

---

# exerClick

Aplicación móvil para el seguimiento de ejercicios en el aula

---

Septiembre de 2015

Autor:

Adrián Núñez Marcos

Directora:

Maite Urretavizcaya Loinaz

eman ta zabal zazu



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea

informatika  
fakultatea



facultad de  
informática



*Dedicado a...*

Introducir los agradecimientos aquí.



# Resumen

Proyecto de Fin de Grado de la especialidad de Computación. Se ha implementado la aplicación para el seguimiento de ejercicios en el aula llamada **exerClick**. Es una aplicación multiplataforma para móviles, evaluada en Android y en iOS y adaptada a esos sistemas gracias a la plataforma Cordova. En su implementación se han utilizado las tecnologías web HTML5, CSS3 y Javascript, además de PHP para el servidor.

La aplicación permite que los profesores añadan ejercicios y los alumnos le envíen *feedback* sobre su realización mediante dos opciones: marcar una duda en el ejercicio o marcar el ejercicio como finalizado.



# Laburpena

Konputazioko espezialitaterako Gradu Amaierako Proiektua. **exerClick** deituriko aplikazio bat inplementatu da klaseko ariketen jarraipena egiteko. Plataforma anitzeko mugikorretarako aplikazioa da, Android eta iOS-en balioztatua eta sistema hoietara egokitua Cordova plataformari esker. Honen inplementazioan HTML5, CSS3 eta Javascript web teknologiak erabili dira, zerbitzarirako PHPaz gain.

Aplikazioak irakasleari ariketak gehitzea ahalbidetzen dio eta ikasleek ariketa horiei dagokien *feedback*-a bidaltzea bi aukeren bitartez: ariketa bat zalantza bezala edo amaitutzat bezala markatu.





# Abstract

Computer Science speciality's Bachelor Thesis. The implemented application, called **exerClick**, aims to monitorize the realization of exercises in classrooms. It is a multi-platform application for mobile phones, adapted to Android and iOS using the Cordova platform. In its implementation have been used HTML5, CSS3 and Javascript web technologies, in addition to PHP for the server.

The application lets the teacher to add exercises and students to send feedback for each exercise using two options: set a question for the exercise or mark the exercise as finished.



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Contexto . . . . .	1
1.2. Propuesta . . . . .	2
1.3. Organización del documento . . . . .	2
<b>2. Documento de objetivos del proyecto (DOP)</b>	<b>4</b>
2.1. Alcance del proyecto . . . . .	4
2.2. Exclusiones del proyecto . . . . .	5
2.3. Fases y tareas del proyecto . . . . .	5
2.3.1. Estudio de alternativas para la creación de una aplicación móvil .	5
2.3.2. Formación . . . . .	6
2.3.3. Documentación del proyecto . . . . .	6
2.3.4. Presentación y Defensa del proyecto . . . . .	6
2.4. Análisis de riesgos . . . . .	6
2.5. Análisis de factibilidad . . . . .	6
<b>3. Gestión del proyecto</b>	<b>7</b>
3.1. Metodología Ágil . . . . .	7
3.2. Metodología InterMod . . . . .	8
3.2.1. Adaptación de InterMod . . . . .	9
3.3. Formación de equipos . . . . .	10
3.4. Canal de comunicación e Infraestructura de almacenamiento . . . . .	10
3.5. Sistemas Operativos evaluados . . . . .	10
3.6. Tecnologías usadas . . . . .	11
3.7. Step 0 - Análisis del proyecto . . . . .	12
3.7.1. UOs inicialmente planteados . . . . .	12
3.8. Iteración 1 . . . . .	13
3.8.1. Step 1.1. Lista de UOs . . . . .	13
3.8.2. Step 2.1. Planificación de la iteración . . . . .	14
3.8.3. Step 3.1. Ejecución de las actividades . . . . .	14
3.9. Iteración 2 . . . . .	14
3.9.1. Step 1.2. Lista de UOs . . . . .	14
3.9.2. Step 2.2. Planificación de la iteración . . . . .	15
3.9.3. Step 3.2. Ejecución de las actividades . . . . .	15

3.10. Iteración 3 . . . . .	15
3.10.1. Step 1.3. Lista de UOs . . . . .	15
3.10.2. Step 2.3. Planificación de la iteración . . . . .	15
3.10.3. Step 3.3. Ejecución de las actividades . . . . .	15
3.11. Iteración 4 . . . . .	16
3.11.1. Step 1.4. Lista de UOs . . . . .	16
3.11.2. Step 2.4. Planificación de la iteración . . . . .	16
3.11.3. Step 3.4. Ejecución de las actividades . . . . .	16
3.12. Iteración 5 . . . . .	16
3.12.1. Step 1.5. Lista de UOs . . . . .	16
3.12.2. Step 2.5. Planificación de la iteración . . . . .	16
3.12.3. Step 3.5. Ejecución de las actividades . . . . .	17
3.13. Iteración 6 . . . . .	17
3.13.1. Step 1.6. Lista de UOs . . . . .	18
3.13.2. Step 2.6. Planificación de la iteración . . . . .	18
3.13.3. Step 3.6. Ejecución de las actividades . . . . .	18
3.14. Documentación de la gestión del proyecto . . . . .	18
<b>4. Análisis de Requisitos . . . . .</b>	<b>19</b>
4.1. Requisitos no-funcionales . . . . .	19
4.2. Requisitos funcionales . . . . .	19
4.2.1. UO1-S: Responder a un ejercicio . . . . .	19
4.2.2. UO1-T: Crear-Lanzar un ejercicio simple . . . . .	19
4.2.3. UO2-T: Crear un ejercicio detallado . . . . .	19
4.2.4. UO3-T: Cambiar el tipo de ejercicio . . . . .	19
4.2.5. UO4-T: Ver estadísticas de un ejercicio . . . . .	20
4.2.6. UO5-T: Ver la descripción completa de un ejercicio . . . . .	20
4.2.7. UO6-T: Editar un ejercicio . . . . .	20
4.2.8. UO7-T: Cerrar sesión . . . . .	20
4.2.9. UO8-T: Cambiar el idioma de la aplicación . . . . .	20
4.2.10. UO9-T: Cambiar de asignatura . . . . .	20
<b>5. Diseño e Implementación . . . . .</b>	<b>21</b>
5.1. Estructura del proyecto . . . . .	21
5.2. Interfaces o lado del cliente . . . . .	21
5.3. Lógica de negocio o lado del servidor . . . . .	21
5.3.1. Base de datos . . . . .	21
<b>6. Conclusiones y Líneas futuras . . . . .</b>	<b>22</b>
6.1. Objetivos alcanzados . . . . .	22
6.2. Lecciones aprendidas . . . . .	22

6.3. Líneas futuras y Propuestas de mejora . . . . .	22
<b>Bibliografía y referencias</b>	<b>22</b>
<b>A. Actas de Reunión</b>	<b>24</b>

# Índice de figuras

# Índice de cuadros





# 1. Introducción

## 1.1. Contexto

Muchas veces nos encontramos con aulas con demasiados alumnos. Estas clases son especialmente comunes en los primeros cursos de los estudios universitarios, donde el número de alumnos alcanza fácilmente las tres cifras. Responder a todas las preguntas y dudas, monitorizar a los alumnos, asegurar que todos o la mayoría hayan acabado el ejercicio, etc. con esa cantidad de alumnos no es posible para una sola persona. Ante estas situaciones la mayoría de veces suelen ignorarse problemas y dudas y seguir adelante.

