

Planificación Y Metodología



Directed by:

- Carlos Lobato
- Pablo Gamarro
- Julio Mérida
- Jesús Martín
- Francisco Jiménez
- José Ramón Casero
- Adrián Sevilla

Change Your City

Índice de figuras

1. Metodología	3
2. Descripción del proceso de planificación	3
3. Planificación general del proyecto	5
4. Backlog	6
5. Primer Sprint	6
6. Segundo Sprint	9
7. Tercer Sprint	11
8. Análisis de Riesgos	13

Índice de tablas

1. Tabla de Reuniones y puntos Tratados	4
2. Diagrama de Gantt Planificación proyecto	5
3. Diagrama de Gantt Primer Sprint	8
4. Diagrama de Gantt Segundo Sprint	10
5. Diagrama de Gantt Tercer Sprint	12
6. Tablas Análisis de Riesgos	13
a. Riesgos y tipo de riesgos	13
b. Análisis de riesgos	13
c. Planificación del riesgo	14
d. Indicadores de riesgo	15

Metodología

Utilizaremos la metodología ágil de gestión de proyectos Scrum. Se realizará como mínimo una reunión por semana, donde se establecerán las tareas y se les serán asignadas a subgrupos en función de disponibilidad de horarios y conocimientos.

Nos hemos decantado por esta metodología debido a que somos un grupo de gran tamaño con unos conocimientos acerca de la gestión e implementación de software un poco prematuro y con Scrum de nuestro lado podremos organizarnos mucho mejor y alcanzar nuestros objetivos más eficazmente. El marcarnos fechas límite para entregas y distintos problemas que puedan surgir es una forma de acelerar el procedimiento, y así el cliente va viendo la evolución de las etapas y podrá evaluar si el producto se muestra como él esperaba.

Descripción del proceso de planificación

Puesto que vamos a utilizar el modelo Scrum, nuestra planificación quedará reflejada en sprints. Cada dos semanas se cerrará un ciclo sprint y se empezará uno nuevo. Cada sprint consta de los siguientes apartados:

- Planificación: se parte del backlog y se escogen los trabajos más prioritarios para realizarse en el sprint actual. Usaremos los diagramas de Gantt para mostrar la planificación del sprint en el tiempo del calendario de actividades.
- Desarrollo
- Revisión: conocer qué se ha hecho, qué no se ha hecho, qué problemas surgieron y cómo se han solucionado.
- Valoración y mejora: identificar mejoras y modificaciones a introducir para optimizar los resultados

Se estimará el tiempo necesario para cada trabajo del sprint en función de una persona. Seguidamente, se asignará a cada trabajo el número de personas que se considere adecuado y se hará otra estimación para dicho número de personas. Muchas tareas se ejecutarán concurrentemente. Se tendrá en cuenta que cuantas más personas trabajen en una tarea menor será el tiempo en finalizarla. Sin embargo, también se requerirá de un tiempo necesario para la cooperación y el acuerdo entre los miembros.

La metodología a usar se ha elegido tras realizar la entrevista al cliente, establecer los requisitos y desarrollar la propuesta con el fin de elegir la metodología más adecuada para nuestro proyecto acorde con lo que nos ha pedido el cliente. Es por ello que estas tareas no aparecen reflejadas en los sprints.

Reuniones llevadas a cabo y puntos tratados

Fecha	Integrantes	Modo	Actividad
24-3-18	Fran, Jesús, Jose Ramón, Julio	Virtual	Entrevista
24-3-18	Adrián, Pablo	Virtual	Propuesta
4-4-18	Adrián, Julio	Física	Requisitos
5-4-18	Fran, Jesús, Carlos, Jose Ramón, Pablo	Física	Requisitos, casos de uso, ampliación de entrevista y propuesta, diagrama de Gantt
6-4-18	Fran, Adrián, Julio	Física	Planificación y metodología.
7-4-18	Carlos, Jesús, Jose Ramón	Virtual	Planificación y metodología. Modelado
16-4-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Resolución de <i>issues</i>
23-4-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Diseño del sistema: diagramas de casos de uso y actividades
30-4-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Diseño del sistema: diagrama de clases
07-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
08-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
09-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
14-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
15-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
17-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
21-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Implementación
22-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Testing

24-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Testing
28-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Testing
29-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Física	Testing
31-5-18	Jose, Adrián, Jesús, Francisco, Julio, Carlos, Pablo	Virtual	Finalización del proyecto

