
This is the **published version** of the bachelor thesis:

Aranda Fernández, Luis; Pons, Jordi. Aplicación web para el seguimiento y tutorización de las prácticas externas SPE 3.0. 2021. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/257817>

under the terms of the  license

Aplicación web para el seguimiento y tutorización de las prácticas externas

SPE 3.0

Luis Aranda Fernández

Resumen– Las prácticas externas son una asignatura presente en la mayoría de estudios de grado. En el grado en Ingeniería Informática el profesorado realiza el seguimiento de las prácticas a través de una aplicación web desarrollada en cursos anteriores llamada Seguimiento de Prácticas Externas (SPE). La herramienta se basa en la gestión de las distintas fases del seguimiento de la estancia de prácticas mediante el intercambio de comentarios. En este trabajo se muestran mejoras en la aplicación ya existente en el ámbito de la seguridad, la usabilidad y la incorporación de nuevas funcionalidades que permiten una mejor comunicación y la flexibilización del sistema de seguimiento para nuevos cursos y otros grados.

Palabras clave– Prácticas Externas, Desarrollo web, PHP, Javascript, VueJS

Abstract– External Internships are a subject present in most undergraduate programs. In the Computer Engineering undergraduate program, teachers monitor this internship through a web application developed in previous courses called "Seguimiento de Prácticas Externas (SPE)". This application is based on the management of internship's different phases through the exchange of comments. In this project different improvements are shown over the existent application on the scopes of security, usability and the inclusion of new functionalities which allow a better communication and flexibility on the monitoring system for new courses and study programmes.

Keywords– External Internships, Web development, PHP, Javascript, VueJS



Seguimiento de Prácticas Externas (SPE) donde se registra el seguimiento del alumnado para un buen aprovechamiento de la estancia de prácticas.

1 INTRODUCCIÓN

Durante sus estudios de grado parte del alumnado realiza prácticas externas curriculares para tener un primer contacto con el mundo laboral y aplicar los conocimientos obtenidos.

En la asignatura Prácticas Externas se realiza un seguimiento para que el alumnado aproveche al máximo su estancia de prácticas. En el seguimiento participan tanto el alumno como la persona tutora académica y la persona tutora de la empresa.

Este tutor académico designado por la universidad se encarga del seguimiento de la estancia. Para facilitar este seguimiento a las personas coordinadoras y tutoras se ha desarrollado en cursos anteriores una aplicación web llamada

Para el seguimiento se definen diferentes fases: inicial, seguimiento y final. Dentro de cada fase hay un intercambio de correos electrónicos y reuniones de seguimiento entre la persona tutora académica y el estudiante o la persona tutora de la empresa. Cada una de estas actividades es una tarea que se puede dividir en distintas acciones que se realizan a lo largo de la estancia. [1]

La aplicación existente, pese a ser totalmente funcional tiene algunos problemas de usabilidad y seguridad que se tratarán en este trabajo. A su vez también se pretende ampliar la funcionalidad y hacer la aplicación más flexible para que se pueda modificar el sistema de seguimiento utilizado, para fomentar el uso de la aplicación en otros grados y mejorar el mantenimiento entre cursos.

En este documento veremos el desarrollo del proyecto desde la toma de requisitos y planificación hasta su desarrollo y posterior análisis de los resultados.

- E-mail de contacto: luis.aranda@autonoma.cat
- Mención realizada: Tecnologías de la Información
- Trabajo tutorizado por: Jordi Pons Aróztegui (DEIC)
- Curso 2021/22

2 ESTADO DEL ARTE

2.1. Tecnologías utilizadas

Como se ha dicho anteriormente, ya existe una aplicación web para gestionar las prácticas externas que se pretende mejorar en este trabajo. Ahora veremos las tecnologías utilizadas en la versión anterior de la aplicación.

La aplicación existente fue desarrollada el curso pasado por un alumno como trabajo de fin de grado. Para el front-end de la aplicación se utiliza HTML, CSS y Javascript. Utiliza la librería Bootstrap para la creación de interfaces de usuario. Para el back-end utiliza PHP y SQL. Se apoya en distintas librerías en el front-end: Bootstrap [2] para el estilo de la página, JQuery para facilitar las peticiones asíncronas al servidor, DataTables para la creación dinámica de tablas y Summernote para proporcionar un editor de texto. [3]

Aunque *frameworks* como JQuery siguen formando una parte importante del mercado de aplicaciones web, ahora las aplicaciones se orientan a una *Single Page Application (SPA)* [4] con nuevos framework como React, AngularJS o VueJS. Durante la fase de desarrollo de la aplicación se valorará la incorporación de alguno de estos frameworks para el desarrollo de las nuevas funcionalidades.

En el backend no se utiliza ningún framework específico, utilizando únicamente PHP con su librería estándar. Aunque PHP es uno de los lenguajes más utilizados para aplicaciones web, hoy en día muchos desarrollos se realizan con un modelo de API REST con frameworks como NodeJS, que permiten la utilización de Javascript tanto en el front-end como en el back-end de la aplicación. La aplicación existente utiliza el clásico patrón de diseño Modelo-Vista-Controlador para el acceso a la base de datos y la representación de los datos. [5]

Cambiar completamente el framework utilizado manteniendo la compatibilidad con las versiones anteriores de la aplicación y añadir nuevas funcionalidades resulta inviable con el número de horas de un TFG. Por lo tanto no se va a sugerir cambiar ningún framework, únicamente incorporar algún framework o librería en las nuevas funcionalidades si se considera conveniente.

2.2. Funcionamiento de la aplicación

Ahora veremos el funcionamiento de la aplicación existente para entender mejor la necesidad de este nuevo proyecto y las mejoras descritas posteriormente. Al entrar en la aplicación los usuarios son redirigidos al *Servei d'Autenticació Centralitzat de la Universitat Autònoma de Barcelona*, el sistema de control de acceso que regula el inicio de sesión en la aplicación como podemos ver en la figura 1.