Los nuevos planes de estudio que pretenden dejar atrás el sistema de educación mediante clases magistrales y dinamizar las clases han supuesto un aumento en el número de clases prácticas y laboratorios que se realizan. Algunos centros han optado incluso por dividir las clases en grupos más pequeños para realizar las prácticas, pero muchas veces ésto no es posible. En esas situaciones el profesor acaba por no poder monitorizar completamente la clase.

Con el fin de tener un medio común se han implantado en los últimos años nuevas tecnologías en entornos docentes. Sin embargo, en muchos casos esta tecnología se limita a entornos de apoyo a la docencia más que al alumnado, siendo muy popular el sistema de gestión del aprendizaje Moodle. Además, el uso más frecuente de estos sistemas es el de simple almacén de recursos bibliográficos (enlaces, apuntes, transparencias, etc.).

Por otro lado, la expansión de las tecnologías móviles y tabletas, con las que los alumnos están cada vez más familiarizados, no ha sido aprovechada. Estas tecnologías están ya mayoritariamente presentes en las aulas, la mayoría del alumnado dispone de alguno de estos dispositivos, pero su uso como herramienta educativa no es real, desperdiciando así todo su potencial como sistema de ayuda al aprendizaje. Es más, muchas veces el uso de estos dispositivos está prohibido o limitado en clase.

## 1.2. Propuesta

Nuestra propuesta pretende modificar y actualizar los modelos educativos presenciales a través de herramientas que faciliten la captura de la información de lo que sucede en el aula, con el objetivo de proporcionar *feedback* a profesores y estudiantes sobre su progreso en el aprendizaje. Esta propuesta se materializa en la aplicación PresenceClick que facilita la captura colaborativa de esta información entre alumnos y profesores de manera ágil. PresenceClick actualmente dispone de una serie de módulos que capturan la asistencia de los alumnos a clase de manera automática, sus sensaciones sobre las diversas actividades de aprendizaje, sus respuestas a preguntas al aire del profesor y sus dudas. En particular, nuestro objetivo en este nuevo proyecto es capturar las interacciones entre profesor-alumnos en sesiones de ejercicios.

El profesor dispondrá de una interfaz a la que accederá mediante su dispositivo móvil o tableta en clase e indicará a sus alumnos los ejercicios a realizar (esta actividad se podrá realizar previa a la clase). Por su parte, los alumnos con sus dispositivos móviles (smartphones o tabletas) recibirán las notificaciones de los ejercicios a realizar y podrán indicar para cada uno si tienen dudas en su realización o si lo han terminado. Se podrá disponer en tiempo real de información sobre el porcentaje de alumnos que lo han realizado, alumnos que indican problemas en su realización y aquellos alumnos que no indican nada. Es decir, el profesor tendrá idea en tiempo real de quiénes y cuántos han realizado el ejercicio y podrá acercarse a comprobar y revisar sus soluciones. Además el profesor podrá valorar su nivel de corrección o satisfacción en la realización, añadiendo las notas oportunas en el sistema que le permitirá seguir la evolución de cada uno de sus alumnos durante el curso. También podrá acercarse a aquellos que señalan problemas en su realización, con el fin de ayudarlos y evitar dificultades en su progreso.

Bajo este contexto surge **exerClick**, la herramienta para seguimiento de ejercicios en el aula. Esta herramienta, con todas sus funcionalidades, nace con el propósito de tener una visión más real de los alumnos, tanto en grupo como individualmente y está dirigida a profesores y a los propios alumnos. De esta manera, el docente puede ofrecer un aprendizaje más adaptado e individualizado, aunque los grupos de alumnos sean muy grandes.

## 1.3. Organización del documento

En este documento se describe principalmente la gestión y el seguimiento realizado durante el desarrollo de **exerClick**.

Introducir organización.



## 2. Documento de objetivos del proyecto (DOP)

Durante las sesiones prácticas de ejercicios en el aula a menudo el docente se pregunta si todo el mundo ha acabado los ejercicios propuestos con la esperanza de poder continuar con otro ejercicio o dando las explicaciones necesarias para poder continuar. Se puede preguntar si sus alumnos tienen vergüenza de plantear dudas o si pasan de ello, si no estará consumiendo demasiado tiempo en este ejercicio, etc. Con un conjunto de 100 alumnos, por ejemplo, resulta imposible observar el avance de los alumnos en los ejercicios y atender a todas las dudas que surgen. Incluso la estrategia de dividir la clase en grupos más pequeños para las clases prácticas, que puede parecer una buena idea, acaba consumiendo más tiempo, al tener que repetir lo mismo varias veces.

Lo ideal sería que existiera un medio para permitir a los alumnos dejar claro el estado en el que se encuentran, que el profesor pudiera ver como va cada alumno y que si hubiera dudas quedaran en algún sitio almacenadas para no dejar a ningún alumno sin su respuesta.

Con estos objetivos nace *exerClick*, que pretender ayudar a alumnos y docentes a llevar a cabo esta tarea. En este capítulo describiremos el alcance del proyecto.

### 2.1. Alcance del proyecto

Dividiremos los objetivos en objetivos del alumno y objetivos del profesor. Estos son los objetivos fijados para el alumno:

- Que pueda ver en todo momento cuales son los ejercicios propuestos en clase.
- Que un alumno pueda dejar constancia del estado en el que se encuentra durante un ejercicio: si lo ha terminado, si tiene dudas o si está realizándolo.
- Ver su progreso y el de sus compañeros en la realización de ejercicios.

Los objetivos del docente serán los siguientes:

- Que pueda ver en todo momento cuales son los ejercicios que va proponiendo en clase, los que se han terminado durante la sesión presente o los que aún no se han propuesto pero están preparados.
- Crear en cualquier momento un nuevo ejercicio (rápidamente o bien preparándolo tranquilamente). Y una vez creado tener la posibilidad de proponerlo a la clase o guardarlo para proponerlo más tarde.
- Ver por cada ejercicio el estado de los alumnos: quiénes lo han terminado, quiénes tienen duda y quiénes no han respondido nada.
- Editar cualquier detalle de un ejercicio en cualquier momento.
- Valorar la realización de un ejercicio a un alumno concreto.

