

TFC GESTIÓN DE PROYECTOS

# relaPro:

Base de Datos para encontrar relaciones entre Proyectos

---

*Autor: Julio Fortis Urbano*

*Consultor: Ana Cristina Domingo Troncho*

---

## RESUMEN

Este documento es una memoria presentando los procesos seguidos y los resultados conseguidos para el diseño e implementación de una Base de Datos Relacional capaz de almacenar proyectos y su progreso que he realizado en el marco del área de Gestión de Proyectos.

Mi proyecto consiste en el desarrollo de una estructura relacional que permite el almacenamiento contextual de todos los elementos de dichos proyectos, de forma que se disponga de toda la información de un proyecto de forma estructurada y se puedan encontrar relaciones con otros proyectos almacenados.

Para el desarrollo de esta Base de Datos Relacional he utilizado PostgreSQL. Los datos de ejemplo vienen visualizados mediante una aplicación de ejemplo desarrollada en HTML, php y CSS.

## ÍNDICE

RESUMEN .....	2
ÍNDICE .....	3
INTRODUCCIÓN.....	7
JUSTIFICACIÓN .....	9
DESCRIPCIÓN.....	11
ENFOQUE Y MÉTODO SEGUIDOS.....	12
DESARROLLO .....	12
DOCUMENTACIÓN .....	12
ETAPA PREVIA AL PROYECTO .....	13
BUSINESS CASE.....	13
Resumen Ejecutivo.....	13
Identificación de la oportunidad.....	13
Evaluación inicial de la tecnología disponible.....	14
Evaluación de las capacidades propias .....	14
Identificación de los riesgos principales .....	15
Objetivos y contenidos del proyecto. Visión preliminar .....	15
Evaluación inicial del tiempo .....	16
CRITERIOS DE GESTIÓN DE PROYECTOS PARA SELECCIONARLO .....	17
FASE DE INICIACIÓN DEL PROYECTO.....	18
PROJECT CHARTER .....	18
Propósito y justificación.....	18
Objetivos y criterios de éxito .....	19
Requisitos.....	20
Descripción del proyecto .....	20
Riesgos .....	21
Resumen del cronograma de los hitos.....	21
Requisitos de aprobación del proyecto, quien lo aprueba y quien lo entrega.....	22
El director nombrado, responsabilidades y nivel de autoridad.....	22
Otros participantes o afectados por el proyecto y sus responsabilidades .....	22
El patrocinador y su nivel de autoridad .....	22
STAKEHOLDERS .....	23

Sponsor .....	23
Otros .....	23
DEFINICIÓN INICIAL DE ALCANCE .....	25
Objetivos .....	25
Características del producto .....	25
Entregables del Proyecto .....	25
Requisitos.....	26
Criterios de aceptación del producto o servicio .....	26
Fronteras del proyecto o relaciones con otros proyectos .....	26
Hitos .....	27
EDT Inicial.....	27
FASE DE PLANIFICACIÓN DEL PROYECTO.....	30
PLAN DE GESTIÓN DEL PROYECTO.....	30
Objetivos y Resultados.....	30
Hitos .....	30
Actividades.....	31
Calendario .....	32
Riesgos .....	32
Calidad.....	33
PLAN DE GESTIÓN DEL ALCANCE .....	34
Recogida de Requisitos .....	34
Definición Detallada del Alcance .....	34
Creación de la EDT .....	36
PLAN DE GESTIÓN DEL TIEMPO .....	38
Definición de las actividades.....	38
Secuenciación de las actividades .....	39
Estimación de la duración de las actividades.....	39
Elaboración del calendario.....	40
PLAN DE CALIDAD .....	43
Criterios de Calidad .....	43
Declaración de Calidad.....	43
PLAN DE GESTIÓN DE RIESGOS .....	45
Clasificación de los Riesgos .....	45