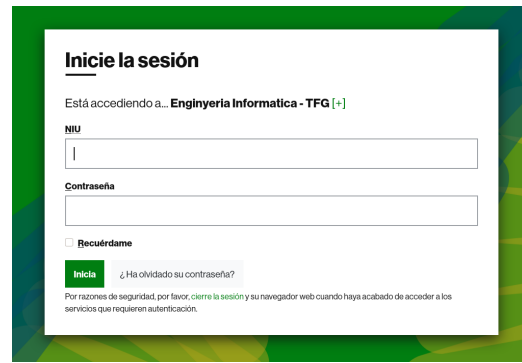


Fig. 1: *Servei d'Autenticació Centralitzat de la Universitat Autònoma de Barcelona*

Una vez los usuarios han iniciado sesión, verán una vista distinta dependiendo de su rol. Los usuarios pueden ser alumnos, profesores y coordinadores. El alumnado al entrar en la aplicación accede a un *dashboard* donde se resume el estado de su estancia de prácticas y se muestra información de utilidad como los datos de contacto de su tutor académico y la persona tutora en la empresa, además de las fechas de inicio y finalización de la estancia. Más allá de visualizar esta información, los usuarios con el rol alumno no pueden interactuar más con la aplicación.

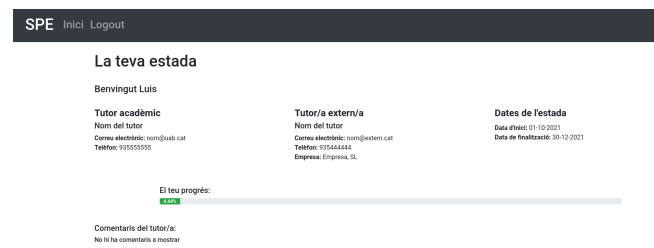


Fig. 2: Resumen de la estancia que ve el alumno

Los usuarios con el rol profesor al entrar en la aplicación, acceden a una vista con una serie de tarjetas que les recuerda las estancias con tareas pendientes de revisar.



Fig. 3: Vista de las estancias con tareas pendientes

Al revisar una estancia, el profesorado ve una lista con las distintas fases. Como cada fase tiene distintas tareas, estas aparecen con sus correspondientes fechas en una tabla. El color de fondo de las fechas determina si se han realizado (verde) o si ya ha pasado su fecha sin realizarse (rojo). Si aun no se ha llegado a la fecha, la tarea no tiene color de fondo.

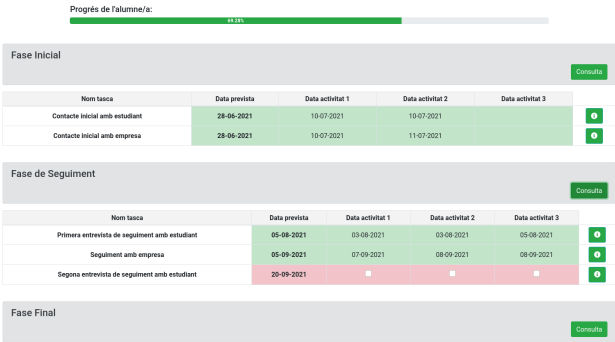


Fig. 4: Lista fases y tareas

Desde esta vista, el profesorado también puede modificar datos sobre el alumno y añadir comentarios acerca de la estancia, que se añaden a una tabla. También se pueden ver y editar las acciones asociadas a cada tarea, así como el correo que se propone enviar.

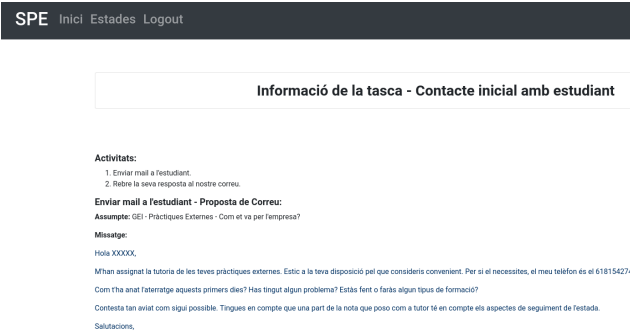


Fig. 5: Edición de la información propuesta sobre una tarea

El profesorado, al ser tutor académico de distintos alumnos tienen acceso a una tabla que detalla el estado de cada alumno tutorizado, donde aparecen las fechas de las distintas actividades. Si la tarea se ha realizado aparecerá con el fondo verde, si se acerca a la fecha límite o se ha sobrepasado aparecerá en naranja, y si han pasado más de 5 días a partir de la fecha límite, aparecerá en rojo.



Fig. 6: Tabla resumen con todas las estancias y su progreso

Los usuarios coordinadores al entrar en la aplicación eligen si quieren entrar con el rol de profesor o el rol de coordinador. Los coordinadores, respecto al seguimiento de las estancias, pueden hacer lo mismo que los profesores excepto publicar comentarios. También pueden eliminar y añadir estancias nuevas. La principal diferencia del rol de coordinador es el acceso a un panel de administración que permite

modificar el profesorado, los departamentos y coordinadores, además de añadir cursos y grados nuevos.

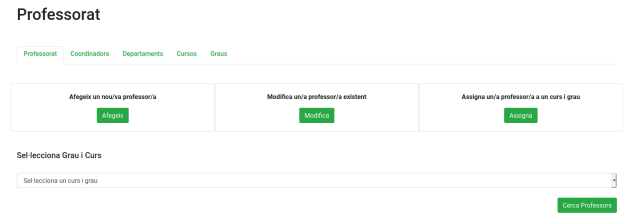


Fig. 7: Panel de administración

3 OBJETIVOS

Los objetivos se han dividido en distintas categorías dependiendo del ámbito de la aplicación que pretende mejorar. Las categorías elegidas son usabilidad, seguridad y funcionalidad. Dentro de cada categoría los objetivos se han ordenado por prioridad para facilitar la planificación del trabajo de cara a terminar el proyecto con la mayor cantidad de objetivos cumplidos. En el anexo A de este documento se encuentra una tabla con todos los objetivos clasificados por su categoría y ordenados por su prioridad.

3.1. Seguridad

Mejorar la seguridad de la aplicación es uno de los objetivos de este proyecto. Primero sería necesario analizar la seguridad actual de la aplicación web y subsanar posibles fallos de seguridad que se detectasen. La utilización de herramientas automáticas de detección de vulnerabilidades web sería interesante para facilitar la tarea de encontrar vulnerabilidades. Actualmente la aplicación permite realizar algunas consultas enviando peticiones al servidor aunque se carezcan de los permisos de usuario necesarios sin respetar el control de acceso basado en roles que posee la aplicación. La seguridad de la aplicación es un objetivo que no debería de olvidarse en ningún momento en el desarrollo del proyecto, sobre todo de cara a desarrollar nuevas funcionalidades que introduzcan código nuevo en la aplicación, que puedan generar nuevos fallos de seguridad.

Por lo tanto, subsanar vulnerabilidades web y controlar la seguridad de las peticiones son dos de los objetivos que entran dentro del ámbito de la seguridad y por lo tanto se les dará una prioridad alta dentro del trabajo.

