objetivo de esta aplicación es la utilización de la misma para un entorno universitario vamos a reducir estos cálculos de inscripciones al número de universidades en España. Actualmente, el número de universidades tanto públicas como privadas es de 82. Al ser el proyecto compatible con Moodle y al ser este, la plataforma digital más influyente en este sector, cabe destacar que no deberíamos de tener ningún inconveniente en vender licencias de este software en el PlayStore. Sería una suposición muy idealizada que todas estas comprarán nuestro producto por lo que realizaremos los cálculos estimando que solamente 40 universidades deciden adquirir la licencia. Ante este estudio preliminar, se decide vender cada licencia por 100,00 €. Este precio solamente incluirá el producto junto con la documentación necesarios para su correcta utilización, es decir, no se incluye un soporte de almacenamiento de la información generada por su utilización. Además, esta distribución de licencia se hará con carácter comercial [10]. Finalmente, cabe destacar que valoraremos este análisis como factible ya que únicamente distribuyendo estas 40 licencias recuperaríamos la inversión realizada.

Viabilidad legal

Al utilizar herramientas que se encuentran bajo licencias de software libre la distribución de esta herramienta no ocasionará ninguna problemática legal. No obstante, la herramienta fruto de este proyecto se distribuirá con la licencia Android License Verification Library [3] propia de PlayStore y utilizada por cualquier aplicación que resida en la misma para evitar posibles pirateos. Como ya se comentaba en la sección anterior este producto se distribuirá como licencia comercial, ya que todos los activos utilizados para el desarrollo del mismo poseen licencia GNU.

Apéndice B

Especificación de Requisitos

B.1. Introducción

Como ya se ha comentado anteriormente, se va utilizar la metodología ágil SCRUM. Es por esto que no vamos a tratar las tareas a realizar como requisitos, sino como historias de usuario. Todas estas historias de usuarios formarán parte del Product Backlog y serán asignadas a determinadas iteraciones durante el desarrollo del proyecto. A continuación, se incluye un listado sobre las historias de usuarios y sus correspondientes diagramas de casos de uso.

B.2. Product Backlog

En esta sección se incluyen todas aquellas historias de usuarios necesarias para los alumnos y los profesores.

Historia de usuario 1: Iniciar sesión

HU1	
Título	Como un usuario autenticado, podrá iniciar sesión.
Rol	Alumno o profesor autenticado.
Descripción	El usuario introducirá sus credenciales y se le mostrará aquellos cursos en los que se encuentre matriculado.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.1: Historia de usuario 1

Historia de usuario 7: Cerrar sesión

HU7	
Título	Como un usuario autenticado, podrá cerrar sesión.
Rol	Alumno o profesor autenticado.
Descripción	Siempre y cuando el usuario se encuentre logeado, podrá cerrar sesión.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.2: Historia de usuario 7

Historia de usuario 8: Recordar campos

HU8	
Título	Como un usuario autenticado, podrá recordar los campos.
Rol	Alumno o profesor autenticado.
Descripción	El sistema permitirá recordar los campos con los que usuario inicia sesión.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.3: Historia de usuario 8

Historia de usuario 12: Cursos

HU12	
Título	Como un usuario autenticado, podrá ver sus cursos.
Rol	Alumno o profesor autenticado.
Descripción	El sistema mostrará los cursos en los que se encuentra matriculado.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.4: Historia de usuario 12

Historias de usuario de los alumnos**Historia de usuario 2: Cuestionarios**

HU2	
Título	Como un usuario autenticado, el sistema notificará sobre los cuestionarios a resolver.
Rol	Alumno autenticado.
Descripción	Una vez el usuario haya iniciado sesión obtendrá todos aquellos cuestionarios en los que se encuentre asignado.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión. - Existan cuestionarios en aquellos cursos en los que se encuentra matriculado el alumno.

Tabla B.5: Historia de usuario 2

Historia de usuario 3: Resolver cuestionario

HU3	
Título	Como un usuario autenticado, podrá resolver un cuestionario.
Rol	Alumno autenticado.
Descripción	Una vez el usuario haya iniciado sesión podrá resolver aquellos cuestionarios que se encuentren asignados, a un curso al que pertenezca.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión. - Existan cuestionarios en aquellos cursos en los que se encuentra registrado el alumno.

Tabla B.6: Historia de usuario 3

Historia de usuario 4: Finalizar cuestionario

HU4	
Título	Como un usuario autenticado, podrá finalizar un cuestionario.
Rol	Alumno autenticado.
Descripción	El usuario podrá finalizar aquel cuestionario que este resolviendo
Precondiciones	- El usuario debe estar previamente registrado.
	- Haber iniciado sesión.
	- Existan cuestionarios en aquellos cursos en los que se encuentra registrado el alumno.
	- Haber iniciado la resolución de un cuestionario.

Tabla B.7: Historia de usuario 4

Historia de usuario 5: Calificación obtenida

HU5	
Título	Como un usuario autenticado, el sistema notificará sobre la calificación obtenida.
Rol	Alumno autenticado.
Descripción	Una vez el usuario finalice el cuestionario, el sistema proporcionará una retroalimentación sobre el cuestionario resuelto.
Precondiciones	- El usuario debe estar previamente registrado.
	- Haber iniciado sesión.
	- Existan cuestionarios en aquellos cursos en los que se encuentra registrado el alumno.
	- Haber iniciado la resolución de un cuestionario.
	- Haber finalizado un cuestionario.

Tabla B.8: Historia de usuario 5

Historia de usuario 6: Revisar cuestionario resuelto

HU6	
Título	Como un usuario autenticado, podrá revisar un cuestionario resuelto.
Rol	Alumno autenticado.
Descripción	El usuario podrá revisar la retroalimentación de un cuestionario que ya esté finalizado.
Precondiciones	<ul style="list-style-type: none"> - El usuario debe estar previamente registrado. - Haber iniciado sesión. - Existan cuestionarios en aquellos cursos en los que se encuentra registrado el alumno. - Haber iniciado la resolución de un cuestionario. - Haber finalizado un cuestionario.

Tabla B.9: Historia de usuario 6

Historias de usuario de los profesores**Historia de usuario 9: Cuestionarios**

HU9	
Título	Como un usuario autenticado, podrá ver los cuestionarios a los que se enfrentan sus alumnos.
Rol	Profesor autenticado.
Descripción	El sistema mostrará al profesor los cuestionarios a los que se enfrentan sus alumnos.
Precondiciones	<ul style="list-style-type: none"> - El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.10: Historia de usuario 9

Historia de usuario 10: Calificaciones

HU10	
Título	Como un usuario autenticado, podrá ver las calificaciones de los alumnos en un cuestionario.
Rol	Profesor autenticado.
Descripción	El sistema mostrará al profesor la calificación de cada alumno en cada cuestionario.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.11: Historia de usuario 10

Historia de usuario 11: Ver cuestionarios

HU11	
Título	Como un usuario autenticado, podrá ver los cuestionarios.
Rol	Profesor autenticado.
Descripción	El sistema mostrará al profesor los cuestionarios a los que se enfrentan sus alumnos.
Precondiciones	- El usuario debe estar previamente registrado. - Haber iniciado sesión.

Tabla B.12: Historia de usuario 11

B.3. Diagrama de casos de uso

En la siguiente ilustración podemos ver el diagrama de los casos de uso de nuestro sistema:

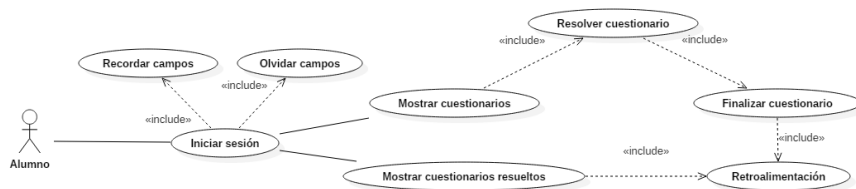


Figura B.1: Diagrama de casos de uso

Caso de uso 1: Iniciar sesión

CU1	
Título	Iniciar sesión.
Descripción	El sistema permitirá al usuario autenticado iniciar sesión para acceder al sistema, así los cuestionarios para ese usuario serán visibles una vez el usuario inicie sesión.
Secuencia	1. El usuario introducirá los campos de cuenta y contraseña. 2. El sistema comprobará si los campos son correctos: 2.1. Si son correctos, el usuario pasa a estar logeado teniendo pleno acceso a las funcionalidades del sistema. 2.2. Si no son correctos, el sistema notificará que los campos son incorrectos.
Precondiciones	- El usuario debe estar previamente registrado en Moodle.
Comentarios	Este caso de uso, CU1, corresponde a la historia de usuario 1.