Además, se han fijado dentro del alcance los siguientes requisitos:

- Contar con la opinión de usuarios finales (alumnos y docentes) durante el desarrollo de la aplicación, asegurando su aceptación general.
- La internacionalización de la aplicación, que tendrá 4 idiomas disponibles: castellano, euskera, inglés y francés.
- Desarrollar la aplicación para que funcione en el mayor número de dispositivos posibles: smartphones, tablets, etc., teniendo en cuenta los diferentes tamaños de pantalla.
- La aplicación será multiplataforma, pudiendo funcionar en Android, iOS y Windows Phone.

## 2.2. Exclusiones del proyecto

- Se van a excluir del proyecto todas las funciones que se puedan desarrollar en PresenceClick ligadas a exerClick (es decir, sobre ejercicios en el aula, como pueden ser estadísticas más trabajadas).

## 2.3. Fases y tareas del proyecto

### 2.3.1. Estudio de alternativas para la creación de una aplicación móvil

Añadir porque se ha escogido Cordova en lugar de Phonegap/Ionic o una aplicación nativa.

### 2.3.2. Formación

Añadir detalles sobre como se ha aprendido sobre tecnologías web, responsive, cordova, etc.

### 2.3.3. Documentación del proyecto

Añadir sobre la documentación algo (como se ha ido desarrollando y tal).

### 2.3.4. Presentación y Defensa del proyecto

Añadir como ha ido el tema de la presentación y preparación de la defensa.

## 2.4. Análisis de riesgos

Análisis de riesgos.

## 2.5. Análisis de factibilidad

Análisis de factibilidades.

## 3. Gestión del proyecto

Añadir introducción.

### 3.1. Metodología Ágil

Las metodologías ágiles son procesos para el desarrollo de software. Se basan en el desarrollo incremental e iterativo, donde varios grupos se unen para dar requerimientos y soluciones nuevas gracias a su colaboración. Se enfatiza en el cara a cara más que en la documentación. Un ejemplo de esta metodología es SCRUM.

La metodología es bastante flexible, pudiendo adaptarse a cada proyecto. Es decir, no tiene reglas absolutamente fijas. En su base esta el *Manifiesto Ágil* en el que se basan todas las variantes de esta metodología. Este manifiesto [4] dice así:

- Individuos e interacciones sobre procesos y herramientas
- Software funcionando sobre documentación extensiva
- Colaboración con el cliente sobre negociación contractual
- Respuesta ante el cambio sobre seguir un plan

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.

Al comienzo del proyecto se fijó el uso de esta metodología, con 4 grupos trabajando en paralelo (los equipos vienen definidos en el apartado 3.3). Sin embargo, las iteraciones acabaron volviéndose excesivamente largas, habiendo varios grupos parados. Si bien se respetó la filosofía de la metodología ágil en cierto sentido, y aun con cierta flexibilidad en su adaptación, al final del proyecto acabó por volverse algo parecido a una metodología ágil sin llegar a serlo.

## 3.2. Metodología InterMod

InterMod es una metodología de trabajo, que será utilizada para este proyecto, desarrollada por el grupo de investigación GaLan de la Facultad de Informática de San Sebastián. Se trata una metodología ágil de desarrollo de alta calidad de software interactivo, incluyendo aplicaciones web.

En InterMod se define el Objetivo de Usuario (User Objective o UO) como el deseo del usuario que puede ser alcanzado mediante una o más funcionalidades. Diferentes UOs son desarrollados durante el proyecto y la unión de todos, en su globalidad, forma la aplicación final. Además, el mismo UO puede estar incluido en uno o más requerimientos funcionales y/o no-funcionales. Existen a su vez diferentes tipos de UO:

- **UO Directo:** Es un objetivo del usuario final.
- **UO Indirecto:** Surge a partir de otros UOs por necesidades interna del desarrollo (no son propiamente deseos del usuario). Aparecen durante el desarrollo debido a la fusión o división de otros UOs.
- **UO Reutilizable:** Es un UO creado y evaluado, total o parcialmente, en otro proyecto o en el proyecto actual que puede ser reutilizado.

Basándose en la propuesta del Object Management Group's Model Driven Architecture, Intermod establece sus actividades basadas en modelos.

Por cada actividad se desarrollan siempre dos fases: la creación del modelo (independiente de la plataforma) y su posterior evaluación. Las evaluaciones de usabilidad son especialmente útiles para los UOs Directos ya que reflejan una necesidad del usuario, por tanto es importante que un grupo de estos esté involucrado. Para agilizar el proyecto, algunos modelos pueden ser evaluados únicamente por expertos en usabilidad. Las actividades no se dan por acabadas y pueden continuar activas durante varias iteraciones hasta conseguir una evaluación positiva.

Existen dos tipos de actividades para el desarrollo de UOs: Actividades de Desarrollo (DA) y Actividades de Integración (IA). Existen tres tipos de DAs:

- **DA-1:** *Análisis y Diseño de la Navegación.*
- **DA-2:** *Construcción de la Interfaz.*
- **DA-3:** *Codificación de la Lógica de Negocio.*

Para asegurar el desarrollo incremental de la aplicación son necesarias las Actividades de Integración (IA). Existen tres tipos de IAs:



- **IA-1:** *Integración de los Modelos de Requerimientos.*
- **IA-2:** *Integración de la Interfaz.*
- **IA-3:** *Integración de la codificación y refactorización.*

Las actividades de desarrollo e integración de cada tipo da lugar a un modelo. Así, las DA-1 y IA-1, relativas al análisis de requisitos, desembocan en el modelo de requisitos (M-1); las DA-2 y IA-2, relativas a las interfaces, crean el modelo de presentación (M-2) y las DA-3 y IA-3, las asociadas a la lógica de negocio, constituyen el modelo de funcionalidad (M-3). El modelo M-1 es un modelo abstracto sobre el que se asientan las bases, basado en él se forma el M-2, que contiene todos los elementos gráficos y otras características definidos en el M-1. Finalmente el modelo M-3 establece la implementación en un lenguaje de programación concreto.

InterMod define una metodología dividida en iteraciones, y, a diferencia del resto, define un paso previo, Step 0. En esta etapa previa se realiza el análisis del proyecto y se definen los UOs iniciales del mismo y el diseño general. A continuación se pasa a la iteración 1, luego la 2, etc. y se continúa así hasta dar por finalizado el proyecto. Cada iteración esta dividida en 3 pasos:

- **Step 1.i:** Construir/Actualizar la lista de UOs.
- **Step 2.i:** Planificar las actividades para los diferentes equipos.
- **Step 3.i:** Realizar las actividades planificadas.

### 3.2.1. Adaptación de InterMod

Debido a las características del proyecto se ha decidido realizar algunas modificaciones al esquema de InterMod:

- Los UOs normalmente se denotan por UOX (siendo X el número del UO). En este proyecto hemos distinguido dos usuarios, por tanto hará falta un identificador extra para saber a que usuario corresponde el UO. Se usará la notación UOX-Y, siendo Y la inicial en inglés del tipo de usuario: 'S' para el estudiante (*Student*) y 'T' para el profesor (*Teacher*).
- Siguiendo la línea del punto anterior, se ha hecho un cambio en la notación típica de los modelos. De M-1(X), el primer modelo del UOX, a M-1(XY), el primer modelo del UOX-Y (siguiendo la notación del punto anterior).