Objetivo [Seguridad]	Prioridad
Revisar y subsanar vulnerabilidades web	Alta
Corregir seguridad de las peticiones	Alta

3.2. Usabilidad

Otro de los objetivos es mejorar la usabilidad de cara a los usuarios que utilizan la aplicación. Existen distintos tipos de usuarios a los que está dirigida la aplicación (alumnos, profesores y coordinadores) y cada grupo tiene unas necesidades específicas respecto a la funcionalidad de la aplicación.

El alumnado solo puede ver un panel con información sobre el estado actual de su estancia de prácticas. Profesores y coordinadores tienen acceso a una tabla que resume

el estado de las distintas estancias de prácticas que tutorizan. Reducir la cantidad de clicks necesarios y orientar la web hacia una Single Page Application (SPA) haciendo que esta no se refresque mejoraría su usabilidad y futura escalabilidad al liberar al servidor de parte del renderizado de la aplicación y hacer que el cliente se encargase de ello.

Mejorar la rapidez al utilizar la aplicación es uno de los objetivos que entran dentro del ámbito de la usabilidad. Armonizar el estilo de la aplicación con el del resto de páginas de la UAB y añadir soporte multilinguaje mejoraría también la usabilidad de la página, pero su prioridad es más baja porque su falta no ha causado grandes problemas en el tiempo que se ha utilizado la versión actual de la aplicación.

Objetivo [Usabilidad]	Prioridad
Reducir el tiempo de interacción con la herramienta	Alta
Facilitar la ordenación de tablas por fechas	Media
Dar soporte multilinguaje	Baja
Armonizar estilo con el resto de aplicaciones de la UAB	Baja

3.3. Nuevas funcionalidades

Ampliar la funcionalidad de la aplicación es otro objetivo de este proyecto. Actualmente el sistema de seguimiento es poco flexible y no se pueden definir fácilmente nuevos tipos de actividades con distintas tareas. Añadiendo la capacidad de crear nuevos sistemas de seguimiento que tengan tareas y actividades distintas la aplicación sería mucho más accesible a otros grados. Para ello, se incluiría una herramienta donde los coordinadores pudiesen crear plantillas que luego serían asignadas a los distintos cursos de cada grado. Hasta ahora para modificar las distintas tareas de las fases de cada curso de prácticas es necesario editar la base de datos, lo que haría que esta nueva funcionalidad también fuese útil al cambiar de curso académico.

Respecto al mantenimiento de la aplicación, añadir la funcionalidad de poder ver y modificar todas las empresas y tutores externos haría que existiesen menos duplicados. Actualmente esto solo se puede realizar editando la base de datos.

La comunicación entre el alumnado y el profesorado se podría mejorar si se añadiese la posibilidad de subir archivos al servidor. De este modo, los alumnos podrían subir una foto de perfil por ejemplo para mejorar su identificación y los tutores podrían adjuntar documentos relacionados con la estancia.

También, si se modifica la fecha de fin de una estancia, las fechas intermedias no se recalculan. Con el fin de mejorar la funcionalidad de la aplicación, habría que hacer que este proceso fuese automático.

Objetivo [Funcionalidad]	Prioridad
Flexibilizar el sistema de seguimiento	Alta
Permitir que el cambio de fechas de inicio y finalización de las estancias se refleje en las fechas de seguimiento	Alta
Facilitar la comunicación entre alumno y tutor	Media
Mejorar la información disponible de empresas y tutores externos	Baja

En el anexo A de este documento se encuentra una tabla a modo de resumen de todos los objetivos ordenados según su ámbito y prioridad con su descripción.

4 METODOLOGÍA Y PLANIFICACIÓN

4.1. Metodología

El uso de metodologías Agile es básico en el desarrollo de software moderno para garantizar un desarrollo ordenado y lo más óptimo posible.

Las metodologías Agile se basan en un desarrollo de software incremental e iterativo, en contraposición al desarrollo de software en cascada. El desarrollo incremental nos permite tener una mayor flexibilidad durante el ciclo de desarrollo si se detectan problemas o surgen nuevas funcionalidades a implementar. [6]

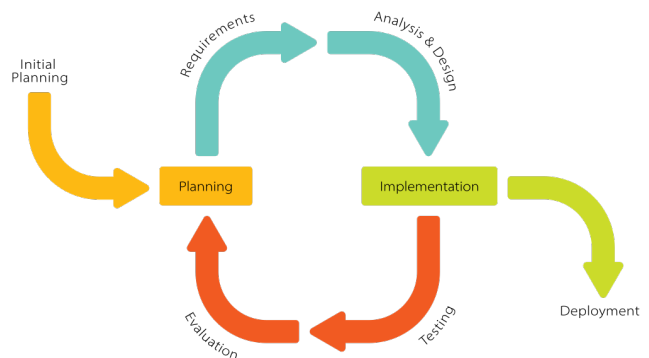


Fig. 8: Esquema del desarrollo de software iterativo [7]

La metodología de desarrollo Agile más extendida es Scrum, pese a ello no ha sido utilizada en este proyecto ya que al constar de un único desarrollador, no tiene mucho sentido utilizar una metodología con múltiples roles. Por esto mismo, la metodología Kanban ha sido la empleada en el proyecto. [8]

Esta metodología se basa en el uso de un tablero donde las tareas van avanzando de columna hasta su finalización. Con esta metodología las entregas y el contacto con el cliente son algo continuo que admiten un cambio en los objetivos y tareas si las circunstancias cambian. [9]

Se ha utilizado el tablero de Trello para colocar las distintas tareas y realizar así la metodología Kanban. Se ha utilizado una plantilla proporcionada por el propio equipo de Trello para desarrollos Kanban [10].

Tenemos un *Backlog* donde se colocan las tareas que se han planeado realizar, que pasan a la columna *Design* cuando se encuentran en una fase de diseño o investigación para encontrar la manera de cumplir la tarea.

Tras ello pasan a la columna *To-Do* en orden de prioridad donde se van escogiendo para ser realizadas y pasando a la columna *Doing*. Una vez ya se ha realizado la tarea pasa a *Code Review* donde se comprueba la funcionalidad deseada y se revisa la calidad del código en caso de tratarse de una tarea relacionada directamente con el desarrollo.

Finalmente cuando la tarea ha terminado la fase de comprobación del código y se ha corregido su calidad pasa a *Testing* donde se somete a pruebas para asegurar el cumplimiento de la funcionalidad deseada. Cuando estas pruebas se pasan satisfactoriamente la tarea pasa a la columna *Done* y en caso contrario vuelve a una de las columnas anteriores para subsanar el error.