Tabla B.13: Caso de uso 1: Iniciar sesión

Caso de uso 2: Mostrar cuestionarios

CU2	
Título	Mostrar cuestionarios.
Descripción	El sistema permitirá al usuario ver los cuestionarios de QuickTest en los que se encuentra
Secuencia	1. El sistema mostrará los cuestionarios.
Precondiciones	- El usuario debe estar previamente registrado en Moodle. - El usuario debe estar logeado en la aplicación para poder acceder a esta funcionalidad. - Deben existir cuestionarios de QuickTest en cursos en los que el usuario se encuentre matriculado.
Comentarios	Este caso de uso, CU2, corresponde a la historia de usuario 2.

Tabla B.14: Caso de uso 2: Mostrar cuestionarios

Caso de uso 3: Mostrar cuestionarios resueltos

CU3	
Título	Mostrar cuestionarios resueltos.
Descripción	El sistema permitirá al usuario ver los cuestionarios de QuickTest en los que se encuentra y ya ha resuelto.
Secuencia	1. El sistema mostrará los cuestionarios resueltos. 1.1. Si existe algún cuestionario resuelto el usuario podrá seleccionar un cuestionario y obtener la retroalimentación del mismo HU5, HU6.
Precondiciones	- El usuario debe estar previamente registrado en Moodle. - El usuario debe estar logeado en la aplicación para poder acceder a esta funcionalidad. - Deben existir cuestionarios de QuickTest en cursos en los que el usuario se encuentre matriculado y que hayan sido resueltos.
Comentarios	Este caso de uso, CU3, corresponde a la historia de usuario 6.

Tabla B.15: Caso de uso 3: Mostrar cuestionarios resueltos

Caso de uso 4: Resolver cuestionario

CU4	
Título	Resolver cuestionario.
Descripción	El sistema permitirá al usuario resolver un cuestionario.
Secuencia	1. El usuario seleccionará un cuestionario a resolver. 1.1. Si el cuestionario está resuelto, el sistema mostrará la retroalimentación del mismo. 1.2. Sino, el sistema mostrará el cuestionario a resolver HU3. 1.2.1. Una vez finalizado el cuestionario, el sistema enviará el cuestionario resuelto HU4. 1.2.2. El sistema mostrará la retroalimentación del cuestionario resuelto HU5, HU6.
Precondiciones	- El usuario debe estar previamente registrado en Moodle. - El usuario debe estar logeado en la aplicación para poder acceder a esta funcionalidad. - Deben existir cuestionarios de QuickTest en cursos en los que el usuario se encuentre matriculado y que hayan sido resueltos.
Comentarios	Este caso de uso, CU4, corresponde a la historia de usuario 3, 4, 5, 6.

Tabla B.16: Caso de uso 4: Resolver cuestionario

Caso de uso 5: Olvidar campos

CU5	
Título	Recordar campos.
Descripción	El sistema permitirá al usuario olvidar los campos con las que el usuario inicia sesión.
Secuencia	1. El usuario seleccionara en el sistema cerrar sesión. 2. El sistema olvidará las credenciales del usuario.
Precondiciones	- El usuario debe estar previamente registrado en Moodle. - El usuario debe haber iniciado sesión y haber activado la opción de recordar campos.
Comentarios	Este caso de uso, CU5, corresponde a la historia de usuario 7.

Tabla B.17: Caso de uso 5: Olvidar campos

Caso de uso 6: Recordar campos

CU6	
Título	Olvidar campos.
Descripción	El sistema permitirá al usuario recordar los campos con las que el usuario inicia sesión.
Secuencia	1. El sistema recordará las credenciales del usuario.
Precondiciones	- El usuario debe estar previamente registrado en Moodle.
Comentarios	Este caso de uso, CU6, corresponde a la historia de usuario 8.

Tabla B.18: Caso de uso 6: Recordar campos

Especificación de diseño

C.1. Introducción

A continuación, se incluye una explicación/descripción del diseño de la aplicación Android. Para ello se dividirán los diagramas en dos enfoques: el APIREST utilizado para comunicarse con el proyecto de partida y la propia app Android que se comunicará con dicho API para la obtención y volcado de los datos.

C.2. Diseño de datos

En este apartado se incluirán los diagramas de clases y paquetes de ambos enfoques.

Diagrama de clases

En estos diagramas se ha decidido utilizar un criterio de color para diferenciar aquellas clases que forman parte del APIREST, en color azul y en amarillo aquellos que forman parte del proyecto de partida.

APIREST

Este paquete será el encargado de comunicarse con el Controlador del proyecto de partida. A continuación, se mostrarán de manera fraccionada cada una de las clases que lo forman y con qué clases del controlador del proyecto de partida se comunican.

Clase SolucionCuestionario Esta clase será la encargada de realizar el control de acceso por parte de los profesores a la web QuickTest. Esta web

proporciona a los profesores una clave privada dentro de la aplicación para poder utilizarla en la sincronización de sus cuestionarios con Moodle. Esta clase permite registrar a un nuevo profesor o comprobar si puede iniciar sesión. Ambas lógicas se controlan mediante una petición post. Para llevar a cabo este cometido se comunican con el controlador llamando al método registrarNuevoUsuario o login respectivamente.

- **Post:** petición que permitirá logearse o registrarse en QuickTest. Para logearse http://localhost/_QuickTest_TFG/app/apiREST/controlAccesoProfesor/login. Para registrarse http://localhost/_QuickTest_TFG/app/apiREST/controlAccesoProfesor/registro.

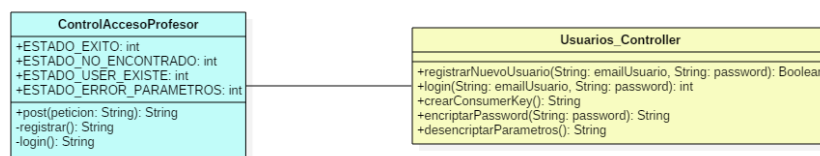


Figura C.1: Clase ControlAccesoProfesor

Clase GestionCuestionario La lógica de esta clase será toda aquella que se encuentre relacionada con la gestión sobre los Cuestionarios: insertar, obtener, duplicar o borrar. Cabe destacar, que en el controlador del proyecto de partida se ha incluido un método para poder duplicar un cuestionario existente, de esta manera evitamos comunicarnos con el modelo. Para ello se comunica con el controlador. Todas estas acciones se realizan mediante diferentes peticiones:

- **Get:** esta petición será la encargada de mostrar aquellos cuestionarios dada un identificador de asignatura. http://localhost/_QuickTest_TFG/app/apiREST/gestionCuestionario/id
- **Post:** esta petición maneja dos lógicas: duplicar o insertar/editar un cuestionario. Para duplicar un cuestionario basta con indicar el identificador de cuestionario y para insertar o editar será necesario incluir en esta petición el resto de información propia de un cuestionario. Cabe destacar que el método insertar manejará la lógica de insertar o editar un cuestionario en función de si un campo se encuentra a verdadero o falso. Esta lógica venía dado por el proyecto de partida. Para duplicar un cuestionario http://localhost/_QuickTest_TFG/app/apiREST/gestionCuestionario/duplicar/id. Para insertar/editar un cuestionario http://localhost/_QuickTest_TFG/app/apiREST/gestionCuestionario/insertar.