### 3.3. Formación de equipos

Se han identificado 4 equipos que participarán en el proyecto (interesados en el proyecto):

- **Equipo 1:** Formado por el alumno, Adrián Núñez. Se encargará del diseño de las interfaces y de la implementación de la aplicación.
- **Equipo 2:** Realizara las evaluaciones pertinentes y estará formado por alumnos de la facultad.
- **Equipo 3:** Segundo equipo para las evaluaciones, estará formado por miembros del grupo GaLan.
- **Equipo 4:** Se encargará de las evaluaciones con los usuarios finales.

### 3.4. Canales de comunicación e Infraestructura de almacenamiento

Con el fin de mantener la comunicación con los interesados en el proyecto se plantea el canal de comunicación *Slack* (<https://slack.com/>) al inicio del proyecto. Mediante este canal se intercambiaran mensajes rápidos entre y Adrián Núñez para cualquier duda u opinión rápida.

También se realizaran periódicamente reuniones presenciales, como mínimo, entre Maite Urretavizcaya y Adrián Núñez para monitorizar el desarrollo del proyecto. A estas reuniones también han asistido Samara Ruíz (en su mayoría) y Juan Miguel López.

Para la comunicación con el Equipo 2 se utilizó la herramienta de mensajería móvil *Telegram* (que permite, al contrario que otras conocidas, el intercambio de cualquier tipo de fichero, vital para pasarles el fichero .apk).

Finalmente, el proyecto y este documento estarán públicos en *GitHub* en el siguiente repositorio: <https://github.com/AdrianNunez/exerClick>. Se realizaran periódicamente actualizaciones del mismo. También se dispone de otras copias en el ordenador personal del autor, Adrián Núñez, y en el espacio de almacenamiento *Google Drive*, en una carpeta compartida con Maite Urretavizcaya y Samara Ruíz.

## 3.5. Sistemas Operativos evaluados

En la propia documentación de Apache Cordova vienen indicados los Sistemas Operativos soportados a los que se puede adaptar la aplicación:

- iOS (Mac)
- Amazon Fire OS (Mac, Linux, Windows)
- Android (Mac, Linux, Windows)
- BlackBerry 10 (Mac, Linux, Windows)
- Windows Phone 8 (Windows)
- Windows (Windows)
- Firefox OS (Mac, Linux, Windows)

Ya que el abanico de posibilidades es extenso se decidió reducir las posibilidades a iOS, Android, BlackBerry 10 y Windows Phone 8. El desarrollo principal y todas las pruebas iniciales se realizarán utilizando Android.

Al final del proyecto, debido a incompatibilidades con la infraestructura disponible, se realizaron sólo evaluaciones en dispositivos físicos Android e iOS.

## 3.6. Tecnologías y Herramientas utilizadas

Al inicio, antes del desarrollo de la aplicación, se concretaron las siguientes tecnologías mínimas a utilizar:

- HTML5
- CSS3
- PHP
- MySQL

Como calidad añadida se utilizan las siguientes tecnologías:

- Javascript

- JQuery

Las herramientas principales para el desarrollo de la aplicación serán Notepad++ y Eclipse. Para la gestión de la base de datos se usará phpMyAdmin y para la gestión del servidor se usará WinSCP.

Para el desarrollo en Android se utilizará la SDK de Android y el emulador Genymotion (mucho más rápido que el emulador por defecto).

## 3.7. Step 0 - Análisis del proyecto

**Fecha de inicio:** 9/10/14

**Fecha de fin:** 18/10/14

**Dedicación total:** 2 horas, 35 minutos

Esta fase previa al desarrollo de la aplicación principal se planteó dentro de la metodología InterMod como una introducción. En este proyecto se realizaron dos tareas principalmente:

- Documentarse sobre el estado del arte en el ámbito de aplicaciones de seguimiento de ejercicios en el aula, sobre bases de datos que representaran el concepto de ejercicio, etc.
- Realizar un *Brainstorming* para plantear todas las ideas posibles para la aplicación. De esta forma se pretende tener una idea más clara de como será la aplicación que se desea.

Finalmente se realizó la lista de UOs inicial que incluía muchos UOs no prioritarios. Sin embargo, se decidió dejar la lista completa e ir abordándola por prioridades y actualizándola iteración por iteración.

### 3.7.1. UOs inicialmente planteados

UOs del alumno

**UO1-S:** *Responder a un ejercicio.* El alumno quiere indicar que ha acabado o que tiene dudas con un ejercicio que el profesor ha propuesto.

## UOs del profesor

**UO1-T:** *Crear-lanzar un ejercicio simple.* El profesor quiere proponer un ejercicio rápidamente, sin escribir mucho.

**UO2-T:** *Crear-lanzar un ejercicio detallado.* El profesor quiere proponer la realización de un ejercicio preparado previamente o con bastantes detalles.

**UO3-T:** *Dar por finalizado un ejercicio propuesto activo.* El profesor quiere terminar con uno de los ejercicios que propuso.

**UO4-T:** *Ver estadísticas de un ejercicio.* El profesor desea ver qué tal le ha ido a la clase en general o a un alumno en un ejercicio.

**UO5-T:** *Ver la descripción completa de un ejercicio.* Un profesor quiere ver la descripción completa de un ejercicio (identificador, enunciado, página, tema, etc.).

**UO6-T:** *Editar un ejercicio.* El profesor desea editar los atributos de un ejercicio.

**UO7-T:** *Evaluar el ejercicio de un alumno.* El profesor quiere valorar la realización de un ejercicio a un alumno.

**UO8-T:** *Cerrar sesión.* El profesor quiere cerrar su sesión activa.

**UO9-T:** *Cambiar el idioma de la aplicación.* El profesor desea cambiar el idioma con el que lee la aplicación.

**UO10-T:** *Cambiar de asignatura.* El profesor, que tiene más de una asignatura, quiere cambiar de una asignatura x a otra asignatura y.

## 3.8. Iteración 1

Fecha de inicio: 21/10/14

Fecha de fin: 2/11/14

**Dedicación total:** 3 horas, 25 minutos

### 3.8.1. Step 1.1. Lista de UOs

Debido a la gran cantidad de UOs planteados se ha decidido seleccionar los considerados como prioritarios. En esta primera etapa se quieren desarrollar los que creemos que son los más importantes:

- UOs del alumno:
  - **UO1-S:** *Responder a un ejercicio.*
- UOs del profesor:
  - **UO1-T:** *Crear-lanzar un ejercicio simple.*
  - **UO2-T:** *Crear-lanzar un ejercicio detallado.*
  - **UO3-T:** *Dar por finalizado un ejercicio propuesto activo.*
  - **UO4-T:** *Ver estadísticas de un ejercicio.*

### 3.8.2. Step 2.1. Planificación de la iteración

En esta primera iteración el plan es el siguiente:

- Equipo 1: M-1 de todos los UOs iniciales
- Equipo 3: Evaluación M-1 de todos los UOs iniciales

### 3.8.3. Step 3.1. Ejecución de las actividades

Con la síntesis de lo acordado en la última reunión se modifica el documento de Brainstorming, que queda corregido para tener todas las ideas previas claras por escrito.