Proceso del reparto de tareas

Nos organizaremos dividiendo las tareas en subgrupos para evitar debates innecesarios producidos por la diversidad de opiniones. Nos disponemos a tratar todas las tecnologías todos los miembros del grupo, dado que queremos aprenderlas y esto también facilitará el diseño y la corrección de errores, por lo que en principio no hay un establecimiento de roles. Simplemente dividiremos las tareas y una vez acabadas intercambiaremos los borradores (es decir, las distintas versiones de los archivos finalizadas por un subgrupo) para corregir posibles errores que se hayan pasado por alto, expresiones mal escritas o que podrían derivar en mal entendimiento. Para que el reparto de tareas sea equitativo se tendrá en cuenta la memoria de reuniones llevadas a cabo con las tareas ejercidas, mostrada en el punto anterior.

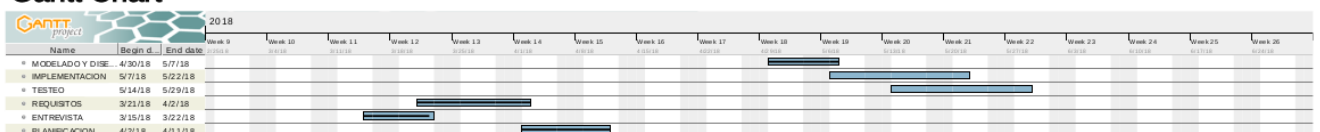
Planificación general a grandes rasgos del proyecto

ChangeYourCity

May 7, 2018

Gantt Chart

3



ChangeYourCity

May 7, 2018

Tasks

2

Name	Begin date	End date
MODELADO Y DISEÑO	4/30/18	5/7/18
IMPLEMENTACION	5/7/18	5/22/18
TESTEO	5/14/18	5/29/18
REQUISITOS	3/21/18	4/2/18
ENTREVISTA	3/15/18	3/22/18
PLANIFICACION	4/2/18	4/11/18

Backlog

1. Como usuario, quiero poder registrarme en la aplicación
 - a. Estimación, 10 días.
2. Como usuario, quiero poder iniciar sesión en la aplicación.
 - a. Estimación, 10 días.
3. Como usuario, quiero disponer de un perfil.
 - a. Estimación 10 días.
4. Como usuario, quiero acceder al conjunto de propuestas de mi zona.
 - a. Estimación 10 días.
5. Como usuario, quiero filtrar las propuestas según la fecha, según la localización o según su clasificación.
 - a. Estimación 5 días.
6. Como usuario, quiero poder iniciar propuestas.
 - a. Estimación 20 días.
7. Como usuario, quiero reportar las propuestas y seleccionar el motivo.
 - a. Estimación 15 días.
8. Como usuario, quiero firmar las propuestas que desee apoyar.
 - a. Estimación 5 días.
9. Como administrador quiero poder hacer envíos de correos electrónicos a los usuarios registrados.
 - a. Estimación 2 días.

Primer Sprint

El primer sprint se va a centrar únicamente en el diseño del sistema. Para diseñarlo realizaremos diagramas de clase e interacción.

Estimación para una persona (con nuestros conocimientos actuales).

- Diagrama de casos de uso (4 días):
 - Registrarse (1 día)
 - Creación de propuestas (1 día)
 - Apoyo de propuestas (1 día)
 - Reportar propuestas (1 día)
 - Verificar desperfectos (1 día)
- Diagrama de actividades (4 días):
 - Registrarse (1 día)
 - Inicio de sesión (1 día)
 - Activar sesión (1 día)
 - Creación de propuestas (1 día)
 - Apoyo de propuestas (1 día)