- **Delete:** esta petición se encarga de eliminar un determinado cuestionario dado un identificador. http://localhost/_QuickTest_TFG/app/apiREST/gestionCuestionario/id.

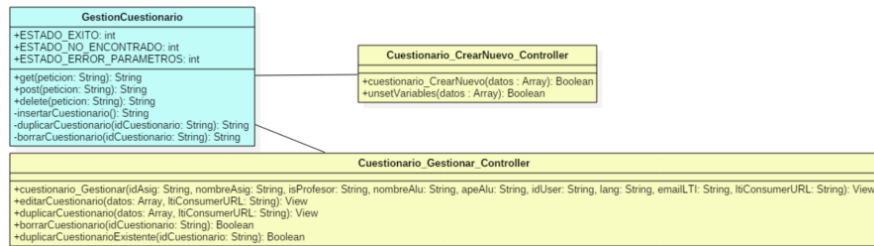


Figura C.2: Clase GestionCuestionario

Clase SolucionCuestionario Esta clase se encarga de la lógica a realizar durante la resolución de un cuestionario: iniciar, finalizar o mostrar resultado. Para ello se comunica con el controlador. Destacar que al tener que basarse en el proyecto de partida existen ciertas funcionalidades que no se encuentran en el controlador. Es por esto que será necesario comunicarse también con el modelo. Todas estas acciones se realizan mediante peticiones post y son las siguiente:

- **Post:** esta petición será la encargada de manejar toda la lógica de esta clase, es decir, iniciar un cuestionario, finalizar un cuestionario y mostrar la calificación. Para iniciar un cuestionario http://localhost/_QuickTest_TFG/app/apiREST/solucionCuestionario/resolver. Para finalizar un cuestionario http://localhost/_QuickTest_TFG/app/apiREST/solucionCuestionario/finalizar. Para mostrar la calificación http://localhost/_QuickTest_TFG/app/apiREST/solucionCuestionario/mostrar

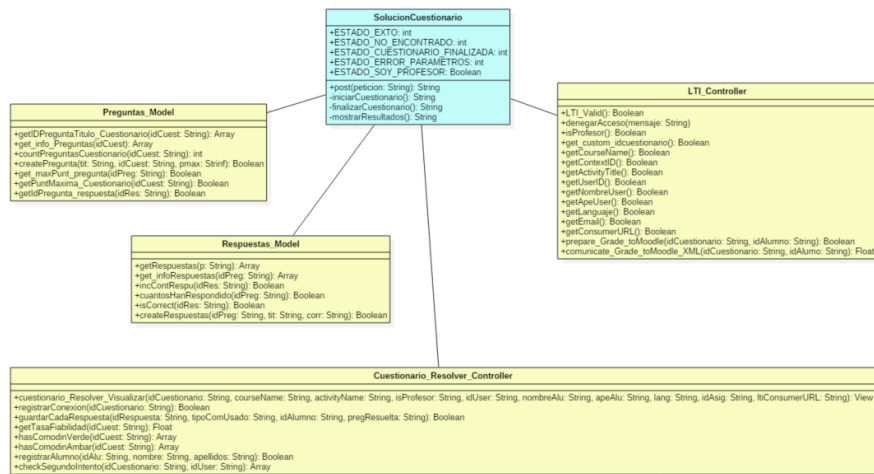


Figura C.3: Clase SolucionCuestionario

Android

C.3. Diagrama de paquetes

Para estos diagramas se ha decidido utilizar el mismo criterio de color empleando anteriormente en el diagrama de clases.

APIREST

Este diagrama muestra un desglose de todos los componentes que entran en contacto en el API, como ya se ha comentado en color amarillo corresponde a la lógica dada en el proyecto de partida y en azul el APIREST desarrollado para interactuar con el controlador.

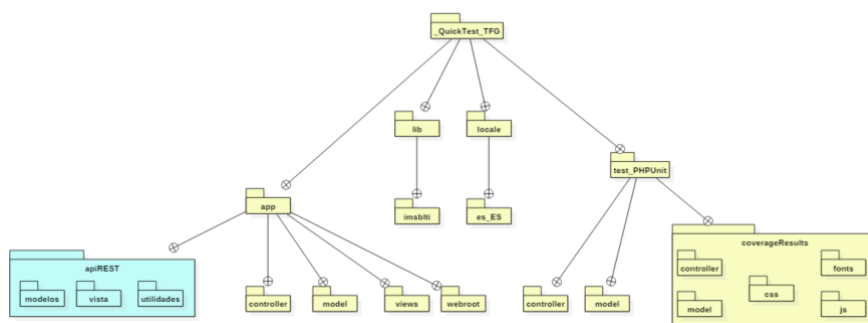


Figura C.4: Diagrama de paquetes del APIRest

Aplicación Android

En este diagrama se muestra el desglose de todos los componentes que forman la arquitectura del FrontEnd de la aplicación.

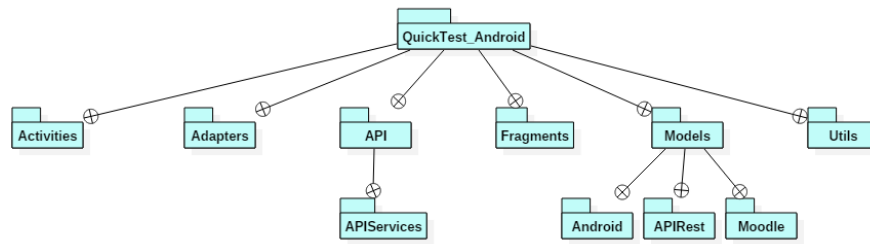


Figura C.5: Diagrama de paquetes de la aplicación Android

C.4. Diseño procedimental

Diagrama de secuencias

C.5. Diseño arquitectónico

Diagrama de despliegue

Documentación técnica de programación

D.1. Introducción

D.2. Estructura de directorios

D.3. Manual del programador

Instalación de las herramientas

Sistema operativo

Servidor

- Acceder a la web oficial de XAMPP en el siguiente enlace: <https://www.apachefriends.org/es/download.html>.
- Seleccionar la versión 5.6.30/PHP 5.6.30. Cabe destacar que es necesario instalar esta versión de XAMPP o inferiores, debido a que el proyecto de partida utiliza ciertas funcionalidades que se encuentran sin soporte en versiones superiores. Y comenzamos a descargar el instalador.

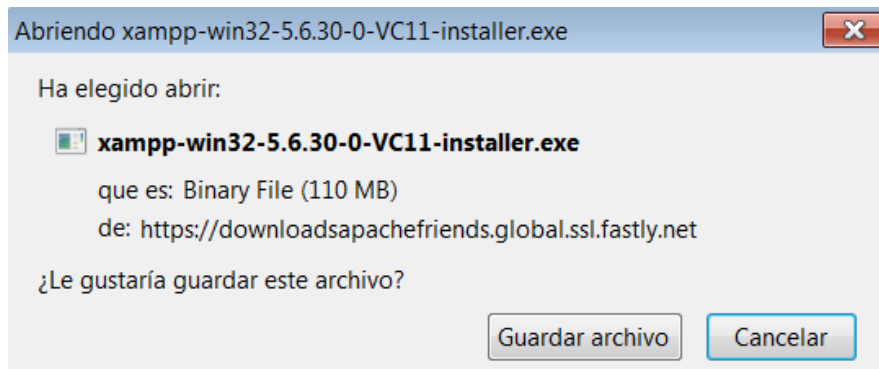


Figura D.1: Instalador de XAMMP

- Una vez descargado, procedemos a la instalación de esta herramienta. Para procedemos a realizar la típica instalación de Siguiente – Siguiente y seleccionamos donde ubicar los archivos a instalar.
- Al finalizar la instalación se lanzará una alerta del firewall de Windows indicando que se necesita conceder acceso a Apache. Marcamos todas las opciones y finalizamos.