Se empiezan realizando los M-1 (prototipos de papel) de los UOs planteados inicialmente (de la lista entera). Los prototipos en papel se envían durante la iteración a Maite y Samara para valoraciones rápidas y posteriores correcciones.

## 3.9. Iteración 2

**Fecha de inicio:** 3/11/14

**Fecha de fin:** 10/11/14

**Dedicación total:** 3 horas, 30 minutos

### 3.9.1. Step 1.2. Lista de UOs

Lista de UOs = UO1-T, UO2-T, UO3-T, UO4-T

### 3.9.2. Step 2.2. Planificación de la iteración

Se continúa con el mismo plan de la pasada iteración.

- Equipo 1: M-1 de todos los UOs iniciales
- Equipo 3: Evaluación M-1 de todos los UOs iniciales

### 3.9.3. Step 3.2. Ejecución de las actividades

Se corrigen los prototipos en papel con lo acordado durante la pasada reunión y se realizan nuevos a limpio.

## 3.10. Iteración 3

**Fecha de inicio:** 11/11/14

**Fecha de fin:** 12/1/15

**Dedicación total:** 72 horas, 55 minutos

### 3.10.1. Step 1.3. Lista de UOs

Lista de UOs = UO1-T, UO2-T, UO3-T, UO4-T

### 3.10.2. Step 2.3. Planificación de la iteración

### 3.10.3. Step 3.3. Ejecución de las actividades

Durante esta iteración se instala el entorno de trabajo de Cordova. Se instalan inicialmente las plataformas Android y Windows Phone. Se decide también iniciar todas las pruebas utilizando Android.

Se comienza con el M-2 del UO1-T, que servirá como base para otros UOs (ya que casi todos comparten la pantalla madre que se desarrolla en esta fase).

El proyecto queda exportado a Eclipse para intentar generar el fichero .apk de modo que cualquiera pueda probar la aplicación, aunque no se consigue que este funcione.

Queda pendiente para una futura iteración.

También se fija que la versión de Android mínima para utilizar la aplicación sin problema es la 3.0.

## 3.11. Iteración 4

**Fecha de inicio:** 13/1/15

**Fecha de fin:** 26/1/15

**Dedicación total:** 16 horas

### 3.11.1. Step 1.4. Lista de UOs

Lista de UOs = UO1-T, UO2-T, UO3-T, UO4-T

### 3.11.2. Step 2.4. Planificación de la iteración

### 3.11.3. Step 3.4. Ejecución de las actividades

Se añaden las correcciones acordadas en la anterior reunión al M-2 del UO1-T y se inician en esta fase los M-2 de los UO4-T y UO1-S. También se sube el proyecto a GitHub para el control de versiones y tener una copia de seguridad extra.

Más avanzados en la iteración se le ha proporcionado al alumno la base de datos de PreseceClick vacía para poder empezar con la lógica de negocio o el M-3 de los UOs realizados. Se han creado dos tablas nuevas exclusivas para exerClick:

- **exercise:** Almacena información sobre un ejercicio.
- **exercisestate:** Almacena el estado de un ejercicio.

## 3.12. Iteración 5

**Fecha de inicio:** 27/1/15

**Fecha de fin:** 6/2/15

**Dedicación total:** 23 horas, 15 minutos

### 3.12.1. Step 1.5. Lista de UOs

Lista de UOs = UO1-S, UO1-T, UO2-T, UO3-T, UO4-T



### 3.12.2. Step 2.5. Planificación de la iteración

- Equipo 1: UO1-T[M-3], UO3-T[M-3], UO4-T[M-3], UO1-S[M-3]
- Equipo 3: EvaluaciónUO1-T[M-2], UO3-3[M-2], UO4-T[M-2]

### 3.12.3. Step 3.5. Ejecución de las actividades

Se empieza la iteración modificando el aspecto de la pantalla madre (correspondiente a casi todos los UOs del docente).

Para pasar la aplicación al móvil es necesario que el código PHP (de servidor) este en un servidor externo, de otro modo no funcionara. Se ha creado para ese propósito un servidor con el espacio que proporciona 000webhost (<http://www.000webhost.com/>) sin costo alguno para realizar pruebas al principio. Se ha instalado la base de datos proporcionada ahí mismo con las tablas originales más las dos creadas en la iteración anterior.

Se ha empezado realizando el M-3 del UO1-T, luego el UO4-T y finalmente el UO3-T. El objetivo de la iteración es tener una aplicación mínimamente funcional para realizar pruebas. Es decir, poder al menos realizar la función básica de que el profesor lance un ejercicio, el alumno lo vea y responda. Además, ver las estadísticas del ejercicio propuesto también sería importante.

Al abordar el UO3-T se han encontrado otros subobjetivos, derivados también de la reunión del día 27 de enero. En la reunión se decidió que los ejercicios, estuvieran en el estado que estuvieran (activos, finalizados o preparados para lanzarse), podrían cambiarse a cualquier estado. Es decir, el UO3-T tenía como objetivo el cambio de ejercicio activo a ejercicio finalizado, pero con esta decisión se deben de poder realizar cualquier otro cambio posible además del anterior.

**UO3-T:** *Cambiar el estado de un ejercicio.* El profesor quiere, por ejemplo, terminar con uno de los ejercicios que propuso o quiere volver a poner como activo un ejercicio que por error ha marcado como finalizado.

En esta iteración no se ha implementado la identificación de usuarios, por lo que sigue estando como en la iteración previa.

Se ha ocultado la pestaña de ejercicios preparados hasta que se aborde el tema.

## 3.13. Iteración 6

**Fecha de inicio:** 9/2/15

**Fecha de fin:** -

**Dedicación total:** - horas, - minutos

### 3.13.1. Step 1.6. Lista de UOs

Lista de UOs = UO1-S, UO5-T, UO6-T, UO7-T, UO9-T

### 3.13.2. Step 2.6. Planificación de la iteración

- Equipo 1: -
- Equipo 2: Evaluación de las .apks generadas durante el desarrollo
- Equipo 3: -

### 3.13.3. Step 3.6. Ejecución de las actividades

La iteración 6 se puede dividir en dos partes importantes: la primera en la que se crean los UOs restantes para tener una aplicación completamente funcional y la segunda en la que se genera finalmente un apk y se comienzan las pruebas.

Primera parte.

Para generar el apk se uso Eclipse Luna y Java 8. Se exportó el proyecto a Eclipse y se realizaron las configuraciones necesarias. Surgieron una gran cantidad de errores que supusieron la mayor parte del tiempo de esta parte. Finalmente, cuando todo funcionaba correctamente y la aplicación se ejecutaba en el emulador, Eclipse generaba un fichero .apk correcto. Este fichero le fue pasado a Maite Urretavizcaya, Samara Ruíz y al Equipo 2 para realizar pruebas.