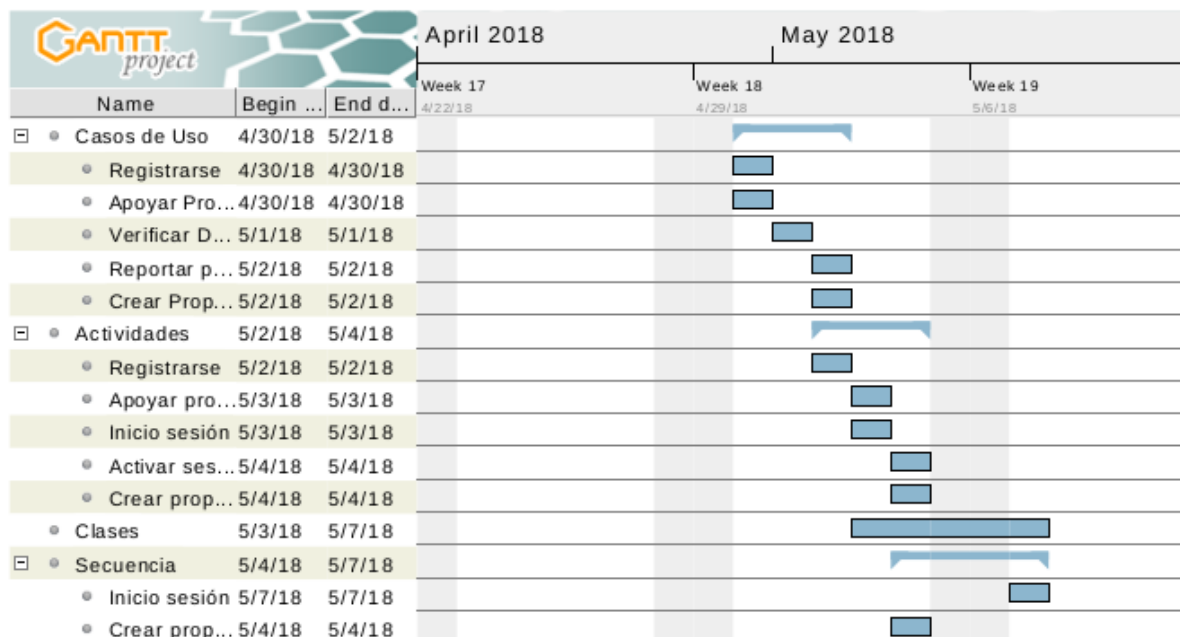
- Diagrama de clases (7 días)
- Diagrama de secuencia (5 días)
 - Inicio de sesión (2 día)
 - Crear propuestas (2 día)

Estimación para el equipo completo (7 personas)

- Diagrama de casos de uso (3 días):
 - Registrarse (1 día)
 - Creación de propuestas (1 día)
 - Apoyo de propuestas (1 día)
 - Reportar propuestas (1 día)
 - Verificar desperfectos (1 día)
- Diagrama de actividades (2 días):
 - Registrarse (1 día)
 - Inicio de sesión (1 día)
 - Activar sesión (1 día)
 - Creación de propuestas (1 día)
 - Apoyo de propuestas (1 día)
- Diagrama de clases (5 días)
- Diagrama de secuencia (4 días)
 - Inicio de sesión (1 día)
 - Crear propuestas (1 día)

Primer Sprint

Gantt Chart



Primer Sprint

Tasks

Name	Begin date	End date
Casos de Uso	4/30/18	5/2/18
Registrarse	4/30/18	4/30/18
Apoyar Propuesta	4/30/18	4/30/18
Verificar Desperfectos	5/1/18	5/1/18
Reportar propuesta	5/2/18	5/2/18
Crear Propuesta	5/2/18	5/2/18
Actividades	5/2/18	5/4/18
Registrarse	5/2/18	5/2/18
Apoyar propuestas	5/3/18	5/3/18
Inicio sesión	5/3/18	5/3/18
Activar sesión	5/4/18	5/4/18
Crear propuestas	5/4/18	5/4/18
Clases	5/3/18	5/7/18
Secuencia	5/4/18	5/7/18
Inicio sesión	5/7/18	5/7/18
Crear propuesta	5/4/18	5/4/18

Este primer sprint se ha realizado sin problemas, se han cumplido los plazos y tenemos los primeros resultados del diseño. Para el siguiente sprint nos proponemos a abordar la implementación. Sin embargo, el cliente nos ha comentado pequeños detalles de los diagramas de actividad que serán corregidos en el segundo sprint mismamente.

Segundo Sprint

El segundo sprint se va a centrar en la implementación del sistema diseñado en el sprint anterior usando NodeJS para el backend. El frontend se desarrollará usando HTML y CSS para finalmente realizar la integración de ambas partes.

En primer lugar, se realizará el login y el sistema de registro tanto en backend como en frontend. Posteriormente, implementaremos el backend general y el frontend general simultáneamente (en diferentes subgrupos de trabajo) con el objetivo final de tener la base de nuestra aplicación web.

Estimación para una persona (con nuestros conocimientos actuales).