También se han añadido etiquetas para saber a qué parte del proyecto corresponde cada tarea (seguridad, usabilidad, backend, frontend...).

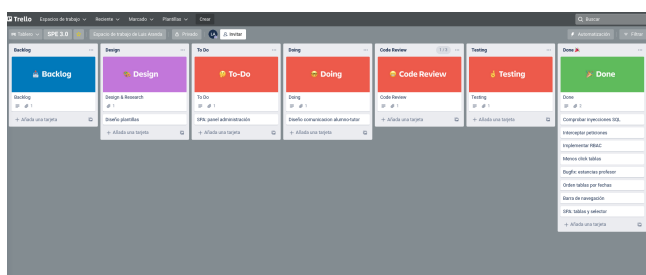


Fig. 9: Tablero Kanban en Trello

La metodología Kanban ha sido muy útil durante el proceso de desarrollo para encontrar errores en la implementación y llevar un desarrollo ordenado, con una correcta fase de diseño anterior a la implementación y una fase de pruebas posterior a la misma.

El desarrollo se ha realizado en un contenedor Docker en local con Apache y MariaDB en un repositorio git local. En un principio se planeó subir los cambios cada poco tiempo a un servidor web proporcionado por el DEIC después de realizar el desarrollo en local. Esto no fue posible durante todo el desarrollo del proyecto debido al ciberataque sufrido en la UAB a principios de octubre. No fue posible probar la aplicación con el servidor de autenticación de la universidad, así que se implementó un pequeño sistema de inicio de sesión para el desarrollo de la aplicación que no requiere credenciales.

Finalmente los servicios de autenticación y el servidor del DEIC se reestablecieron y se pudo probar la aplicación con el servidor de autenticación, funcionando correctamente tras unos ajustes. El cliente pudo entonces probar la aplicación para hacer unas últimas correcciones.

Para integrar las nuevas funcionalidades y mejorar la usabilidad de la página, se ha añadido un framework web más actual basado en los datos reactivos. Finalmente se ha decidido optar por VueJS debido a que ya estoy familiarizado con él y su funcionalidad basada en componentes le proporciona modularidad. VueJS permite que pequeñas partes de la aplicación como la herramienta de creación de plantillas estén escritas utilizando VueJS sin necesidad de reescribir la aplicación por completo [11].

4.2. Planificación

Al comienzo del proyecto se comenzó con la toma inicial de requisitos del cliente en la primera semana. En la segunda semana se empezaron a determinar los objetivos del proyecto y se montó el entorno de desarrollo local. Durante la tercera semana se redactó el informe inicial de seguimiento y se adaptó el código de la aplicación al desarrollo local.

En la cuarta semana tuvimos el primer milestone del proyecto, la entrega del informe inicial. Tras completar este primer milestone, se dedicaron las siguientes dos semanas a los objetivos de mejora de seguridad de la aplicación, debido a su carácter crítico.

A la vez que se terminaron de corregir las vulnerabilidades de la aplicación, se comenzará a trabajar en la usabilidad mejorando el cuadro de fechas. La próxima semana (semana 7) se integró el código y las librerías necesarias como VueJS para hacer que las nuevas funcionalidades de la aplicación se orienten hacia una *Single Page Application*.

Tras esto, se comenzó a trabajar en la ampliación de funcionalidades con el diseño de las mismas a nivel de desarrollo de software. La semana 9 se llegó al segundo milestone del proyecto, la entrega del primer informe de progreso.

Al entregar este milestone se continuó trabajando en la ampliación de funcionalidades, comenzando la implementación de las nuevas plantillas durante 2 semanas y implementando las funcionalidades relacionadas con la comunicación entre alumno y profesor otras 2 semanas.

Llegados a este punto se hicieron cambios en la planificación inicial al entender que había que dividir el desarrollo de las nuevas funcionalidades entre *backend* y *frontend*.

Llegados a la semana 14 se retomó el trabajo respecto a la usabilidad del proyecto, añadiendo posibles mejoras o solucionando fallos que aparecieron aparecer en etapas anteriores. Después se redactó el segundo informe de progreso y se entregará la cuarta milestone, el segundo informe (semana 15). Las siguientes dos semanas se dedicaron a posibles contingencias como añadir o arreglar funcionalidades o aspectos de usabilidad que se han quedado atrás.

Finalmente, podemos decir que se ha seguido un desarrollo cercano al método evolutivo, donde a partir de un prototipo inicial se ha ido refinando con los requisitos del cliente.

Durante la semana 19 se redactó la propuesta de informe final y presentación, para entregar este informe como milestone la semana 19. La semana 21 sería el último milestone, la defensa del TFG.

En el anexo B de este documento se encuentra un diagrama de Gantt con la planificación del proyecto semana a semana.

4.3. Entorno de desarrollo

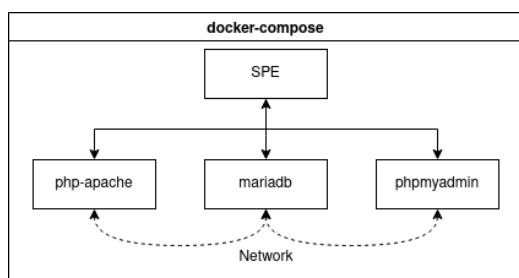


Fig. 10: Contenedores docker creados

Para comenzar el desarrollo se montó un contenedor docker que provee 3 servicios para simular el servidor donde se alojará la aplicación. Tenemos un contenedor docker con el servidor web Apache y PHP, otro con el sistema gestor de base de datos MariaDb y otro con la herramienta phpmyadmin para gestionar la base de datos. El contenedor donde está el servidor web accede a través de un volumen virtual montado al directorio donde está el código de la aplicación.

Como herramienta de control de versiones se ha utilizado git tanto para el desarrollo de la aplicación como para las distintas versiones de los informes.

5 RESULTADOS

En este apartado revisaremos los objetivos marcados a principios del proyecto y describiremos qué se ha hecho para su cumplimiento.

■ Revisar y subsanar vulnerabilidades web [Seguridad]

Se han utilizado herramientas automáticas de detección de vulnerabilidades como sql-map para detectar posibles inyecciones SQL [12].

No se han detectado vulnerabilidades de tipo inyección SQL, comunes en aplicaciones web. Tampoco se detectaron posibles ataques XSS (Cross Site Scripting).