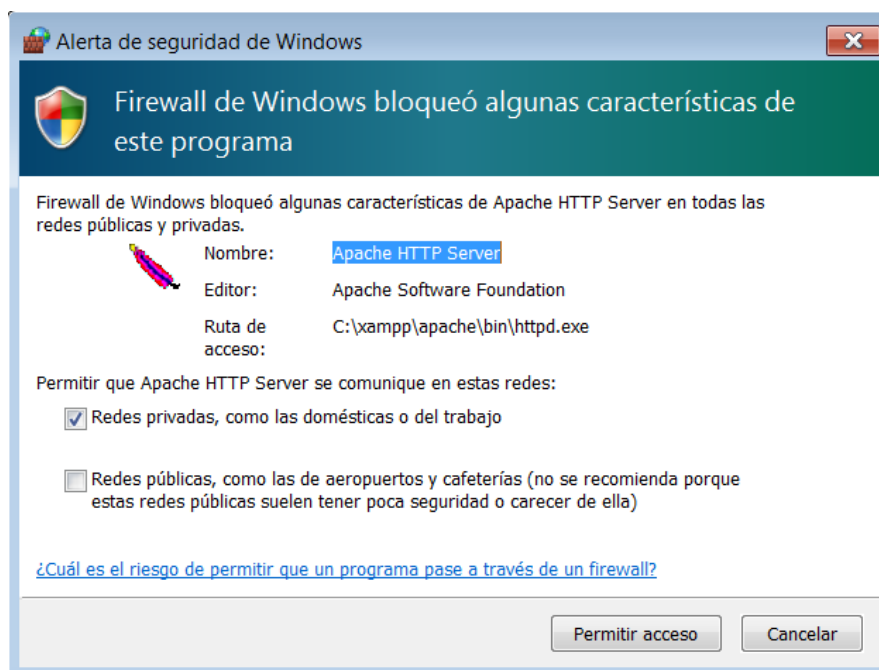


Figura D.2: Alerta del firewall de Apache

- Una vez instalado, ya tendremos instalado y disponible nuestro servidor XAMPP. Para el correcto funcionamiento del proyecto de partida y de QuickTest en Android es necesario arrancar Apache y MySQL al iniciar esta herramienta como podemos ver en la siguiente ilustración.

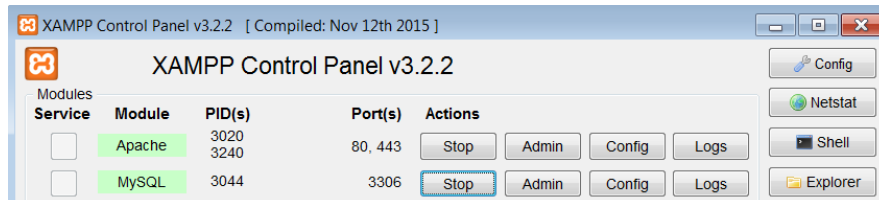


Figura D.3: Panel de XAMPP

- Además, al arrancar estos servicios nos volverá a saltar el firewall de Windows pidiendo conceder permisos a MySQL.

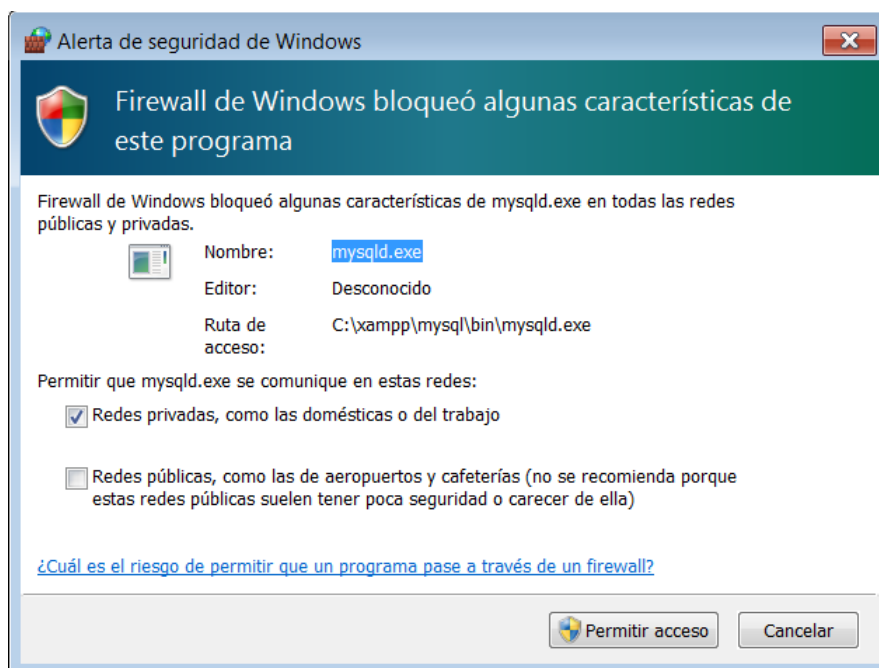


Figura D.4: Alerta del firewall de MySQL

- Concedemos los permisos y la instalación de nuestro servidor a finaliza-
- do.

Moodle

Respecto al proyecto de partida se decidió utilizar como LMS, el mismo utilizado por la Universidad de Burgos, Moodle. Al tener que basarse en este proyecto no se ha realizado ningún cambio en la herramienta utilizada como LMS. No obstante, en esta ocasión la versión de Moodle empleada es la más actual hasta la fecha Moodle 3.2.2+. Este cambio en la versión utilizada es realizado al tener que utilizar las funciones que proporciona el propio web services de Moodle. De manera que cuanto más actual sea la versión, más funcionalidades tendremos a nuestra disposición.

- Descargamos el archivo comprimido de Moodle correspondiente a la versión 3.2.2+: <https://sourceforge.net/projects/moodle/files/Moodle/stable32/moodle-3.2.2.zip/download>
- Seleccionamos la versión comentada anteriormente y comenzamos la descarga del archivo comprimido.

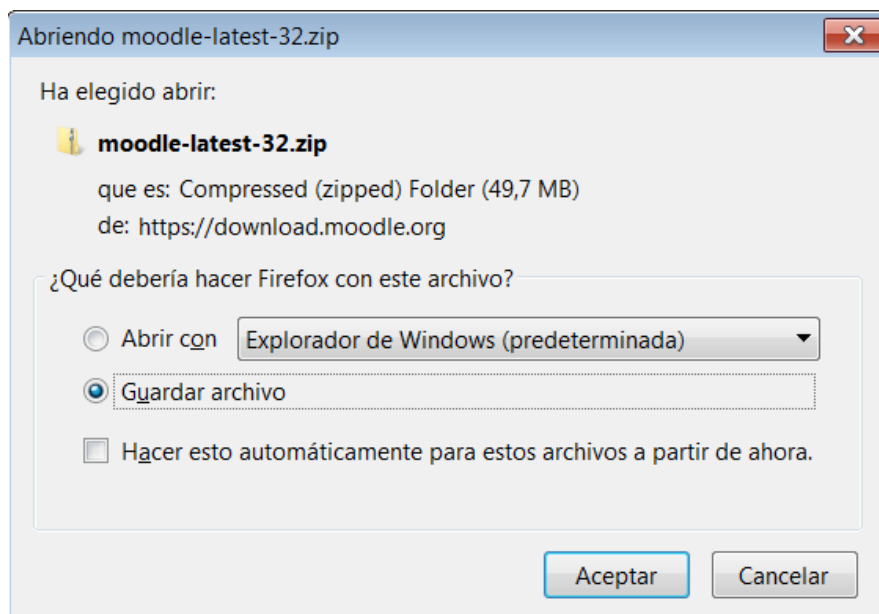


Figura D.5: Carpeta comprimida de Moodle

- Una vez finalizada la descarga, descomprimos el archivo en la carpeta htdocs. Esta carpeta se encuentra en el directorio raíz de XAMPP. Si la ruta de instalación de XAMPP se ha realizado con los campos por defecto, esta se encontrará en "C:\xampp\htdocs". Esta es la carpeta raíz del localhost, desde donde el servidor XAMPP comenzará a buscar cuando le mandemos una petición.

