# Documento de planificación

Contenido

[Documento de planificación 1](#_Toc73719462)

[Identificación de tareas 1](#_Toc73719463)

[1. Creación de la entidad XXX: 1](#_Toc73719464)

[2. Actualizar UML 1](#_Toc73719465)

[3. Poblar base de datos 1](#_Toc73719466)

[4. Back-end funcionalidad de creación de XXX 2](#_Toc73719467)

[5. Front-end funcionalidad de creación de XXX 2](#_Toc73719468)

[6. Back-end funcionalidad de listar XXX 2](#_Toc73719469)

[7. Front-end funcionalidad de listar XXX 2](#_Toc73719470)

[8. Back-end dashboard con XXX 3](#_Toc73719471)

[9. Front-end dashboard con XXX 3](#_Toc73719472)

[10. Comprobación del listado y creación 3](#_Toc73719473)

[11. Comprobación del dashboard 3](#_Toc73719474)

[Observaciones 4](#_Toc73719475)

## Identificación de tareas

### Creación de la entidad XXX:

Necesitamos crear la entidad que se nos pide en los requisitos. Para completar esta tarea, debemos crear la entidad correspondiente y además hacer las modificaciones necesarias en la entidad “shout”, para crear la asociación. Como material de apoyo, se pueden usar las diapositivas del segundo segmento de la asignatura “Devising a WIS”, donde hay instrucciones claras para la creación de entidades y asociaciones.

### Actualizar UML

Debemos plasmar la entidad que se nos pide realizar junto a su asociación y atributos en el diagrama UML. Fijándonos en los requisitos, hay que extender el diagrama UML.

### Poblar base de datos

\*Esta tarea depende de 1, ya que necesita tener la entidad implementada para poderse añadir esta clase de instancias en la base de datos.

Hay que actualizar los elementos del fichero “sample-data.xml” con la nueva información que se nos exige en este ejercicio. Asegúrate de que no haya errores con la relación con “shout”. Para apoyarse, se puede mirar otras entidades que ya están en la base de datos y hacen uso de relaciones, como “manager” y “task”, o incluso mirar en el proyecto “Acme-Jobs”.

### Back-end funcionalidad de creación de XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para crearle un controlador y un servicio.

Hay que crear una funcionalidad de creación de XXX. El desarrollador puede implementar esto como vea más oportuno, pero como manager recomiendo que se implemente todo sobre el servicio y el controlador de “shout”, ya que todo tiene que pertenecer a la misma vista. Para apoyarse, se pueden mirar las transparencias de “Devising a WIS”, especialmente aquellas que explicaban el funcionamiento de los servicios; no copies lo que se dice en las transparencias, es importante entenderlo para poder completar con éxito esta tarea.

### Front-end funcionalidad de creación de XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para poderle añadir funcionalidades, y 4, porque sin la implementación de “unbind”, no se puede completar la vista.

Hay que crear una vista para la funcionalidad de creación de XXX. Debe compartir el mismo formulario que “shout”, es decir, no debemos crear una vista desde cero, sino editar la vista de “shout” para poder aplicar esto. Es posible que haya dificultades a la hora de implementar esto, si fuese necesario, se pueden realizar cambios en el servicio. Para apoyarse, se pueden mirar las transparencias de “Devising a WIS”, especialmente aquellas que explicaban el funcionamiento de los servicios; no copies lo que se dice en las transparencias, es importante entenderlo para poder completar con éxito esta tarea.

### Back-end funcionalidad de listar XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para poderle añadir funcionalidades, y de 3, porque sin la actualización del contenido de la base de datos, no podremos listar apropiadamente la información.

Tenemos que crear una funcionalidad que permita listar los “shouts” con los atributos de la nueva entidad que va asociada a “shout”. El desarrollador puede implementar esto como vea más oportuno, pero como manager recomiendo que se implemente todo sobre el servicio y el controlador de “shout”, ya que todo tiene que pertenecer a la misma vista. Para apoyarse, se pueden mirar las transparencias de “Devising a WIS”, especialmente aquellas que explicaban el funcionamiento de los servicios; no copies lo que se dice en las transparencias, es importante entenderlo para poder completar con éxito esta tarea.

### Front-end funcionalidad de listar XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para poderle añadir funcionalidades, de 3, porque sin la actualización del contenido de la base de datos, no podremos listar apropiadamente la información, y de 6, porque sin la implementación de “unbind”, no se puede completar la vista.

Hay que crear una vista para la funcionalidad de listar los “shouts” con su XXX; esto quiere decir que debemos editar la vista ya existente, no crear una vista nueva. Es posible que haya dificultades a la hora de implementar esto, si fuese necesario, se pueden realizar cambios en el servicio. Para apoyarse, se pueden mirar las transparencias de “Devising a WIS”, especialmente aquellas que explicaban el funcionamiento de los servicios; no copies lo que se dice en las transparencias, es importante entenderlo para poder completar con éxito esta tarea.