■ Corregir seguridad de las peticiones [Seguridad]

El control de acceso no era el adecuado en la mayoría de funcionalidades de la web. Se ha implementado un correcto control de acceso por rol centralizando todo en un único archivo php para facilitar su mantenimiento.

Un aspecto de la seguridad que se ha corregido ha sido una correcta implementación del RBAC (Role Based Access Control), que limita a cada usuario a realizar únicamente las operaciones permitidas en su rol. Se han aprovechado los roles de usuarios ya existentes en la base de datos (alumno, profesor y coordinador) para establecer los niveles de permisos necesarios.

En el anexo C se puede observar el diagrama entidad-relación de la parte de la base de datos destinada a determinar los roles de usuario (parte resaltada en rojo). En la tabla *tipos_usuario* se guardan los distintos tipos de usuarios posibles, y en dos claves foráneas de cada entrada en la tabla *Usuarios* se guardan los roles del

usuario. Como hay usuarios que pueden acceder con dos roles es necesario tener dos campos para identificar los roles.

Estos campos de las bases de datos son los que se han utilizado al comprobar si el usuario tiene los roles necesarios en la base de datos al realizar las distintas peticiones.

Una adecuada implementación del RBAC se ha llevado a cabo siguiendo las directrices de OWASP (*Open Web Application Security Project*) [13], un referente a nivel de seguridad web. Se han interceptado las peticiones que realizaba la aplicación web mediante la herramienta Burpsuite y se han intentado replicar sin los permisos necesarios, corrigiendo el control de permisos donde era necesario [14].

■ Flexibilizar el sistema de seguimiento [Funcionalidad]

Para permitir flexibilizar el sistema de seguimiento se ha creado una herramienta que permite la creación de plantillas. Estas plantillas se componen de distintas tareas a las que se puede asignar una fase de seguimiento específica y acciones a realizar. Cuando se crea un nuevo curso/grado se selecciona una plantilla (se proporciona una por defecto) para copiar todas las tareas de esta al crear cada nueva estancia de prácticas.

Las tablas de la base de datos involucradas en la creación de un nuevo curso/grado se pueden ver resaltadas en azul en el diagrama del anexo C.

La herramienta de plantillas ha sido desarrollada como una SPA utilizando la librería VueJS. Esto nos proporciona la capacidad de tener datos reactivos en la herramienta, que cambian automáticamente al editarse. Desde la herramienta se pueden crear nuevas plantillas con el número de tareas que se necesite y especificar los detalles de cada tarea. La plantilla se visualiza como una tabla donde cada tarea es una línea de la tabla como se puede ver en la figura 11.





Se ha desarrollado un componente de VueJS para mostrar el visor/editor de plantillas desde el componente principal donde se listan todas las plantillas. Todo esto se encuentra en una nueva vista dentro del panel de administración. También se ha añadido una vista donde se pueden añadir y modificar acciones para luego poder escogerlas al crear las tareas de una plantilla.

Ha sido necesario integrar el editor de texto *Summernote* dentro de un componente de Vue para que fuese compatible con el framework. [15] De esta manera, el editor de texto simula un *input text* de HTML para el framework VueJS y funciona sin problemas, a pesar de que *Summernote* utilice JQuery.

Afegir una plantilla x

Plantilla de prova

Tasques

	Nom	Missatge	Fase	Acció 1	Acció 2	Acció 3	Percentatge
	Tasca test 1	Mostra	Inicial	Enviar mail a l'estudiant.			10 %
	Tasca test 2	Mostra	Seguiment	Entrevista de seguiment	Enviar mail a l'estudiant.		30 %
	Tasca final	Mostra	Final	Confirmació de recepció.	Enviar mail o trucar l'estudiant per quedar.	Entrevista de seguiment	100 %
	<input type="text"/>	Editar	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Tanca Afegir

Fig. 11: Herramienta de creación de plantillas

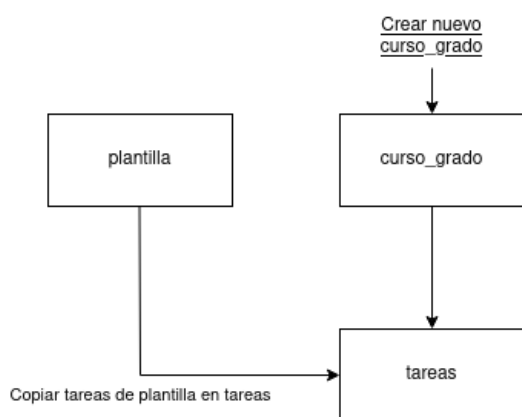


Fig. 12: Procesos al crear un nuevo curso_grado

En la figura 12 podemos ver los nuevos procesos que se realizan al añadir un nuevo curso o una estancia a un alumno. Al crear un nuevo curso_grado, se selecciona una plantilla para determinar las tareas de seguimiento de las estancias. Como se puede ver en el diagrama de la figura 12, las tareas de la plantilla determinada se copian en la tabla de tarea, quedando relacionadas así con el curso_grado.

Cuando un estudiante se asigna a una estancia, se deben de copiar de nuevo estas tareas para asignarlas al estudiante y poder seguir su progreso como se observa en el diagrama de la figura 13.

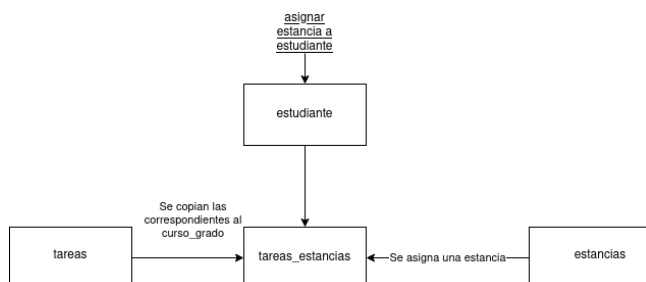


Fig. 13: Procesos al asignar una estancia a un estudiante

Se crea una nueva entrada en la tabla *estancias* que queda relacionada con el alumno y las futuras tareas. Después se copian las tareas de la tabla *tareas* en la

tabla *tarea_estancias* quedando relacionadas las tareas con la estancia.

- **Permitir que el cambio de fechas de inicio y finalización de las estancias se refleje en las fechas de seguimiento [Funcionalidad]**

Ahora al modificar la fecha de finalización de una estancia se da la opción de recalcular las fechas de seguimiento mediante una checkbox.

Al recalcular las fechas de seguimiento se incrementa el número de días que se haya añadido a la finalización de la estancia si el checkbox se ha marcado.