Identificar los Riesgos .....	46
Análisis Cualitativo de los Riesgos.....	46
Planificación de la Respuesta.....	47
FASE DE EJECUCIÓN DEL PROYECTO.....	48
INFORME DE LANZAMIENTO .....	48
GESTIÓN DE INCIDENCIAS.....	49
GESTIÓN DE CAMBIOS .....	50
GESTION DEL CALENDARIO.....	51
FASE DE CIERRE DEL PROYECTO.....	53
CICLO DE DESARROLLO DEL PRODUCTO.....	54
FASE DE ANÁLISIS.....	54
Análisis de Requisitos.....	54
Modelado del Sistema .....	59
Técnicas de Obtención de Requisitos utilizadas .....	61
Casos de uso.....	61
FASE DE DISEÑO.....	64
Diseño preliminar.....	64
Diseño detallado .....	67
Calidad del software .....	79
FASE DE CODIFICACIÓN .....	80
FASE DE PRUEBAS .....	81
Pruebas de Caja Blanca.....	81
Pruebas de Caja Negra .....	82
FASE DE MANTENIMIENTO.....	83
DOCUMENTACIÓN DEL PRODUCTO FINAL .....	84
INSTALACIÓN .....	84
Base de Datos.....	84
Interfaz Web de Prueba .....	84
CONCLUSIONES.....	85
PROPUESTA DE MEJORA Y LÍNEAS DE FUTURO .....	86
AUTOEVALUACIÓN.....	88
RECURSOS UTILIZADOS.....	89
BIBLIOGRAFÍA.....	90



## INTRODUCCIÓN

He realizado un proyecto de diseño e implementación de una Base de Datos Relacional que permite almacenar proyectos y su progreso. Con el nombre de “**relaPro**” quiero hacer referencia a “*Proyectos Relacionados*” o “*Relación entre Proyectos*”, una referencia tanto al carácter relacional de la Base de Datos como a la búsqueda de elementos relacionados en otros proyectos.

A diferencia de otras alternativas en el mercado, mi idea principal ha sido el almacenamiento contextual de todos los diferentes elementos de dichos proyectos, de forma que no sólo se disponga de toda la información de un proyecto de forma estructurada, sino que además se pudiesen comparar diferentes proyectos entre ellos para permitir al Project Manager buscar posibles causas comunes de éxito o fracaso, lo que denomino proyectos relacionados.

También quiero recalcar que, desde el punto de vista metodológico, considero que realizar un TFC en el área de Gestión de Proyectos y no tratarlo seriamente como un Proyecto parece un error conceptual de base. Por eso esta memoria viene dividida en función de las distintas fases del ciclo de vida del proyecto.

Así, para la realización del Proyecto, he seguido la metodología planteada en los materiales de la asignatura de Gestión de Proyectos, por lo que se basa principalmente en las metodologías presentadas en el Project Management Body of Knowledge (actual estándar de facto para la industria) y parcialmente en el Goal Directed Project Management de Andersen.

Por otro lado, tratándose de un ejercicio académico, no he entrado en el detalle de los costes asociados al proyecto (ni materiales ni de recursos humanos). No obstante, si se desearan conocer estos, bastaría tomar en consideración las horas de trabajo de su autor y el coste del material utilizado (ordenador personal) para calcularlos.

## JUSTIFICACIÓN

El propósito de este TFC es el desarrollo de una estructura que nos permita almacenar proyectos, realizar su seguimiento y compararlos entre ellos. Esto permitirá al Director de Proyectos obtener información contextualizada sobre su proyecto y realizar previsiones intentando reducir la incertidumbre.

Algunas de las preguntas que se podrían responder con este sistema son:

- ¿Sufrieron retrasos todos los proyectos que incluyeron a una determinada persona?
- ¿Un promotor cambia siempre los requisitos en el último momento?
- ¿Tienden a fracasar los proyectos que se retrasaron en la fase de planificación?

Para poder llegar a dichas conclusiones, se ha desarrollado por un lado una Base de Datos capaz de almacenar nuestros proyectos de la manera más contextualizada posible, por otro una aplicación de ejemplo que permite visualizar dichos datos mostrándonos un ejemplo de “dashboard” con el que realizar nuestras previsiones.

Este sistema también ofrece ventajas respecto a las Bases de Datos de proyectos tradicionales. Al mostrarnos el sistema automáticamente otros proyectos relacionados, sabremos directamente que proyectos históricos consultar (ya que guardan relación directa con el nuestro), mientras que en un sistema tradicional deberíamos revisar toda la documentación histórica buscando las relaciones, cosa a todas luces poco práctica.

Por último, se desea generar una estructura con elementos altamente reusables. Un proyecto que no se reutiliza, es un proyecto muerto. Mi idea es desarrollar una estructura que se pueda reutilizar en otras aplicaciones de Gestión de Proyectos.