- Documentación y pruebas de NodeJS (30 días)
- Documentación y pruebas de HTML y CSS (10 días)
- Desarrollo del login/registro (20 días)
 - Formulario de registro (< 1 día)
 - Backend de registro (12 días)
 - Backend de inicio sesión (12 días)
 - Frontend de registro e inicio de sesión (6 días)
 - Testing (3 días)
- Desarrollo backend general de la aplicación (10 días)
 - Enrutamiento (10 días)
 - Gestión Base de datos (2 día)
 - Controladores (10 días)
 - Modelos (10 días)
 - Implementar vistas (1 día)
- Desarrollo frontend general de la aplicación (8 días)
 - Home (2 días)
 - Proyectos (2 días)
 - Desperfectos (2 días)
 - Sobre nosotros (1 día)
 - Contacto (1 día)

Estimación para el equipo completo (7 personas)

- Documentación y pruebas de NodeJS (16 días)
- Documentación y pruebas de HTML y CSS (5 días)
- Desarrollo del login/registro (11 días)
 - Formulario de registro (< 1 día)
 - Backend de registro (8 días)
 - Backend de inicio sesión (8 días)
 - Frontend de registro e inicio de sesión (2 días)
 - Testing (1 día)

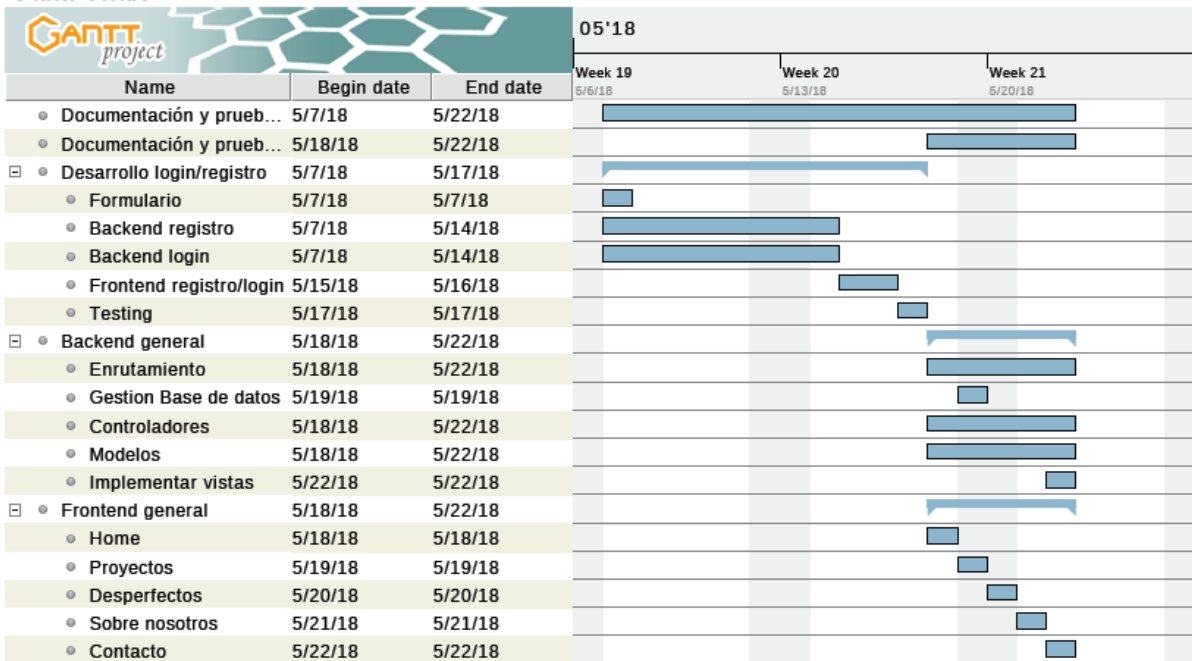
- Desarrollo backend general de la aplicación (5 días)
 - Enrutamiento (5 días)
 - Gestión Base de datos (1 día)
 - Controladores (5 días)
 - Modelos (5 días)
 - Implementar vistas (1 día)
- Desarrollo frontend general de la aplicación (5 días)
 - Home (1 día)
 - Proyectos (1 día)
 - Desperfectos (1 día)
 - Sobre nosotros (1 día)
 - Contacto (1 día)