- **Facilitar la comunicación entre alumno y tutor [Funcionalidad]**

Se ha implementado la posibilidad de que los alumnos tengan una foto de perfil para ser a su vez mejor identificados. Esta foto de perfil se puede ver al observar la vista del alumno.

Cuando el alumno inicia sesión se le recuerda subir una foto de perfil si no tiene ninguna.

Configurar compte d'usuari x

La teva imatge de perfil



Escolliu una imatge Navega

Tanca Guarda

Fig. 14: Modal para cambiar la imagen de perfil

Los profesores y coordinadores pueden subir documentos adjuntos a los comentarios. Al enviar un comentario se puede subir un documento que queda guardado junto al comentario. Luego al visualizarse el comentario se muestra un botón para descargar el archivo si existe.

Tipus de comentari:

Privat

Categoria del comentari:

Estudiant

Comentari:

B *I* U ~~ABC~~ **S** **X'** **X** 16 **A** **-** **≡** **≡** **≡** **T1**

Document Escolliu un arxiu .pdf

Afegeix

Fig. 15: Campo para añadir un documento al comentario

Al listar los comentarios, se muestra una nueva columna con un botón para visualizar el documento en caso de que exista. Al dar click a este botón se abre un modal con la previsualización del archivo.

10-07-2021	Empresa	Privat	Cap document
11-07-2021	Empresa	Privat	Cap document
30-01-2022	Estudiant	Privat	Document
30-01-2022	Estudiant	Privat	Document

Fig. 16: Botón para ver el documento

Ha sido importante investigar como hacer que la subida de ficheros sea segura y no se puedan descargar posteriormente por usuarios no autorizados. El archivo se guarda quedando registrado el hash en la base de datos, para poder luego recuperarse del disco de forma segura cuando un usuario lo solicite. De esta forma evitamos que un usuario pueda intentar descargar por fuerza bruta todos los documentos subidos al servidor que no le corresponden.

También destacar la importancia de asignar correctamente los permisos de los ficheros en el servidor, para que únicamente el administrador y el usuario del servidor web puedan acceder a los ficheros y no permitir la ejecución de ficheros subidos por los usuarios.

■ Mejorar la información disponible de empresas y tutores externos [Funcionalidad]

Como es fácil que existan tutores externos y empresas duplicadas debido al funcionamiento de la herramienta, se ha implementando un panel de control en el área de administración para poder modificar estos datos sin acceder directamente a la base de datos. Este panel es similar al ya existente para la modificación de profesores de los distintos departamentos.

■ Reducir el tiempo de interacción con la herramienta [Usabilidad]

El número de clicks necesarios para realizar determinadas tareas se ha reducido y también ha aumentado el aprovechamiento del espacio disponible en pantalla, sobre todo en vistas donde se muestran muchos datos en tablas como la vista de estancias (como se puede ver en la figura 17). Algunas peticiones han reducido su tamaño por lo que la web se carga de forma más ágil y se ahorran recursos.

En un principio se iba a transformar el panel de administración en una SPA pero la tarea se retrasó y debido a la falta de tiempo finalmente no se pudo realizar.

■ Facilitar la ordenación de tablas por fechas [Usabilidad]

En las tablas no era posible ordenar los campos de la tabla por fecha ya que o bien trataba la fecha como un número o utilizaba el formato americano de fechas. Utilizando un nuevo elemento de HTML5 y Javascript se ha podido ampliar la librería utilizada para mostrar las tablas (Datatables) para poder así ordenar las fechas y que estas se representen en el formato correcto.

■ Dar soporte multilinguaje [Usabilidad]

Este objetivo tenía una prioridad baja y finalmente ha quedado fuera del proyecto. Quedará disponible como una futura línea de mejora.

■ Armonizar estilo con el resto de aplicaciones de la UAB [Usabilidad]

Se ha cambiado la paleta de colores y la distribución de los elementos en la aplicación para que siga la imagen corporativa del resto de herramientas web de la UAB. A su vez se ha aprovechado el espacio en blanco de la aplicación para mostrar más datos en una misma vista.

6 LÍNEAS DE MEJORA

Una vez finalizado el análisis de resultados, en esta sección revisaremos posibles líneas de mejora futuras para posibles ampliaciones de la aplicación.

■ Alertas y notificaciones

Incorporar un sistema de notificaciones *push* o envío de correos electrónicos al acercarse fechas destacadas en el seguimiento mejoraría la experiencia de usuario.

■ Actualizar librerías

Algunas librerías utilizadas como Bootstrap no utilizan las últimas versiones. En el caso de Bootstrap la última versión de la librería (Bootstrap 5) rompe parte de la compatibilidad con la versión anterior pero proporciona nuevas características que podrían ser interesantes en el desarrollo de futuras funcionalidades. Sería necesario reescribir parte del código para poder actualizar las librerías.

■ Soporte multilinguaje

Añadir soporte multilinguaje y traducir la aplicación al castellano e inglés es una posible mejora. Actualmente la aplicación únicamente se encuentra en catalán pero no supone ningún problema.

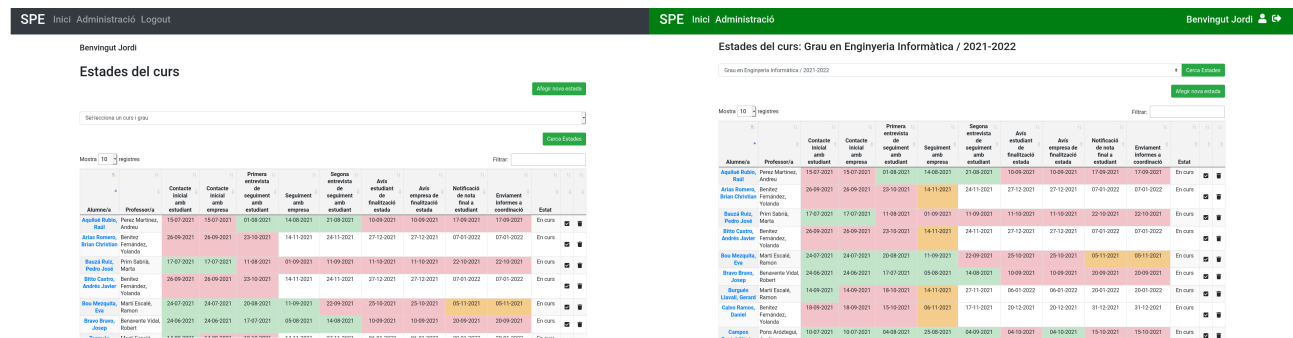


Fig. 17: Comparativa del cambio de colores e imagen corporativa y aprovechamiento del espacio.