## DESCRIPCIÓN

Proyecto de desarrollo de una Base de Datos Relacional que permita almacenar proyectos y compararlos entre ellos. Debe permitir el almacenamiento progresivo de la documentación de los proyectos, para permitir así su seguimiento por parte del Director de Proyecto mediante una aplicación de prueba, así como permitir reutilizar la información ya introducida para otros proyectos que sea aplicable (por ejemplo stakeholders, sponsors... ) minimizando la repetición de información.

El Proyecto se compone del análisis y desarrollo de una estructura de Base de Datos Relacional que permita almacenar proyectos y compararlos entre ellos.

Para ello, se identificarán todas las entidades de un Proyecto TIC genérico y se codificarán mediante tablas relacionales en dicha BDR, buscando minimizar la información repetida.

También se desarrollará una aplicación web de ejemplo que utilizando dicha estructura, permita visualizar un Proyecto y compararlo con otros ya existentes en base a criterios comunes.

Por último se generará toda la documentación del Proyecto, así como una presentación audiovisual para el Tribunal de evaluación.

## ENFOQUE Y MÉTODO SEGUIDOS

De manera general, he utilizado un ciclo de desarrollo iterativo e incremental basado en el ciclo clásico (en cascada), separando cada módulo en las etapas de Análisis, Diseño, Codificación y Prueba. Al final del proyecto, debido a las limitaciones de tiempo he optado por un desarrollo más cercano a la programación exploratoria para terminar el producto dentro del plazo establecido.

## DESARROLLO

No se ha utilizado ningún entorno de desarrollo específico, la base de datos se ha creado directamente en PostgreSQL utilizando la herramienta pgAdmin III. Para el desarrollo web se ha utilizado el bloc de notas, php e Internet Information Server en un ordenador Windows.

## DOCUMENTACIÓN

Durante la Evaluación Continua, he producido una serie de documentación (pruebas, búsquedas de información, diagramas...) que ha cumplido tanto una función de control como una labor preparativa para la memoria final. Este documento agrupa la mayor parte de dicha documentación para facilitar su consulta y seguimiento.

Según se ha ido detallando el proyecto, ha sido necesario modificar y ampliar esta documentación, por lo que su redacción ha sido iterativa hasta llegar a la última fase del proyecto que ha sido la redacción y construcción final de esta memoria.

## ETAPA PREVIA AL PROYECTO

### BUSINESS CASE

#### Resumen Ejecutivo

Proyecto de desarrollo de una Base de Datos Relacional que permita almacenar proyectos y compararlos entre ellos. Debe permitir el almacenamiento progresivo de la documentación de los proyectos, para permitir así su seguimiento por parte del Director de Proyecto mediante una aplicación de prueba, así como permitir reutilizar la información ya introducida para otros proyectos que sea aplicable (por ejemplo stakeholders, sponsors... ) minimizando la repetición de información.

#### Identificación de la oportunidad

Si bien existen en el mercado numerosas aplicaciones de gestión de proyectos (como MS Project o Project Libre), estas normalmente se orientan a la planificación temporal, de las tareas y los recursos y no a la gestión documental. Igualmente, estas aplicaciones suelen tratar cada proyecto como un documento aislado, no permitiendo encontrar fácilmente elementos comunes entre distintos proyectos.

Por otro lado, otros TFC desarrollados en este campo, suelen centrarse en la gestión del proyecto en sí (sus aspectos colaborativos) o simplemente en el almacenamiento histórico, por lo que un aspecto diferenciador de este TFC será la reutilización y la comparación.

## Evaluación inicial de la tecnología disponible

En la actualidad, existen numerosas tecnologías de Bases de Datos relacionales que permiten la implementación de la estructura deseada. Igualmente casi cualquier lenguaje de programación permitirá extraer información fácilmente de dicha Base de Datos permitiendo el desarrollo de una aplicación de prueba que demuestre el funcionamiento de dicha Base de Datos.

Entre las alternativas disponibles, la idea inicial será utilizar una Base de Datos PostgreSQL (gratuita, orientada al objeto y de código abierto) para implementar la susodicha Base de Datos y desarrollar una aplicación web a medida con PHP/HTML.