- Una vez descomprimido la carpeta abrimos un navegador y realizamos una petición al servidor indicando que queremos acceder a la URL de Moodle.

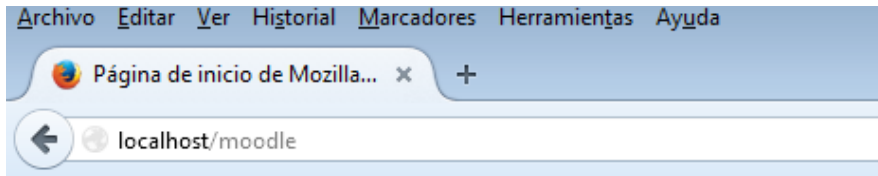


Figura D.6: url de Moodle

- Al no tener instalada ni configurada la herramienta, nos pedirá seleccionar el idioma deseado. Una vez seleccionado el idioma, tendremos que seleccionar la ruta de instalación de los nuevos componentes. En este caso dejamos la configuración por defecto como podemos ver en la siguiente ilustración.

Dirección Web

Directorio de Moodle

Directorio de Datos

« Anterior **Siguiente »**

Figura D.7: Ubicación de donde instalar los directorios

- Posteriormente, tendremos que seleccionar el tipo de base de datos a utilizar. En este caso la opción elegida es MariaDB para evitar problemas de compatibilidad.

Seleccione el controlador de la base de datos

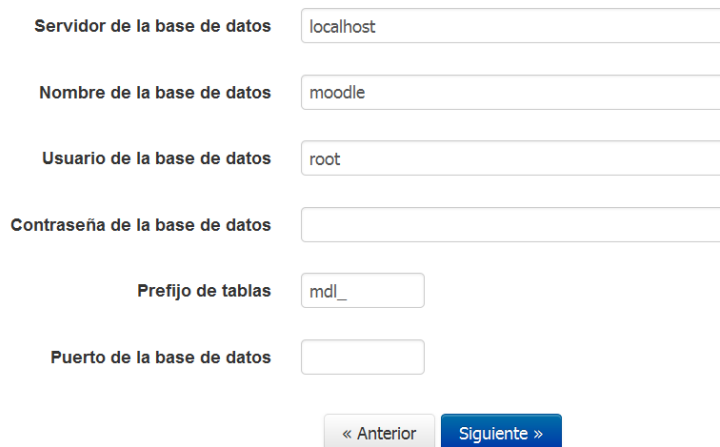
Moodle soporta varios tipos de servidores de base de datos. Por favor, póngase en contacto con el administrador del servidor si no sabe qué tipo usar.

Tipo

« Anterior **Siguiente »**

Figura D.8: Selección de la base de datos

- Al finalizar la selección de la base de datos la instalación nos permitirá modificar la configuración por defecto de la misma. En este caso solamente añadiremos root en la sección Usuario de la base de datos como podemos ver en la siguiente ilustración y aceptamos la licencia.



The image shows a web form for configuring a Moodle database. It contains several labeled input fields: 'Servidor de la base de datos' with 'localhost', 'Nombre de la base de datos' with 'moodle', 'Usuario de la base de datos' with 'root', 'Contraseña de la base de datos' (empty), 'Prefijo de tablas' with 'mdl_', and 'Puerto de la base de datos' (empty). At the bottom, there are two buttons: '« Anterior' and 'Siguiente »'.

Figura D.9: Configuración de Moodle

- Antes de pulsar en el botón de siguiente tendremos que abrir una nueva pestaña en nuestro navegador e introducir el siguiente enlace localhost/phpmyadmin.

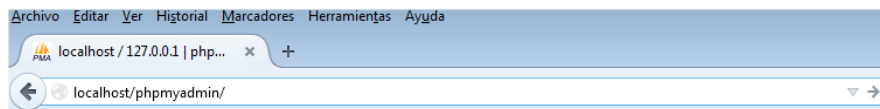


Figura D.10: url de MySql

- Una vez nos encontremos en la página tendremos que cambiar el cotejamiento de la conexión al servidor como podemos ver en la siguiente ilustración.

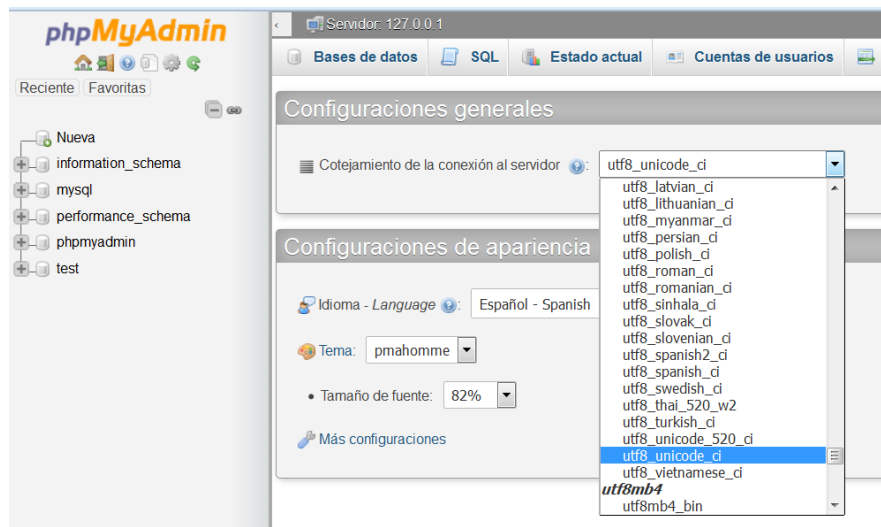


Figura D.11: Cambiamos el formato de la base de datos

- Una vez realizada la creación de la base de datos de manera manual, podemos volver a la pestaña de instalación de Moodle y proseguir con la instalación.
- Si hemos realizado los anteriores pasos de forma correcta el sistema nos comunicará que el entorno cumple todos los requerimientos mínimos y continuamos con la instalación. Este proceso puede durar varios minutos dependiendo de la conexión a Internet.
- En esa misma página seleccionamos en Base de datos. Dentro de esta sección en el apartado Crear base de datos introducimos la siguiente información y la creamos.

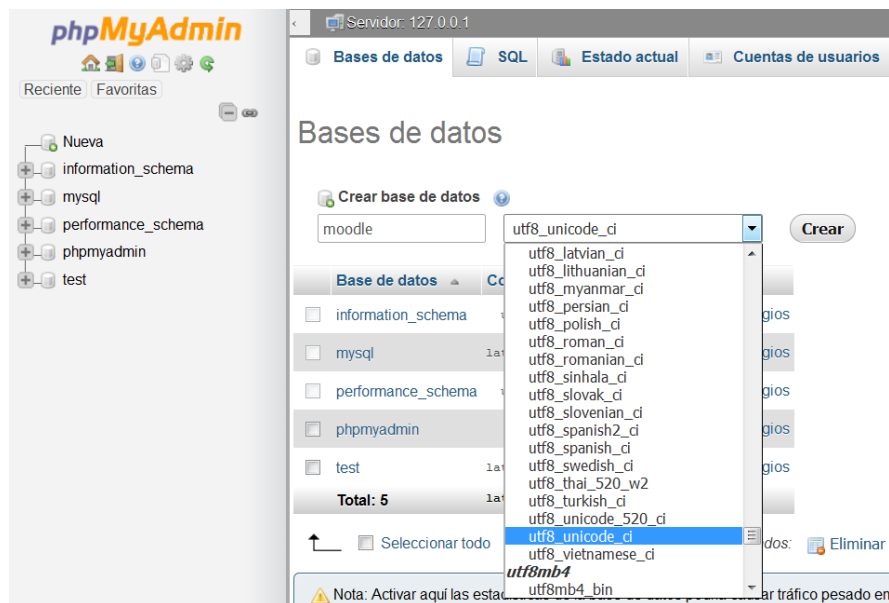


Figura D.12: Creación de la tabla Moodle con el formato deseado

- Una vez realizada la instalación de Moodle, es necesario la configuración de la plataforma. Para ello es necesario establecer un administrador de la misma. En nuestro caso la configuración es la que se puede ver en la siguiente ilustración.