## 3.14. Documentación de la gestión del proyecto

Añadir documentos sobre UOs: creación, tablas, esquemas, etc.

## 4. Análisis de Requisitos

### 4.1. Requisitos no-funcionales

### 4.2. Requisitos funcionales

En este apartado se presentan los requisitos funcionales recogidos en interfaces de papel

Siguiendo las pautas fijadas por los requisitos no-funcionales se han realizado interfaces sencillas e intuitivas. Se le ha dado mucha importancia a la filosofía de *en pocos clicks* que sigue la aplicación. Por tanto, ante todo, se han minimizado la cantidad de transiciones entre pantallas y el uso de excesivos botones para buscar mucha funcionalidad en pocos *clicks*.

#### 4.2.1. UO1-S: Responder a un ejercicio

Añadir pantallas, autómata de estados, info de seguimiento, etc.

#### 4.2.2. UO1-T: Crear-Lanzar un ejercicio simple

Añadir pantallas, autómata de estados, info de seguimiento, etc.

#### 4.2.3. UO2-T: Crear un ejercicio detallado

Añadir pantallas, autómata de estados, info de seguimiento, etc.

#### 4.2.4. UO3-T: Cambiar el tipo de ejercicio

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.5. UO4-T: Ver estadísticas de un ejercicio**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.6. UO5-T: Ver la descripción completa de un ejercicio**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.7. UO6-T: Editar un ejercicio**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.8. UO7-T: Cerrar sesión**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.9. UO8-T: Cambiar el idioma de la aplicación**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

**4.2.10. UO9-T: Cambiar de asignatura**

Añadir pantallas, autómata de estados, info de seguimiento, etc.

## 5. Diseño e Implementación

Añadir descripción del apartado.

### 5.1. Estructura del proyecto

Hablar sobre la estructura de ficheros del proyecto, sobre la estructura que sigue Cordova en los ficheros y sobre como está todo en eclipse.

### 5.2. Interfaces o lado del cliente

Hablar sobre la parte del cliente, las interfaces y el javascript.

### 5.3. Lógica de negocio o lado del servidor

Hablar sobre la parte de servidor y la base de datos.

#### 5.3.1. Base de datos

El diseño de la base de datos se ha realizado considerando desarrollos anteriores del grupo GaLan. Por ello muchos de los elementos son comunes con una de sus herramientas genérica de creación de sistemas docente, MAGADI, que ha sido proporcionada para este proyecto.

Se han incorporado unas tablas nuevas a la base de datos ya existente para la gestión de ejercicios.

## 6. Conclusiones y Líneas futuras

Añadir texto.

### 6.1. Objetivos alcanzados

De los objetivos iniciales indicar los alcanzados.

### 6.2. Lecciones aprendidas

Añadir lecciones aprendidas durante el proyecto.

### 6.3. Líneas futuras y Propuestas de mejora

Futuros mejoras, usos, etc. de la app.

# Bibliografía y referencias

- [1] Begoña Losada, Maite Urretavizcaya, Isabel Fdez. de Castro. *N\_InterMod*: Una Propuesta de notación de Diálogo enriquecida para el desarrollo ágil de aplicaciones interactivas. URL: [http://galan.ehu.es/Galan/sites/default/files/N\\_InterMod-18154-autores.pdf](http://galan.ehu.es/Galan/sites/default/files/N_InterMod-18154-autores.pdf) (visto 9-10-14)
- [2] The Apache Software Foundation. *Apache Cordova*. URL: <https://cordova.apache.org/> (visto 1-12-14)
- [3] Font Awesome. URL: <http://fontawesome.github.io/Font-Awesome/> (visto 1-12-14)
- [4] Manifiesto Ágil. URL: <http://www.agilemanifesto.org/iso/es> (visto 24-3-15)

# A. Actas de Reunión

## Reunión de Trabajo 1

**Fecha:** 8 de octubre de 2014

**Hora de inicio:** 12:30

**Hora de finalización:** 13:45

**Presentes:** Maite Urretavizcaya, Adrián Núñez

### **Temas tratados durante la reunión:**

- Inicio del proyecto: presentación del proyecto y de la metodología de trabajo a seguir durante el desarrollo del mismo.
- Acordar tareas a realizar antes de la siguiente reunión.

### **Resumen de la reunión:**

- Se presenta ExerClick: la aplicación web para gestión de ejercicios en el aula. Será una aplicación accesible desde dispositivos pequeños como el teléfono móvil de un alumno hasta dispositivos con pantallas más grandes como las de un ordenador que puede haber en el aula.
- La primera idea general de ExerClick es que sea una aplicación que puedan manejar tantos alumnos como profesores (dos roles definidos). En principio la idea es que los ejercicios se realicen dentro del aula. Los profesores podrán proponer ejercicios en la aplicación para que los alumnos los realicen. Los alumnos durante o después de la realización del ejercicio podrán responder a la propuesta del profesor: si lo han terminado o han tenido dudas, si están atascados, etc. De esa forma el profesor puede realizar un seguimiento más cercano, rápido y sencillo del alumnado.
- La filosofía de la aplicación es que tenga mucha funcionalidad pero en pocos Clicks.
- Se buscan 4 factores fundamentales en la aplicación:



- El uso de tecnologías actuales: HTML5, CSS3, PHP, MySQL y Symfony.
  - La simplicidad de la aplicación (en pocos clicks se deben de poder realizar muchas cosas).
  - Que sea internacional: internamente estará escrito en inglés (variables, comentarios, etc.) con el fin de que pueda llegar a ser código libre accesible a cualquiera. Además se quiere presentar en 4 idiomas (castellano, euskera, inglés y francés).
  - Uso de la tecnología Responsive Web Design (RWD). La aplicación se quiere adaptar a cualquier dispositivo.
- Presentación y explicación general sobre la metodología de desarrollo InterMod a utilizar. Intermod es una metodología ágil, basada en modelos y centrada en los usuarios.
  - Se identifica el equipo de trabajo:
    - Adrián Núñez, el alumno.
    - Maite Urretavizcaya, la profesora.
    - Juan Miguel López y Begoña Losada, parte del grupo GaLan, que actuarán como validadores.
    - Usuarios finales, tanto profesores como alumnos, que actuarán como validadores.

**Acordado para la siguiente reunión:**

- Inicio de la fase previa al desarrollo de la aplicación: recopilación de información sobre cualquier tema de interés para el proyecto.
- Generar ideas, prototipos, etc. para tener una visión más concreta del tipo de aplicación que se quiere hacer.