■ Desarrollo de una API REST

La incorporación de una API REST que tuviese la mayoría de funcionalidades haría que en un futuro el front-end de la aplicación pudiese reescribirse de forma más sencilla como una SPA.

Esto también permitiría integrar la funcionalidad de la aplicación en otras aplicaciones como por ejemplo una extensión de Moodle.

■ Ampliar la aplicación al proceso de evaluación

Actualmente la aplicación solo se enfoca en el seguimiento de las prácticas. La asignatura Prácticas Externas además de un seguimiento tiene una evaluación que realizan los tutores de la estancia.

El estudiante ha de entregar una memoria de prácticas que luego es evaluada por la persona tutora académica. La persona tutora de la entidad también valora las competencias del estudiante. Todo esto se podría integrar en la herramienta incorporando también el proceso de evaluación de la asignatura.

■ Encuesta de valoración de la estancia

Al final de la estancia de prácticas, los alumnos realizan una encuesta de satisfacción respecto a la estancia en la entidad externa. Esta encuesta podría integrarse en la aplicación, tanto la recogida de datos como su posterior visualización.

■ Módulo estadístico

Sería interesante la incorporación de un módulo que integrase estadísticas respecto a las estancias de los alumnos y se pudiese filtrar por tutor, estancia, empresa... etc. A estas estadísticas también se podrían añadir métricas relacionadas con la evaluación y la valoración de la estancia.

7 CONCLUSIONES

Durante el desarrollo del proyecto se han podido ir completando con éxito la mayoría de objetivos prioritarios descritos al inicio del proyecto. Trabajar ampliando una aplicación desarrollada con anterioridad supone un reto al necesitar llegar a comprender un trabajo ajeno. Además algunos fallos encontrados en la aplicación original han supuesto un contratiempo al necesitar más horas para subsanarlos antes de incorporar nuevas funcionalidades.

La gestión del tiempo ha resultado correcta aunque algunas semanas los hitos previstos se hayan retrasado o no hayan podido cumplirse. La planificación del proyecto ha cambiado sobre todo en la etapa donde se han incorporado nuevas funcionalidades al adaptar la planificación a la división del desarrollo del backend y frontend.

Los objetivos del proyecto relacionados con la seguridad y especialmente prioritarios se han podido completar con éxito. También se han completado mejoras de usabilidad como la reducción del tiempo de interacción con la herramienta y la mejora de determinados formatos como las fechas en las tablas.

Se ha completado el desarrollo de la herramienta de creación de nuevas plantillas. Mediante esta herramienta la aplicación es mucho más flexible para su incorporación en otros grados y su adaptación de un curso a otro.

Se ha puesto en práctica la metodología de desarrollo Kanban que ha permitido una correcta organización del trabajo.

Al iniciar el proyecto se pretendía subir periódicamente los cambios a un servidor proporcionado por el DEIC para poder recibir feedback continuo de la aplicación y así poder realizar un desarrollo iterativo. Debido al ciberataque sufrido en la UAB no se pudo realizar esto y las pruebas en la aplicación por parte del cliente han tenido que moverse al final del desarrollo del trabajo, cuando el servidor ha vuelto a estar disponible.

Aún así, al replicar el desarrollo en local mediante contenedores Docker el desarrollo no se ha visto frenado.

En general ha mejorado la utilidad de la herramienta al ofrecer nuevas funcionalidades y una mejor experiencia de usuario a la hora de realizar determinados procesos relacionados con el seguimiento de las prácticas.

A nivel personal la realización de este proyecto me ha servido para conocer mejor mis habilidades en el ámbito del desarrollo web y mejorar mi gestión del tiempo.

AGRADECIMIENTOS

Me gustaría agradecer a mi tutor Jordi Pons por la posibilidad de realizar este proyecto y la atención recibida durante el mismo.

Así como también agradecer a Adrián Sánchez que ha hecho posible que la aplicación se pueda probar en el servidor del DEIC pese al ciberataque.