▼ General

Nombre de usuario

Escoger un método de identificación: ☐ Cuentas manuales

La contraseña debería tener al menos 8 carácter(es), al menos 1 dígito(s), al menos 1 minúscula(s), al menos 1 mayúscula(s), al menos 1 carácter(es) no alfanuméricos como *, -, o #

Nueva contraseña

☐ Forzar cambio de contraseña

Nombre

Apellido(s)

Dirección de correo

Figura D.13: Configuramos los datos del administrador de Moodle

- A continuación, tendremos que configurar el sitio. Para ello introducimos la siguiente información.

Instalación

Nuevos ajustes - Ajustes de la página principal

Nombre completo del sitio <small>fullname</small>	<input type="text" value="UBUVirtual TFG"/>
Nombre corto para el sitio (una palabra) <small>shortname</small>	<input type="text" value="UBUVirtual"/>

Figura D.14: Configuramos el sitio

- Una vez realizamos de forma correcta todos los anteriores pasos, la instalación y configuración de Moodle ha finalizado

Instalar base de datos

A continuación, vamos a importar la base de datos que utiliza QuickTest. Esta base de datos es la misma proporcionada por el proyecto de partida. Para llevar a cabo esta tarea tendremos que seguir los siguientes pasos:

- Dentro del proyecto de partida nos situamos en la ruta `src/InstalarBaseDeDatos`.
- En esta carpeta tendremos un único archivo llamado: `instalar_BaseDatosQuickTest.sql`
- Abrimos una nueva pestaña en nuestro navegador e introducimos el siguiente enlace.

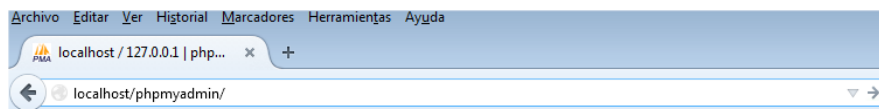


Figura D.15: Url de Mysql

- Una vez nos encontremos en el panel de administración de la base de datos tendremos que importar el archivo del punto 2. Para ello seguimos los siguientes pasos.

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN³⁷

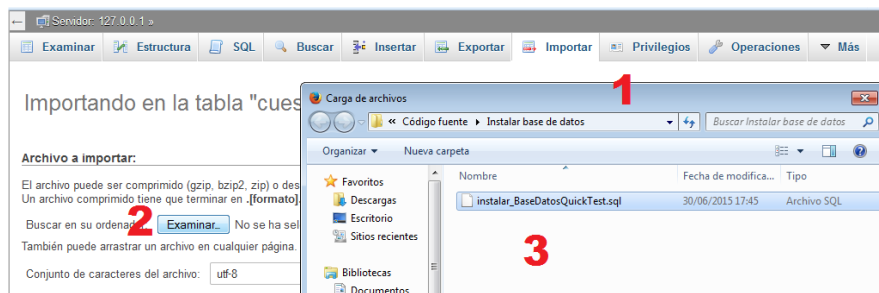


Figura D.16: Añadimos el fichero de la base de datos

- Una vez importado el archivo seleccionamos continuar y el proceso de importación de la base de datos de QuickTest ha finalizado.

Instalar/Configurar QuickTest

Para instalar QuickTest tendremos copiar la carpeta `_QuickTest.TFG` en la carpeta raíz del localhost de XAMPP para que el servidor pueda buscar el directorio pedido al mandarle una petición. Para ello tendremos que:

- Copiar la carpeta `_QuickTest.TFG` en la carpeta `htdocs` de XAMPP.

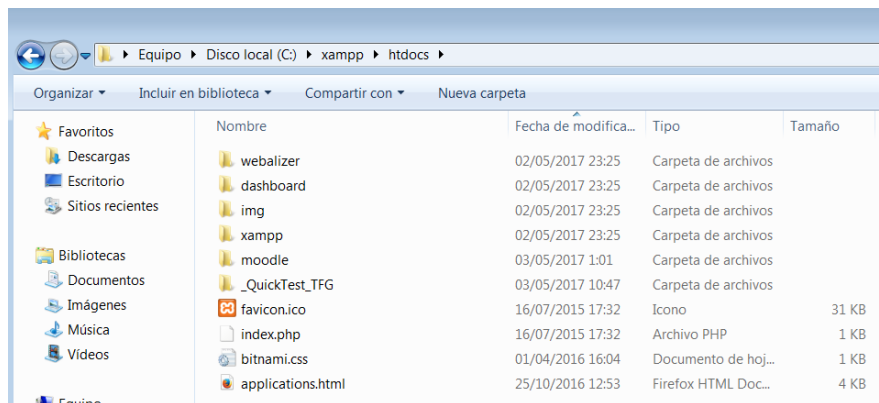


Figura D.17: Añadimos la carpeta al htdocs

Registrar un usuario en QuickTest

Para poder utilizar QuickTest desde un LMS, en este caso Moodle, tendremos que estar previamente registrado en QuickTest. Para ello tendremos que:

- Abrir una nueva pestaña en el navegador e introducir el siguiente enlace:
http://localhost/_QuickTest_TFG/app/views/managementView/startQuickTest_View.php

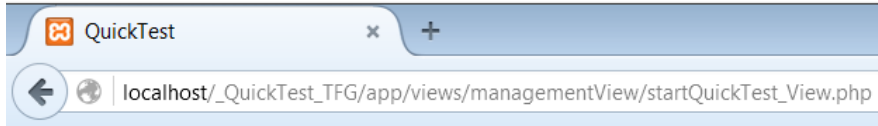


Figura D.18: Abrimos la página de inicio de QuickTest

- Seleccionamos Acceder/Registrarse. Una vez seleccionado este botón aparecerá una ventana emergente que permitirá Entrar o Registrarse.
- Para llevar a cabo el registro de un nuevo usuario en QuickTest tendremos que seguir dos criterios importantes.
 - En el campo Email tendremos que usar el mismo que en Moodle.
 - En el campo Contraseña podremos utilizar cualquiera siempre y cuando la contraseña este formada entre 5 y 10 caracteres.

En la siguiente ilustración podemos observar un inicio de sesión. Este usuario se encuentra por defecto en la base de datos inicial por lo que si se desea podemos saltarnos el paso de registrar un usuario. La contraseña es 12345.

Entrar

Registrarse

¿Por qué?

Email:

amg0098@alu.ubu.es

Contraseña:

•••••

Entrar

Cerrar

Figura D.19: Entramos en QuickTest

- Una vez realizado el registro y su posterior inicio de sesión el sistema informará con una ventana de bienvenida.