---

## Reunión de Trabajo 2

**Fecha:** 21 de octubre de 2014

**Hora de inicio:** 11:00

**Hora de finalización:** 12:25

**Presentes:** Maite Urretavizcaya, Samara Ruiz, Adrián Núñez

### **Temas tratados durante la reunión:**

- Presentación de las ideas pensadas en la fase inicial del proyecto.
- Partiendo de la discusión de las ideas las decisiones más importantes han sido:
  - Crear una aplicación para móviles en lugar de una aplicación web.
  - Iniciar el proyecto teniendo en cuenta que la aplicación sólo va a usarse dentro de clase.

### **Resumen de la reunión:**

- Se comienza explicando las ideas reunidas para la aplicación (en el documento de ideas acumuladas por parte del alumno durante las semanas anteriores), de forma que se tenga más claro el tipo de aplicación que se quiere realizar.
- Samara Ruiz se une a la reunión.
- Se comentan más ideas y unos bocetos iniciales. Comienza la discusión de la que salen las siguientes ideas:
  - En lugar de una aplicación web se realizará una aplicación para móviles.
  - Se limitará de momento el uso de la aplicación a cuando el docente y el alumnado estén en una sesión lectiva (excluyendo algunas funciones).
  - Se deja como pendiente un sistema de valoración de dificultad de los ejercicios por parte del alumno y de valoración del grado de satisfacción en la resolución de ejercicios de los alumnos por parte del profesor. Ambas usando el sistema que se usa en PresenceClick.
- Se proponen ideas para simplificar los bocetos iniciales. El fin es tener una versión más simple, con menos botones y pantallas (se pretende realizar todo en pocos clicks). Se proponen más ideas respecto a la interfaz y queda pendiente el realizar nuevos bocetos.

- Se da inicio a la iteración 1 del proyecto como una iteración larga con dos objetivos: responder a un ejercicio (por parte del alumno, UO-1S) y proponer un ejercicio (por parte del profesor, UO-1T). Se escoge como primer UO (Objetivo de Usuario) el de responder a ejercicios por parte del alumno (UO-1S). Los nuevos bocetos serán parte del prototipo en papel para el UO.
- Se comparte el documento de ideas general con Maite Urretavizcaya y Samara Ruiz.

**Acordado para la siguiente reunión:**

- Corregir el documento de ideas para el proyecto con lo decidido/propuesto en la reunión.
- Iniciar la iteración 1 con la fase de Análisis y Captura de requisitos. Enviar cuanto antes unos prototipos en papel resultado de esta fase para ser validados antes de la siguiente reunión.

---

## Reunión de Trabajo 3

**Fecha:** 3 de noviembre de 2014

**Hora de inicio:** 8:40

**Hora de finalización:** 10:30

**Presentes:** Maite Urretavizcaya, Samara Ruiz, Adrián Núñez

### **Temas tratados durante la reunión:**

- Discusión sobre los prototipos en papel desarrollados: definir las funcionalidades y diseño.

### **Resumen de la reunión:**

- Se decide separar el UO-1T en dos UOs: una cuando haya una sesión activa y otra cuando no. Las denotaremos como UO-1T y UO-2T.
- Se decide que el nombre del profesor/alumno no es tan importante y que es mejor añadir el nombre de la asignatura. En lugar del nombre aparecerá un icono que dará acceso al perfil del usuario donde aparecerán las siguientes opciones:
  - Ver el nombre del usuario.
  - Escoger la asignatura para el caso del profesor. Las asignaturas se ordenan por orden de proximidad horario (la siguiente clase será la primera de la lista).
  - Cerrar sesión.

Además, cuando un profesor inicia sesión, si tiene más de una asignatura y ninguna sesión actualmente, será redirigido al perfil para escoger alguna asignatura.

- Los ejercicios se identificarán por lo menos con un string identificativo (único para cada sesión, pero no único para una asignatura, pues identificadores como “1” pueden repetirse muchas veces a lo largo de una asignatura). Este string puede contener un sólo un número como al principio o cosas más detalladas como “1 - con ordenación”, “1 - con x”, “2.3”, etc.
- Se decide que una vez se inicia sesión, si no hay una clase en ese momento se irá abrirá la pantalla principal en la pestaña de ejercicios preparados. En caso de que haya clase se abrirá en la pestaña de ejercicios activos.
- Para los botones de lanzar se usará el icono del avión de papel.

- Los botones de editar/borrar ejercicios deben estar separados para que no haya ningún problema por darle sin querer a borrar.
- En lugar de que haya un botón para ver el ejercicio completo se harán los ejercicios clickables, de modo que al pinchar sobre un ejercicio te muestre los detalles del ejercicio (nos ahorramos un botón).
- Se piensa en un “Modo Ordenar” para poder ordenar manualmente la lista de los ejercicios activos o preparados para que aparezcan con un orden concreto. El objetivo es mostrar que ejercicios se desea que se realicen primero.
- En la lista de ejercicios, cada ejercicio tendrá un pequeño indicador de cuánta gente lo ha dado por acabado y cuanta tiene dudas en el ejercicio como vista previa.
- Un color para identificar el estado de los ejercicios, como primer acercamiento: rojo (activo), amarillo (preparado) y azul (finalizado).
- Al crear un nuevo ejercicio si su identificador ya existe tenemos dos posibilidades, queda pendiente ver cuál es la mejor:
  - Añadir al final del identificador un número como en el caso: 1 que pasaría a ser 1(2), por ejemplo.
  - Mostrar un mensaje de error.
- Cuando acaba una sesión los ejercicios activos pasan al estado preparados, quedando guardados todos los avances realizados durante la sesión.
- Se ha pensado en dejar la parte de eliminar completamente ejercicios a Presence-Click, entrando como administrador.

**Acordado para la siguiente reunión:**

- Desarrollar unos prototipos en papel con lo acordado durante la reunión, a limpio.

---

## Reunión de Trabajo 4

**Fecha:** 11 de noviembre de 2014

**Hora de inicio:** 11:00

**Hora de finalización:** 11:40

**Presentes:** Maite Urretavizcaya, Samara Ruiz, Adrián Núñez

### **Temas tratados durante la reunión:**

- Planteamiento de la funcionalidad de los UO-1, UO-2, UO-3 y UO-4 mediante los prototipos en papel y los autómatas de estados (fin de M-1).
- Inicio de M-2 (implementación de la interfaz).

### **Resumen de la reunión:**

- Se muestran los prototipos en papel junto a sus correspondientes autómatas de estados (dos: uno del profesor y otro del alumno).
- Se acuerda que cada UO tendrá el siguiente contenido:
  - Nombre del UO.
  - Descripción breve.
  - Autómata de estados individual del UO.
  - Pantallas necesarias para el UO.
- Se visualiza, por cada UO (del 1 al 4), qué pantallas debería usar y cómo debería ser la interacción con el usuario para cumplir el UO.
- Se plantean las siguientes ideas/propuestas:
  - Vista alumno:
    - Cuando el alumno responde a un ejercicio, el botón que ha marcado (ya sea que ha acabado o que ha tenido una duda) se resaltará. También es posible que se le cambien los bordes al contenedor en el que está el ejercicio.
    - Siguiendo con los mismos botones: añadir un lapso de tiempo antes de que la respuesta del alumno cuente dentro de las estadísticas por si se hubiera equivocado pulsando el botón.