## Evaluación de las capacidades propias

Para la realización de este proyecto, serán necesarias las siguientes capacidades:

- Conocimientos de Gestión de Proyectos y de la documentación relacionada
- Conocimientos de Análisis e Ingeniería del Software, para generar el modelo conceptual a utilizar
- Conocimientos del lenguaje SQL, para la creación de la Base de Datos que implementará el modelo
- Conocimientos de programación en PHP/HTML, para la creación de la aplicación de demostración que utilizará la implementación

Haciendo un pequeño ejercicio de autoevaluación, considero que dispongo de todas las capacidades anteriormente expuestas, por lo que no será necesario adquirir nuevas habilidades durante la realización del proyecto.

### Identificación de los riesgos principales

El riesgo o amenaza principal en este proyecto es el tiempo. Sin la dedicación apropiada, fácilmente puede sobrepasarse la fecha límite sin haber podido desarrollar todos los aspectos deseados del producto. En un trabajo de estas características, esto no es aceptable.

### Objetivos y contenidos del proyecto. Visión preliminar

Los objetivos principales a alcanzar son:

- Identificar todos los elementos únicos de un proyecto y desarrollar una estructura lógica apropiada para su implementación en una Base de Datos relacional.
- Realizar una aplicación de ejemplo que permita visualizar todos los datos un proyecto y comparar estos con otros proyectos almacenados en la Base de Datos.
- Documentar todo el proceso como framework lógico de manera que pueda ser reutilizado posteriormente en otras aplicaciones.
- Realizarlo en los tiempos establecidos por el Plan de Estudios y con la mayor calidad obteniendo la mejor nota posible.

Así, el producto final se deberá componer de:

- Una estructura de Base de Datos

- Datos de ejemplo
- Una aplicación web de ejemplo
- Una presentación audiovisual del producto
- Documentación

### Evaluación inicial del tiempo

El proyecto contará en total con un total de 16 semanas, siendo inamovibles tanto el inicio (26/2/2015) como el final (18/6/2015).

## CRITERIOS DE GESTIÓN DE PROYECTOS PARA SELECCIONARLO

En este proyecto, a diferencia de las otras alternativas planteadas previamente al consultor, se considera que tanto los objetivos y resultados, como el alcance y los riesgos están bien definidos. Se cuenta con todas las capacidades adecuadas para realizarlo y el proyecto tendrá la dedicación necesaria, por lo que se ha considerado la mejor de las opciones iniciales.

Además, aunque en el plano personal me resultase atractiva la idea de realizar un Gestor de Proyectos orientado a otra disciplina (Ciencias del deporte, Arqueología, Antropología, Ciencias Forenses...), en realidad el producto final sería simplemente gestor con una estructura de proyecto determinada, siendo la mayor parte del desarrollo inicial identificar los rasgos determinantes de un Proyecto en dicha disciplina. Siendo este TFC de Gestión de Proyectos y no de programación, he considerado más interesante centrarme en el propio campo del Project Management y desarrollar un producto que se adapte a dicho cuerpo de conocimiento y que mi tiempo se dedique a profundizar en los conocimientos sobre las propias prácticas de gestión de estos.

## FASE DE INICIACIÓN DEL PROYECTO

### PROJECT CHARTER

#### Propósito y justificación

El propósito del Proyecto es obtener un producto que nos permita almacenar proyectos, realizar su seguimiento y compararlos entre ellos. Un producto con estas características permitirá al Director de Proyectos obtener información contextualizada sobre su proyecto y realizar previsiones gracias a ello. ¿Sufrieron retrasos todos los proyectos que incluyeron a una determinada persona? ¿Un promotor cambia siempre los requisitos en el último momento? ¿Tienden a fracasar los proyectos que se retrasaron en la fase de planificación?

Para poder llegar a dichas conclusiones, se necesitará desarrollar por un lado una Base de Datos capaz de almacenar nuestros proyectos de la manera más contextualizada posible, por otro una aplicación de ejemplo que nos permita visualizar dichos datos mostrándonos un ejemplo de Dashboard con el que realizar nuestras previsiones.

También se desea generar un corpus documental adaptado al Proyecto, y de acuerdo con los requisitos de la asignatura, que ilustre todo el proceso de desarrollo tanto del producto como del propio proyecto.