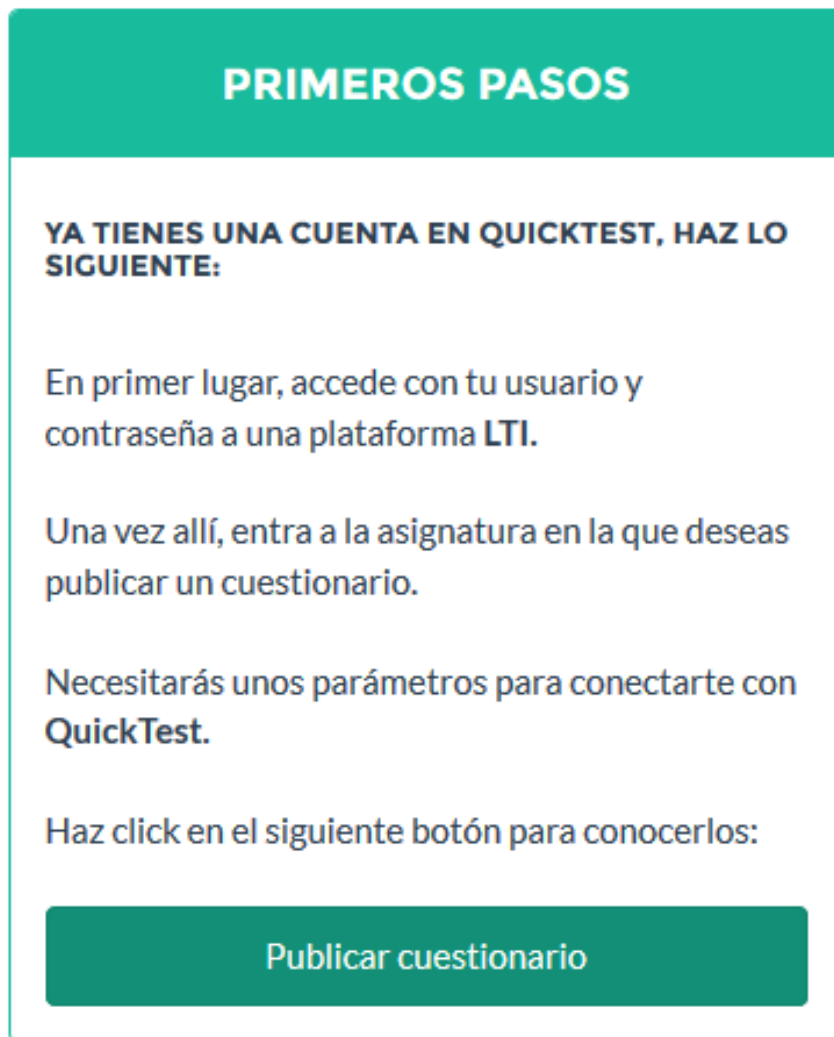


Figura D.20: Panel de primeros pasos

- Al pinchar sobre Publicar cuestionario se abrirá una ventana emergente. Es en esta ventana donde se encuentra los tres campos necesarios para poder publicar y enlazar nuestros cuestionarios con Moodle. La información necesarios es la siguiente.

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN41

PARA PUBLICAR, SIGUE ESTOS PASOS: ×

Crea una actividad desde tu plataforma LTI de tipo: **ACTIVIDAD EXTERNA**

Y copia estos datos en las casillas correspondientes

URL de inicio	Clave de cliente	Shared secret
localhost/_QuickTest_TFG/index.php	drQ\$1mF	Tu contraseña de QuickTest

IMPORTANTE:

En la pestaña 'privacidad', no olvides marcar:

☒ Aceptar calificaciones desde la herramienta
☒ Compartir el e-mail del usuario con la herramienta
☒ Aceptar calificaciones desde la herramienta

Close

Figura D.21: Ventana emergente de publicar un cuestionario

Configurar/Publicar un cuestionario en Moodle

En esta sección vamos a configurar Moodle para poder utilizar QuickTest. Para llevar a cabo esta tarea tendremos que seguir los siguientes pasos.

Configuración

- Abrimos una nueva pestaña del navegador y accedemos a Moodle.
- Una vez allí tendremos que iniciar sesión con el nombre de usuario y contraseña propios del administrador del sistema, en este caso son las credenciales utilizadas en la sección [D.3](#).

UBUVirtual TFG

☐ Recordar nombre de usuario

[¿Olvidó su nombre de usuario o contraseña?](#)

Las 'Cookies' deben estar habilitadas en su navegador



Algunos cursos permiten el acceso de invitados

Figura D.22: Inicio de sesión de Moodle

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN⁴²

- Una vez iniciada la sesión tendremos que crear un curso donde añadir nuestro cuestionario, para ello tendremos que:
 - Entramos en Administración del sitio.
 - Una vez allí entramos en la categoría de Cursos y dentro de esa sección en Administrar cursos y categorías.

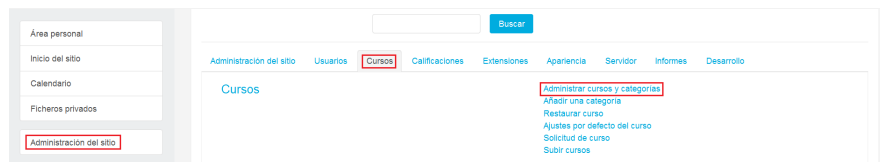


Figura D.23: Creamos un curso

- Una vez en esa sección seleccionamos Crear un nuevo curso y rellenamos los campos con la siguiente información. Al finalizar guardamos la información.

Crear un nuevo curso

▼ General

Nombre completo del curso	*	?	TRABAJO FIN DE GRADO
Nombre corto del curso	*	?	TFG

Figura D.24: Rellenamos los campos del nuevo curso

- Al crear el curso tendremos que matricular a los usuarios. Para ello vamos a crear dos usuarios.
 - Tendremos que entrar en Administración del sitio, una vez allí tendremos que ir a la categoría de Usuarios y dentro de allí a la sección Crear un nuevo usuario.

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN⁴³

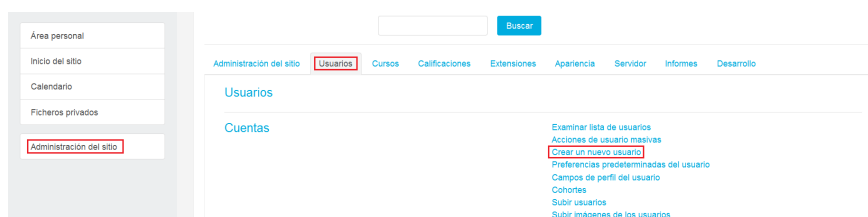


Figura D.25: Creamos los nuevos usuarios

- A continuación, tendremos que crear dos usuarios como podemos ver en las siguientes ilustraciones.

Nombre de usuario ?

Escoger un método de identificación: ?

☐ Cuenta de usuario suspendida ?

☐ Generar contraseña y notificar al usuario

La contraseña debería tener al menos 8 caracter(es), al menos 1 dígito(s), al menos 1 minúscula(s), al menos 1 mayúscula(s), al menos 1 caracter(es) no alfanuméricos como *, -, o #

Nueva contraseña ? 🔍

☐ Forzar cambio de contraseña ?

Nombre *

Apellido(s) *

Dirección de correo *

Figura D.26: Rellenamos los campos del primer usuario

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN44

The screenshot shows a web form for creating a new user. The fields and their values are as follows:

- Nombre de usuario:** profesor
- Escoger un método de identificación:** Cuentas manuales (selected from a dropdown)
- ☐ Cuenta de usuario suspendida
- ☐ Generar contraseña y notificar al usuario
- La contraseña debería tener al menos 8 caracter(es), al menos 1 dígito(s), al menos 1 minúscula(s), al menos 1 mayúscula(s), al menos 1 caracter(es) no alfanuméricos como *,-, o #**
- Nueva contraseña:** Asdf1234! (with a strength indicator and a search icon)
- ☐ Forzar cambio de contraseña
- Nombre:** Profesor
- Apellido(s):** Profesor Ape
- Dirección de correo:** profesor@ubu.es

Figura D.27: Rellenamos los campos del segundo usuario

- Una vez los usuarios tendremos que matricular dichos usuarios en el curso creado para ello tendremos que entrar en Administración del sitio, una vez allí tendremos que ir a la categoría de Cursos y dentro de allí a la sección Administrar cursos y categorías.
- Allí seleccionamos el nuevo curso creado y pinchamos en Usuarios matriculados.
- A continuación, matriculamos ambos usuarios creados anteriormente, el alumno con rol de Estudiante y el profesor con el rol de Profesor.

Publicar

Para llevar a cabo la publicación de un cuestionario tendremos que:

- Tendremos que entrar en Administración del sitio, una vez allí tendremos que ir a la categoría de Cursos y dentro de allí a la sección Administrar cursos y categorías.
- Una vez allí, seleccionamos el curso creado y en Vista.
- A continuación, activamos la edición y añadimos una herramienta externa.
 - Para ello, tendremos que seleccionar Agregue una actividad o recurso.



Figura D.28: Agregamos una nueva herramienta externa

- En este instante comenzamos a configurar la herramienta externa para poder utilizar QuickTest. Tendremos que rellenar los campos con la siguiente información.