- Añadir dos barras de progreso de la sesión: una del alumno y otra de la clase (para poder ver su progreso en comparación con el resto de compañeros).
  - Vista profesor:
    - Los iconos para ejercicio acabado o alumno con dudas deben ser iguales para ambas vista: profesor y alumno.
    - En vez de usar estadísticas del tipo “20/100 han acabado este ejercicio” interesa usar porcentajes (20 %), ya que para otros valores puede que no quede tan claro el porcentaje real.
    - En la parte de dudas, en estadísticas, se mostrarán sólo los alumnos que tienen en el momento dudas, no los que tuvieron anteriormente.
    - Cuando el ejercicio está finalizado se propone añadir diferentes estadísticas (nuevo UO que queda como pendiente): gente que ha tenido duda en algún momento, dudas resueltas, gente que se quedó con dudas, gente que terminó el ejercicio y tiempo de realización del mismo.
  - Para esta iteración se acuerda que, para simplificar las funcionalidades que vamos a añadir (no queremos añadir todas las funcionalidades de golpe), omitiremos detalles del diseño original que más tarde serán añadidos:
    - Pestaña de ejercicio preparados.
    - Opciones avanzadas de creación de un ejercicio.
    - Guardar un ejercicio (para que se envíe a preparados). Sólo se podrá lanzar.
- Si bien es posible que existan más elementos a omitir, estos no fueron identificados en el momento de la reunión.
- Se acuerda finalmente iniciar con la implementación de la interfaz mediante las tecnologías HTML y CSS. Además, se usará PhoneGap para portar estas tecnologías a las plataformas móviles (en principio Android, iOS y Windows Phone).

**Acordado para la siguiente reunión:**

- Implementación de la interfaz en HTML y CSS y adaptación a móviles de esta mediante PhoneGap.

---

## Reunión de Trabajo 5

**Fecha:** 13 de enero de 2015

**Hora de inicio:** 11:50

**Hora de finalización:** 12:30

**Presentes:** Maite Urretavizcaya, Samara Ruíz, Juan Miguel López, Adrián Núñez

### **Temas tratados durante la reunión:**

- Demostración y discusión del M-2 del UO1-T.

### **Resumen de la reunión:**

- Se empieza mostrando el M-2 del UO1-T desarrollado en un Samsung Galaxy S5 al grupo. La aplicación ha sido añadida al dispositivo mediante USB, no se ha conseguido una apk funcional para instalar la aplicación en cualquier dispositivo.
- De la próxima reunión en adelante se enviarán capturas de pantalla del modelo para agilizar la reunión y que todos los participantes tengan claro el escenario.
- Aspectos de diseño a mejorar:
  - Los botones rápidos que aparecen en las cajas de los ejercicios deben ir siempre a la derecha, alineados con el identificador del ejercicio (aunque haya que acortarlo mucho, se supone que debe ser una clave corta). De esta forma la caja ocupa menos. Además, al pulsar el ejercicio para verlo completo (más adelante) se podrá ver el identificador entero.
  - Alguna forma más visual de identificar un ejercicio en el que haya especialmente un número significativo de dudas (color de fondo, algo llamativo, texto de otro color, etc.).
- Se acuerda realizar finalmente el M-3 del UO1-T. Además, se acuerda seguir con el UO1-S (los 3 modelos) para poder realizar así una evaluación más funcional (con el profesor pudiendo lanzar un ejercicio y el alumno respondiendo a este). También se plantea realizar el UO4-T (ver estadísticas de un ejercicio) para poder tener una aplicación verdaderamente funcional.

### **Acordado para la siguiente reunión:**

- M-3 del UO1-T.



- M-1, M-2 y M-3 del UO1-S y UO4-T.
- Obtener una apk funcional de la aplicación.

---

## Reunión de Trabajo 6

**Fecha:** 27 de enero de 2015

**Hora de inicio:** 9:30

**Hora de finalización:** 10:10

**Presentes:** Maite Urretavizcaya, Adrián Núñez

### **Temas tratados durante la reunión:**

- Análisis de la validación de los M-2 de UO1-T, UO4-T y UO1-S.

### **Resumen de la reunión:**

- Intercambiar los iconos de ejercicios activos (señal de alerta) y de ejercicios preparados (avión de papel).
- Modificar los colores de los botones: rojo siempre activo, azul finalizado y amarillo preparado. Por ejemplo, el de lanzar un nuevo ejercicio (que salía con el avión de papel en amarillo) debería cambiar a rojo.
- Colocar 3 botones por cada ejercicio para poder mover los ejercicios entre los estados activo, finalizado y preparado cuando se desee.
- Hacer la barra superior más grande y la parte del nombre de la asignatura y el fondo tipo pizarra más pequeño.
- Usar las fuentes EHU Sans para el idioma castellano y EHU Serif para euskera.
- A la vista de alumno añadirle la barra extra de progreso de la clase. También añadirle a los ejercicios colores de fondo dependiendo de sus estados.

### **Acordado para la siguiente reunión:**

- Mejorar lo comentado y continuar con el proyecto.
- Actualizar la lista de UOs con información actualizada.

## Reunión de Trabajo 7

**Fecha:** 6 de febrero de 2015

**Hora de inicio:** 11:30

**Hora de finalización:** 12:20

**Presentes:** Maite Urretavizcaya, Samara Ruíz, Adrián Núñez

### **Temas tratados durante la reunión:**

- Demostración de los M-3 de los UO1-T, UO3-T, UO4-T y UO1-S.

### **Resumen de la reunión:**

- Se les enseña a los asistentes los M-3 de los UO1-T, UO3-T, UO4-T y UO1-S. La funcionalidad básica (o ciclo principal de la aplicación) de enviar ejercicios y que los alumnos respondan y se refleje esta respuesta en la vista de profesor no está completa.
- Se deciden realizar las siguientes modificaciones:
  - Modificar las barras de progreso: colocar un texto más corto, luego la barra y finalmente el porcentaje, todo en una misma línea para que no ocupen tanta altura (esto se aplica para las barras de progreso de la vista de profesor y para las de la vista de estudiante).
  - El tamaño de la barra superior y de la zona de título de la asignatura hacerlos más grandes para que se vean mejor.
- La internacionalización de la aplicación (tenerla disponible en 4 idiomas) queda fijada como objetivo a corto plazo para ir pensando y planificando para añadir. Se dan algunas ideas basadas en qClick y PresenceClick del modo de traducir el texto.
- Se comentan algunas modificaciones para hacer a la memoria (se dejan por escritas en un documento previamente preparado por la directora del proyecto que le es entregado al alumno).

### **Acordado para la siguiente reunión:**

- Seguir intentando generar el archivo .apk.
- Buscar la internacionalización de la aplicación.

- 
- Conseguir que el ciclo principal de la aplicación funcione.
  - Corregir la memoria y el diseño de la aplicación con lo comentado durante la reunión.