REFERENCIAS

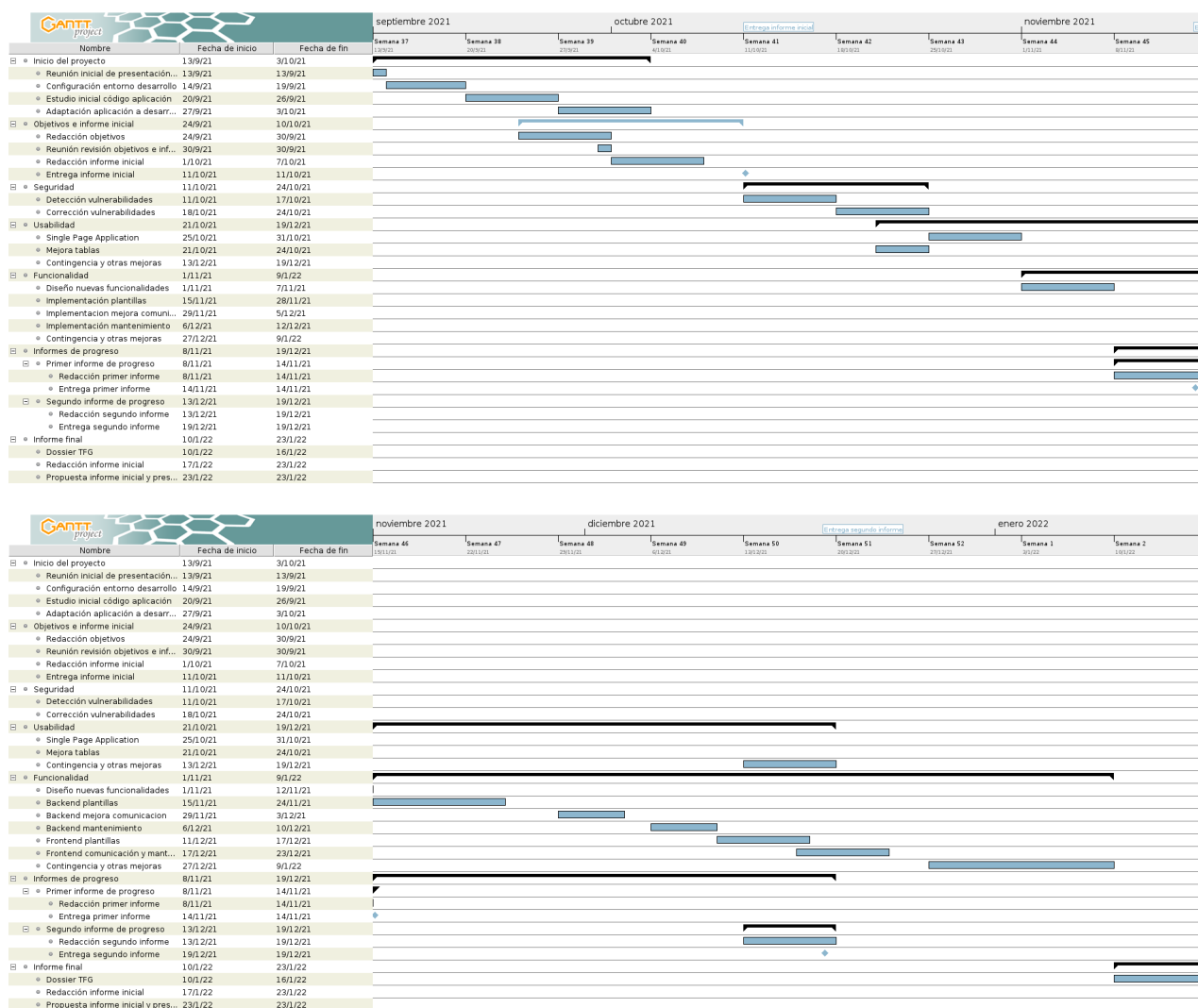
- [1] J. Pons Aróztégui, *Pràctiques Externes. Guia docent.*, 2021-22. <https://ddd.uab.cat/record/243763>.
- [2] Bootstrap, “Bootstrap 4.6 documentation.” <https://getbootstrap.com/docs/4.6/getting-started/introduction/>. Accedido el 8/11/2021.
- [3] O. Martinez Rubio and d. Pons Aróztégui, Jordi, “Desarrollo de una herramienta de seguimiento y tutorización de las prácticas externas (spe),” 2021. <https://ddd.uab.cat/record/248522>.
- [4] Mozilla, “Spa (single-page application).” <https://developer.mozilla.org/en-US/docs/Glossary/SPA>. Accedido el 01/02/2022.
- [5] U. d’Alacant, “Model view controller (mvc).” <https://si.ua.es/en/documentacion/asp-net-mvc-3/1-day/model-view-controller-mvc.html>. Accedido el 01/02/2022.
- [6] V. Farcic, “Software development models: Iterative and incremental development.” <https://technologyconversations.com/2014/01/21/software-development-models-iterative-and-incremental-development/>. Accedido el 5/10/2021.
- [7] Wikimedia, “Iterative process diagram.” https://commons.wikimedia.org/wiki/File:Iterative_Process_Diagram.svg. Accedido el 5/10/2021.
- [8] M. Gant, “Scrum and the solo dev.” <https://medium.com/@jmgant/cleareyeconsulting/scrum-and-the-solo-dev-fb8e810ed42b>. Accedido el 5/10/2021.
- [9] Deloitte, “Kanban vs scrum.” <https://www2.deloitte.com/es/es/pages/technology/articles/kanban-vs-scrum.html>. Accedido el 5/10/2021.
- [10] Trello, “Kanban template.” <https://trello.com/templates/engineering/kanban-template-LGHXvZNL>. Accedido el 8/11/2021.
- [11] Vue.js, “Vue.js documentation.” <https://v3.vuejs.org/guide/introduction.html>. Accedido el 14/12/2021.
- [12] Sqlmap, “Sqlmap wiki.” <https://github.com/sqlmapproject/sqlmap/wiki>. Accedido el 8/11/2021.
- [13] OWASP, “Access control.” https://owasp.org/www-community/Access_Control. Accedido el 14/12/2021.
- [14] Portswinger, “Using burp’s request in browser function to test for access control issues.” <https://portswigger.net/support/using-burp-suites-request-in-browser-function-to-test-for-access-control-issues>. Accedido el 8/11/2021.
- [15] A. A. E., “Vue + summernote editor.” <https://alexanderae.com/vue-summernote-editor.html>. Accedido el 01/02/2022.

APÉNDICE

A TABLA DE OBJETIVOS

Objetivo	Ámbito	Prioridad	Descripción
Revisar y subsanar vulnerabilidades web	Seguridad	Alta	Revisar y corregir posibles vulnerabilidades como SQL Injection y Cross Site scripting.
Corregir seguridad de las peticiones	Seguridad	Alta	Actualmente algunas peticiones de consulta e incluso actualización de datos en la BBDD se pueden realizar sin los permisos necesarios.
Flexibilizar el sistema de seguimiento	Funcionalidad	Alta	Para hacer la aplicación más flexible respecto a su utilización en otros cursos y grados, sería conveniente crear una herramienta que permita crear nuevas plantillas de seguimiento. Al crear un nuevo curso de un grado, se asociaría esta plantilla.
Permitir que el cambio de fechas de inicio y finalización de las estancias se refleje en las fechas de seguimiento	Funcionalidad	Alta	Ahora mismo, al modificar la fecha de fin de una estancia las fechas de seguimiento no se modifican.
Facilitar la comunicación entre alumno y tutor	Funcionalidad	Media	Hay varias funcionalidades a implementar que mejorarían la comunicación. Como la posibilidad de que los alumnos tengan fotos de perfil o que los profesores y coordinadores pudiesen subir documentos adjuntos a los comentarios.
Mejorar la información disponible de empresas y tutores externos	Funcionalidad	Baja	Para listar y modificar tutores externos y empresas es necesario acceder a la BBDD. Añadir al panel de administración la posibilidad de cambiar estos datos.
Reducir el tiempo de interacción con la herramienta	Usabilidad	Alta	Orientar la web hacia una Single Page Application para reducir el número de clicks necesarios. Reducir el tamaño de las peticiones para una carga más ágil.
Facilitar la ordenación de tablas por fechas	Usabilidad	Media	Las tablas que contienen fechas tienen algunos problemas como la ordenación de sus campos.
Dar soporte multilinguaje	Usabilidad	Baja	La aplicación solo esta disponible en catalán, sería interesante añadir soporte multilinguaje en castellano e inglés.
Armonizar estilo con el resto de aplicaciones de la UAB	Usabilidad	Baja	Mejorar el estilo siguiendo la guía de identidad e imagen corporativa de la UAB haría que la aplicación se viese más en armonía con el resto de aplicaciones web.

B PLANIFICACIÓN TEMPORAL DEL PROYECTO



C DIAGRAMA E-R SIMPLIFICADO DE LA BASE DE DATOS