Por último, como justificación, no hay que olvidar que al utilizar una estructura con elementos altamente reusables, se ayuda a las organizaciones orientadas comercialmente a replicar los proyectos de éxito en diversos clientes. Ayudar además a realizar previsiones, favorece el control de la evolución de los proyectos, ayudando a minimizar algunas de las principales causas de fracaso de un proyecto.

## Objetivos y criterios de éxito

### *Objetivos principales*

- Analizar y desarrollar una estructura lógico que permita almacenar un Proyecto en una Base de Datos Relacional reutilizando el máximo de elementos ya presentes en la Base de Datos.
- Realizar una aplicación web de ejemplo que permita visualizar todos los datos un proyecto y comparar estos con otros proyectos almacenados en la Base de Datos buscando puntos en común con proyectos pasados.
- Documentar todo el proceso como de manera que el trabajo realizado pueda ser reutilizado posteriormente en otras aplicaciones.
- Realizarlo en los tiempos establecidos por el Plan de Estudios y con la mayor calidad obteniendo la mejor nota posible.

### *Criterios de éxito*

- Finalizar el proyecto en el tiempo establecido
- La estructura de la Base de Datos permite almacenar un Proyecto estándar
- La aplicación web permite visualizar un Proyecto introducido en la Base de Datos
- La aplicación web permite visualizar Proyectos con elementos comunes al Proyecto seleccionado

- Tanto el Consultor como el Tribunal están satisfechos con el producto final, obteniéndose la mejor calificación posible.

## Requisitos

Como normal general, el producto deberá ajustarse a los estándares mínimos establecidos por el Consultor, así como a la planificación temporal establecida por el Plan de estudios.

Siendo además un trabajo académico, deberá ser un producto propio, desarrollado de manera individual por el alumno y ajustarse a todas las normas éticas y legales apropiadas.

## Descripción del proyecto

El Proyecto se compone del análisis y desarrollo de una estructura de Base de Datos Relacional que permita almacenar proyectos y compararlos entre ellos.

Para ello, se identificarán todas las entidades de un Proyecto TIC genérico y se codificarán mediante tablas relacionales en dicha BDR, buscando minimizar la información repetida.

También se desarrollará una aplicación web de ejemplo que utilizando dicha estructura, permita visualizar un Proyecto y compararlo con otros ya existentes en base a criterios comunes.

Por último se generará toda la documentación del Proyecto, así como una presentación audiovisual para el Tribunal de evaluación.

## Riesgos

El principal riesgo a tener en cuenta es el tiempo, ya que la fecha final de entrega del Proyecto es inamovible. En caso de que se produzcan retrasos, habrá que aumentar la dedicación para compensarlos y conseguir mantener la planificación dentro de lo establecido.

Otros riesgos a tener en cuenta son que el proyecto no se valore positivamente, debido a que los entregables no se ajusten a los estándares del Consultor o el tribunal, para intentar minimizar este riesgo, se deberá trabajar de con el Consultor en todo momento para asegurarse que el proyecto se ajusta a lo esperado.

## Resumen del cronograma de los hitos

En base a las entregas a realizar, podemos identificar inicialmente los siguientes hitos:

1. **Plan de Trabajo** (del 26/2/2015 al 18/3/2015)
2. **PAC2** (del 19/3/2015 al 22/4/2015)
3. **PAC3** (del 23/4/2015 al 21/5/2015)
4. **Entrega Final** (del 22/5/2015 al 18/6/2015)

### Requisitos de aprobación del proyecto, quien lo aprueba y quien lo entrega

El Proyecto deberá ser aprobado por el Consultor, Ana Cristina Domingo Troncho, y valorado en base a su opinión por un Tribunal de evaluación. La entrega será responsabilidad en todo momento del alumno responsable del proyecto.

### El director nombrado, responsabilidades y nivel de autoridad

Yo, Julio Fortis Urbano, en calidad de alumno de la asignatura será el único responsable del Proyecto, debiendo realizar la totalidad del trabajo de forma individual.

### Otros participantes o afectados por el proyecto y sus responsabilidades

Otros afectados por el proyecto serán el Tribunal de evaluación, que emitirá el juicio final sobre el TFC, Ana Cristina Domingo Troncho, como Consultor de la asignatura, y mi actual empleador, ya que siendo trabajador por cuenta ajena además de estudiante, mi disponibilidad podría verse afectada por cambios en mi carga de trabajo.