### Back-end dashboard con XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para poder extraer datos de sus instancias, y es preferible que 3 también esté hecho para poder trabajar al menos sobre las instancias que haya en la base de datos, pero no es necesario.

Tenemos que editar las funcionalidades de “dashboard” actualmente existentes para que también incluya información de XXX. Para más información, se pueden consultar los requisitos. Si no se sabe cómo implementar alguna de las funciones que nos piden, recomiendo que se consulte en internet. Algunas de las funciones ya están implementadas para otras entidades, sugiero mirar cómo se implementaron para ellas, y replicar el procedimiento. Si hubiera algún otro problema, recomiendo que se consulte el cómo se ha formado el “dashboard” actual o cómo se ha hecho el de “Acme-Jobs”.

### Front-end dashboard con XXX

\*Esta tarea depende de 1, ya que es necesario que exista la entidad para poder extraer datos de sus instancias, y es preferible que 3 también esté hecho para poder trabajar al menos sobre las instancias que haya en la base de datos, pero no es necesario. También necesitamos tener 8 hecho porque hace falta tener la funcionalidad hecha para poder visualizar dicha información en la vista.

Tenemos que editar la vista de “dashboard” actualmente existente para que también incluya información de XXX. Para más información, se pueden consultar los requisitos. Para poder ver cómo realizar la vista de “dashboard”, recomiendo observar lo que ya se ha hecho para nuestro proyecto y replicarlo. Si hubiera algún otro problema, recomiendo que se consulte el cómo se ha formado el “dashboard” actual o cómo se ha hecho el de “Acme-Jobs”.

### Comprobación del listado y creación

\*Esta tarea depende de todas las tareas relacionadas con la creación de la entidad, listar las instancias de la entidad y creación de nuevas instancias de la entidad.

Una vez terminadas las tareas relacionadas con las funcionalidades básicas de la nueva entidad, necesitamos que alguien ajeno a las personas que realizaron estas tareas las revisen en búsqueda de fallos. Si los fallos localizados son pequeños, se pueden solucionar durante la revisión, pero se debe tomar nota de estos fallos para tenerlo en cuenta. Si se localizan fallos más complejos, debe tomar nota de ellos, y se crearán nuevas tareas o se reabrirán tareas antiguas en base a ellos. Un fallo que se considere pequeño debe poderse resolver en media hora como máximo. Si se ha intentado resolver un error y se ha tardado más de 30 minutos, se deben revertir los cambios y notificar el error como un error complejo.

Se debe prestar especial atención a las validaciones. Realice las pruebas con los requisitos por delante, y pruebe las restricciones una a una.

### Comprobación del dashboard

\*Esta tarea depende de todas las tareas relacionadas con la creación de la entidad, creación de nuevas instancias de la entidad y de mostrar el “dashboard” con información relacionada con la entidad.

Una vez terminadas las tareas relacionadas con las funcionalidades básicas de la nueva entidad, necesitamos que alguien ajeno a las personas que realizaron estas tareas las revisen en búsqueda de fallos. Si los fallos localizados son pequeños, se pueden solucionar durante la revisión, pero se debe tomar nota de estos fallos para tenerlo en cuenta. Si se localizan fallos más complejos, debe tomar nota de ellos, y se crearán nuevas tareas o se reabrirán tareas antiguas en base a ellos. Un fallo que se considere pequeño debe poderse resolver en media hora como máximo. Si se ha intentado resolver un error y se ha tardado más de 30 minutos, se deben revertir los cambios y notificar el error como un error complejo.

Es necesario que probemos que las funciones del “dashboard” están funcionales. Se deben realizar cambios en las instancias, creando nuevas, y se debe estimar cómo esperamos que cambien los valores mostrados en el “dashboard”; si los valores varían como deben, entonces podemos suponer que nuestro “dashboard” funciona correctamente, en caso contrario, se debe localizar el error, que puede ser de la implementación o de la estimación. Recomiendo que cada vez que encontremos algo incoherente, antes de asumir que es un error de la implementación, revisemos que no se trata de un error en la estimación.

# Observaciones

El reparto de tareas debe hacerse de tal forma que o bien dos personas que trabajen juntas o una misma persona sea la(s) que tenga(n) las tareas de front-end y back-end de la misma funcionalidad, para facilitar la coordinación y detección de dificultades entre las tareas. Lo mismo ocurre con la creación de la entidad, actualizar el diagrama UML y actualizar el archivo “sample-data.xml”. Por último, las dos tareas de revisión debe hacerlas un miembro que no haya tenido que ver en las tareas que se están revisando, a ser posible.