Nombre de la actividad: Enlace de prueba a QuickTest

Descripción de la actividad: IdCuestionario=1

Figura D.29: Configuramos la nueva herramienta externa 1

Tipo de herramienta externa: Automático, basado en la URL de inicio

URL de inicio: localhost/_QuickTest_TFG/index.php

Secure tool URL:

Iniciar el contenedor: Nueva ventana

Clave de cliente: drQS1mF

Shared secret: 12345

Parámetros personalizados: IdCuestionario=1

Figura D.30: Configuramos la nueva herramienta externa 2

- Además, revisar que en el apartado Privacidad se encuentren activas las siguientes opciones.

▼ [Privacidad](#)

- ☒ Compartir el nombre del usuario con la herramienta
- ☒ Compartir el e-mail del usuario con la herramienta
- ☒ Aceptar calificaciones desde la herramienta

Figura D.31: Campos de la privacidad

- En este instante la publicación ha finalizado.

Configuración del Webservice Moodle

Antes de comenzar a la explicación de los distintos pasos a realizar, tendremos que encontrarnos logeado con los credenciales del administrador del sitio.

- Una vez logeado nos dirigimos a Administración del sitio. Allí nos seleccionamos la categoría Extensiones y dentro de esta categoría en Servicios Web.
- Si hemos realizado el punto anterior de manera correcta, nos encontramos en panel central de la configuración del Web Service. Este entorno de configuración se puede observar en la siguiente ilustración.

Pasos	Estado	Descripción
1. Habilitar Servicios Web	No	Los servicios Web deben estar habilitados en las características avanzadas.
2. Habilitar los protocolos	Ninguno	Al menos un protocolo debe estar habilitado. Por razones de seguridad, los protocolos que se van a utilizar deben estar habilitados.
3. Crear un usuario específico		Se necesita un usuario de servicios web para representar el sistema que controla Moodle.
4. Comprobar privilegios del usuario		El usuario debe tener las capacidades (privilegios) apropiadas de acuerdo al protocolo empleado, por ejemplo webservice/rest:use, webservice/soap:use. Para lograr esto, cree un rol de servicios web con las capacidades del protocolo habilitadas y asígnelo a usuario de servicios web como un rol del sistema.
5. Seleccione un servicio		Un servicio es un conjunto de funciones de 'servicios web'. Permitirá acceder al usuario a un nuevo servicio. En la página "Añadir servicio" marque las opciones 'Habilitado' y 'Usuario autorizado'. Seleccione 'No se requiere permiso'.
6. Agregar funciones		Seleccione las funciones requeridas para el servicio de nueva creación.
7. Seleccione un usuario específico		Agregue el usuario de servicios web como un usuario autorizado.
8. Crear ficha para un usuario		Crear ficha para usuario de servicios web.
9. Habilitar documentación del desarrollador	No	Está disponible para los protocolos habilitados una documentación detallada sobre los servicios de la web.
10. Comprobar el servicio		Simula el acceso externo al servicio usando el cliente de prueba de servicio web. Use un protocolo permitido con autenticación por ficha (token). ADVERTENCIA: ¡Las funciones que usted pruebe SERÁN EJECUTADAS para este usuario, por lo que debe ser cauteloso con lo que decida probar!

Figura D.32: Panel de configuración del Web Service de Moodle

- Para llevar a cabo la correcta configuración vamos a realizar los pasos de manera ordenada de menor a mayor como se puede ver en la anterior ilustración.
 - **Habilitar los Servicios Web:** para ello entramos en el paso 1, y activamos el checkbox y guardamos los cambios.
 - **Administrar protocolos:** para poder interactuar con el Web Service tendremos que habilitar al menos uno de ellos. En nuestro caso utilizaremos REST al ser el mas común utilizado en la comunidad de los Web Service de Moodle. Además, permite mostrar el contenido de retorno en formato JSON, el cual es muy ventajoso para el el tratamiento de los datos en la aplicación Android. Por último, es aconsejable habilitar la documentación al ser un gran material de apoyo y guardamos los cambios.

- Como se ha comentado anteriormente, vamos a realizar la configuración de manera ordenada. No obstante, en este ejemplo vamos a ignorar los pasos 3, 4, 5 y 6. En estos pasos, se crea un usuario específico para interactuar con el Web Service, este usuario debe de tener ciertos privilegios para poder desempeñar esta funcionalidad, se crea un servicio y se agregan las funcionalidades de interés a dicho servicio. Estos pasos son ignorados, ya que se va a utilizar el propio usuario administrador del sitio al tener todos los privilegios necesarios y de esta manera no limitaremos las funcionalidades del Web Service al incluir todas ellas. Destacar que no es recomendable utilizar el mismo usuario administrador de Moodle por motivos de seguridad, para interactuar con el Web Service. Al ser un trabajo meramente didáctico, utilizaremos este mismo usuario para evitar complicaciones. Por último, estos pasos omitidos se encuentran explicados en el siguiente enlace: <http://www.vitalinnova.com/servicios-web-en-moodle-2-0/>
- **Seleccione un servicio:** para poder habilitar el servicio externo tendremos que seleccionar el paso 5. Una vez allí, click sobre editar en servicio *Moodle mobile web service* y activamos el checkbox.

La configuración del Web Service ha finalizado. Para poder comprobar el correcto funcionamiento del mismo, vamos a pedir a este, el token del administrador. Para ello abrimos un nuevo navegador e introducimos la siguiente url: http://localhost/moodle/login/token.php?username=admin&password=Asdf1234!&service=moodle_mobile_app. Si todo ha ido correctamente la respuesta será la siguiente:

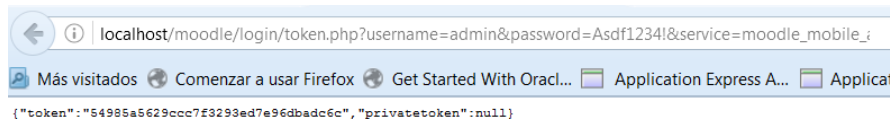


Figura D.33: Obtención del token del administrador

D.4. Compilación, instalación y ejecución del proyecto

D.5. Pruebas del sistema

Documentación de usuario

- E.1. Introducción**
- E.2. Requisitos de usuarios**
- E.3. Instalación**
- E.4. Manual del usuario**

Bibliografía

- [1] CakePhp. Cakephp — cakephp cookbook. [Internet; descargado 29-marzo-2015].
- [2] Google. Android — android. guía de desarrollo de aplicaciones para smartphones y tabletas - 2ª edición. [Internet; descargado 2-marzo-2017].
- [3] Google. Licencia comercial — app licensing. [Internet; descargado 2-marzo-2017].
- [4] Laravel. Laravel — love beautiful code? we do too. [Internet; descargado 29-marzo-2015].
- [5] Lavanguardia. Moodle — el australiano que cambió la forma de aprender en todo el mundo. [Internet; descargado 2-marzo-2017].
- [6] Moodle. Moodle — moodle statistics. [Internet; descargado 2-marzo-2017].
- [7] PHP. Php — referencia del lenguaje. [Internet; descargado 29-marzo-2015].
- [8] Slim. Slim — slim is a php micro framework. [Internet; descargado 29-marzo-2015].
- [9] Symfony. Symfony — the platform that powers work. [Internet; descargado 29-marzo-2015].
- [10] Wikijuegos. Licencia comercial — definición. [Internet; descargado 2-marzo-2017].
- [11] Wikipedia. Android studio — wikipedia, la enciclopedia libres. [Internet; descargado 29-marzo-2015].

- [12] Wikipedia. Moodle — wikipedia, la enciclopedia libres. [Internet; descargado 29-marzo-2015].
- [13] Wikipedia. Tortoiseshn — wikipedia, la enciclopedia libre. [Internet; descargado 2-marzo-2017].
- [14] Wikipedia. Xampp — wikipedia, la enciclopedia libre. [Internet; descargado 2-marzo-2017].