Change Your City

May 28, 2018

Gantt Chart

3



Tasks

2

Name	Begin date	End date
Documentación y pruebas NodeJS	5/7/18	5/22/18
Documentación y pruebas HTML/CSS	5/18/18	5/22/18
Desarrollo login/registro	5/7/18	5/17/18
Formulario	5/7/18	5/7/18
Backend registro	5/7/18	5/14/18
Backend login	5/7/18	5/14/18
Frontend registro/login	5/15/18	5/16/18
Testing	5/17/18	5/17/18
Backend general	5/18/18	5/22/18
Enrutamiento	5/18/18	5/22/18
Gestion Base de datos	5/19/18	5/19/18
Controladores	5/18/18	5/22/18
Modelos	5/18/18	5/22/18
Implementar vistas	5/22/18	5/22/18
Frontend general	5/18/18	5/22/18
Home	5/18/18	5/18/18
Proyectos	5/19/18	5/19/18
Desperfectos	5/20/18	5/20/18
Sobre nosotros	5/21/18	5/21/18
Contacto	5/22/18	5/22/18

Este sprint se ha resuelto de manera complicada, hemos visto oportuno un cambio de tecnología para el backend: usar NodeJS en vez del framework Spring. El desarrollo de la creación de propuestas ha sido retirado de la planificación dado que nuestras estimaciones iniciales del tiempo requerido para la implementación eran muy optimistas y nos hemos encontrados con problemas a resolver usando las diferentes tecnologías. Esto queda para una ampliación en el futuro. Aun así, hemos acabado el sistema de login y registro y desarrollado todo el frontend.

Para el tercer sprint nos dispondremos a comprobar que no hay fallos en este código.

Tercer Sprint

En el tercer sprint se realizará el testeo de todo el software desarrollado. Para ello, crearemos las pruebas usando Mocha, Supertest, Sinon.js y otras tecnologías aplicables a nuestro proyecto y las ejecutaremos solucionando los errores que se encuentren en nuestra implementación.

Estimación para una persona (con nuestros conocimientos actuales).

- Creación de pruebas (6 días)
- Realización de pruebas (15 días)
- Arreglo de fallos (22 días)
- Documentación pruebas (18 días)

Estimación para el equipo completo (7 personas)

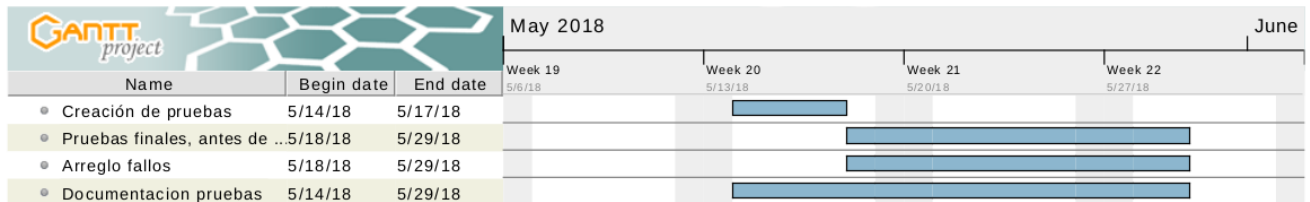
- Creación de pruebas (3 días)
- Realización de pruebas (11 días)
- Arreglo de fallos (11 días)
- Documentación pruebas (15 días)

Change Your City

May 7, 2018

Gantt Chart

3



Change Your City

Tasks

Name	Begin date	End date
Creación de pruebas	5/14/18	5/17/18
Pruebas finales, antes de entrega	5/18/18	5/29/18
Arreglo fallos	5/18/18	5/29/18
Documentacion pruebas	5/14/18	5/29/18

Durante el sprint hemos tenido problemas para realizar pruebas de Mocking dado que es una nueva tecnología y difícil de aplicar a nuestro código. Las pruebas unitarias se han resuelto satisfactoriamente, comprobando que nuestro código de la implementación no presenta fallos en dichas partes. Tras este sprint tenemos la aplicación web de nuestro proyecto, donde los usuarios pueden registrarse e iniciar sesión y acceder a diferentes páginas tales como: desperfectos, proyectos, contacto, etc.

Análisis de riesgos

En este epígrafe procederemos a dar una relación detallada de los posibles riesgos que se pudieran dar durante el desarrollo de nuestro proyecto y la solución que deberíamos dar en el supuesto de darse cada uno de ellos.

Identificación de Riesgos

RIESGOS Y TIPO DE RIESGOS	
Tipos de riesgos	Posibles riesgos
Estimación	1. Se subestima el tiempo necesario para la implementación de cada una de las partes del back-end / front-end.
Tecnológicos	1. La base de datos que usemos no será capaz de soportar un gran número de acciones simultáneas. 2. El código de cada programador es caótico lo que produce un mayor tiempo de desarrollo al dificultar la comprensión.
Personal	1. Alguien que tenía unos conocimientos más avanzados sobre una tecnología enferma y no puede asistir. 2. Por problemas personales una persona que tenía el cometido de aprender una tecnología concreta para la posterior implementación no ha podido cumplir con ello.
Requisitos	1. El cliente de nuestro producto pida un cambio en los requisitos en medio de un sprint. 2. Propuesta de nuevos requisitos que hagan muy complejo su implementación.
Organización	1. Un compañero tiene colaborar en otra tarea que no es la suya porque los compañeros no pueden sacarla adelante, dejando así su tarea desorganizada y con retraso.
Herramientas	1. La herramienta que pasará el esquema de la base de datos a código no produce un resultado correcto.

ANÁLISIS DE RIESGO		
Riesgos	Probabilidad	Efectos
Estimación (1)	Alta	Tolerable
Tecnológicos (1)	Muy baja	Tolerable
Tecnológicos (2)	Moderada	Serio

Personal (1)	Baja	Insignificante
Personal (2)	Baja	Catastrófico
Requisitos (1)	Moderado	Serio
Requisitos (2)	Moderado	Serio
Organización (1)	Muy alta	Tolerable
Herramientas (1)	Muy baja	Serio

Planificación del Riesgo	
Riesgos	Estrategia
Estimación (1)	<ol style="list-style-type: none"> 1. Contratar un programador externo para colaborar e implementar dichas partes a tiempo. 2. Aumentar la productividad en otras secciones para dar más prioridad y personal a esta tarea posteriormente.
Tecnológicos (1)	<ol style="list-style-type: none"> 1. Buscar una base de datos que soporte el número de transacciones de nuestros usuarios 2. Limitar el número de transacciones de los usuarios.
Tecnológicos (2)	<ol style="list-style-type: none"> 1. Instruir a dicho empleado acerca de la importancia de un código limpio y cómo puede influir en el desarrollo y sobre todo en los tiempos de entrega.
Personal (1)	<ol style="list-style-type: none"> 1. Motivar a los empleados a que se documenten mejor sobre esa tecnología y avancen en sus habilidades con ella. 2. Contactar con algún programador conocido para que ayude al grupo con su experiencia. 3. Si es muy urgente contratar a un programador externo para que supla esta experiencia que no tienen los demás.
Personal (2)	<ol style="list-style-type: none"> 1. Dar una oportunidad al empleado y buscar algún tipo de formación que facilite el aprendizaje de la misma. 2. Buscar al empleado que más experiencia tenga en esta tecnología para que pueda abarcar su parte del trabajo.
Requisitos (1)	<ol style="list-style-type: none"> 1. Informar al cliente que cuando estamos en medio de un sprint no podemos reestructurar todo lo que teníamos organizado. 2. Comentar al cliente que si no cambia de idea con respecto a esta parte esto supondrá un aumento en el tiempo de desarrollo y además un aumento en el coste estipulado.

Requisitos (2)	1. Incorporar a alguien a la plantilla con experiencia en ese nuevo campo para colaborar a realizarlo, e informar al cliente del aumento del coste de la aplicación por la contratación del mismo.
Organización (1)	1. Concienciar a los compañeros que han tenido que ser ayudados a que hay que arrimar el hombro y ayudar al damnificado en su desarrollo para llegar a tiempo.
Herramientas (1)	1. Probar a realizar la tarea con otra herramienta distinta, y en el caso de obtener el resultado escribir el código acerca de la creación de la base de datos e inserción de contenido a mano.

Indicadores de riesgo	
Riesgos	Indicador
Estimación (1)	Entrega tardía del código o multitud de informes problemáticos sobre el software.
Tecnológicos (1)	Informes sobre problemas acerca del sobrecalentamiento del hardware hosteador ó notificación sobre demasiado tiempo de espera entre solicitud y solicitud.
Tecnológicos (2)	Incumplimiento de los tiempos estipulados en el back-log y la queja grupal de que el código de un programador específico sea caótico.
Personal (1)	No asistencia al trabajo, recibimiento de mensaje o informe acerca de su estado de salud.
Personal (2)	Disminución de la comunicación del programador con el grupo, incontactable en algún que otro día, desconocimiento del lenguaje técnico de esas tecnologías usadas en el grupo.
Requisitos (1)	Multitud de peticiones del cliente ó insatisfacción en bastantes informes de seguimiento.
Requisitos (2)	
Organización (1)	Incumplimiento de los tiempos estipulados en el back-log y la queja grupal por los retrasos a la hora de entregar código que al final se vuelven inexistentes.
Herramientas (1)	Errores en consultas a la base de datos en la fase de estados.