### El patrocinador y su nivel de autoridad

Originalmente había pensado en el Consultor de la asignatura como Sponsor en el Proyecto, pero finalmente seré yo mismo como principal interesado en este TFC el que tendrá que actuar como cliente, estableciendo mis propios objetivos y requisitos y pudiendo solicitar los cambios al Proyecto que considere apropiados para el mejor desarrollo de este.

## STAKEHOLDERS

Tratándose de un Proyecto académico, los Stakeholders son reducidos. Tal como se definió en el Project Charter, básicamente todo se reduce a mí mismo y a unos actores secundarios que se definirán a continuación.

### Sponsor

- **Julio Fortis Urbano**, principal interesado
  - Grado de influencia: Alto
  - Cooperación: Necesaria
  - Grado de Impacto: Alto

### Otros

- **Ana Cristina Domingo Troncho**, Consultor de la asignatura
  - Grado de influencia: Alto
  - Cooperación: Necesaria
  - Grado de Impacto: Alto
- **Tribunal de evaluación**
  - Grado de influencia: Bajo
  - Cooperación: No necesaria
  - Grado de Impacto: Alto
  - Notas: Se deberá responder a sus preguntas tras la entrega del TFC
- **Mi actual dador de trabajo**

- Grado de influencia: Alto
- Cooperación: No necesaria
- Grado de Impacto: Bajo
- Notas: Si aumentase mi carga de trabajo, podría influir negativamente en la disponibilidad de tiempo de dedicación al Proyecto

## DEFINICIÓN INICIAL DE ALCANCE

### Objetivos

Antes del 18 junio 2015 se habrá debido alcanzar los siguientes objetivos:

1. Disponer de una Base de Datos operativa con algunos proyectos de ejemplo
2. Disponer de una interfaz web mínima que permita visualizar los datos y realizar comparaciones básicas entre proyectos
3. Realizar una presentación audiovisual del funcionamiento para el tribunal
4. Finalizar la documentación del proyecto

### Características del producto

El producto final se compondrá de los siguientes elementos:

1. Base de Datos Relacional
2. Datos de ejemplo
3. Aplicación Web mínima de demostración
4. Presentación audiovisual
5. Documentación

### Entregables del Proyecto

Durante el desarrollo del proyecto, se producirán los siguientes materiales:

- Documentación del TFC en PDF (incluyendo los documentos parciales para la EC)
- Archivo SQL con la estructura de la BD

- Archivo SQL con datos de ejemplo
- Archivo ZIP con el sitio web mínimo para visualizar los datos de ejemplo (archivos PHP)
- Presentación audiovisual del proyecto para el tribunal en video y PowerPoint

## Requisitos

El Proyecto deberá cumplir con todos los requisitos establecidos en el plan de estudios de la asignatura así como todos los adicionales que establezca el consultor a lo largo del desarrollo de la asignatura.

## Criterios de aceptación del producto o servicio

Por un lado, el consultor deberá aceptar el desarrollo del proyecto mediante la establecida Evaluación Continua de la asignatura, por otro lado la valoración final del proyecto la llevará a cabo el tribunal de evaluación siguiendo las recomendaciones del consultor una vez entregado la totalidad del Proyecto.

## Fronteras del proyecto o relaciones con otros proyectos

En este momento no se han detectado relaciones de dependencia con otros proyectos preexistentes, aunque se han encontrado algunos trabajos del repositorio institucional de la UOC que podrán ser utilizados como fuente de consulta durante la futura fase de planificación.

Una vez finalizado este TFC, no se descarta desarrollos posteriores para ampliar este marco conceptual inicial en proyectos reales (por ejemplo una futura aplicación móvil que implemente la estructura definida en este proyecto en un formato comercializable).

## Hitos

En este momento se pueden identificar los siguientes hitos en el desarrollo del Proyecto:

1. Final de la Fase de Iniciación (y fases previas) -> PAC1 (18/3/2015)
2. Final de la Fase de Planificación -> PAC2 (22/4/2015)
3. Final de las Fases de Ejecución, Seguimiento y Control -> PAC3 (21/5/2015)
4. Final de la Fase de Cierre -> Entrega Final (18/6/2015)

## EDT Inicial

La división inicial propuesta del trabajo sería la siguiente:

### 1) Base de Datos Relacional

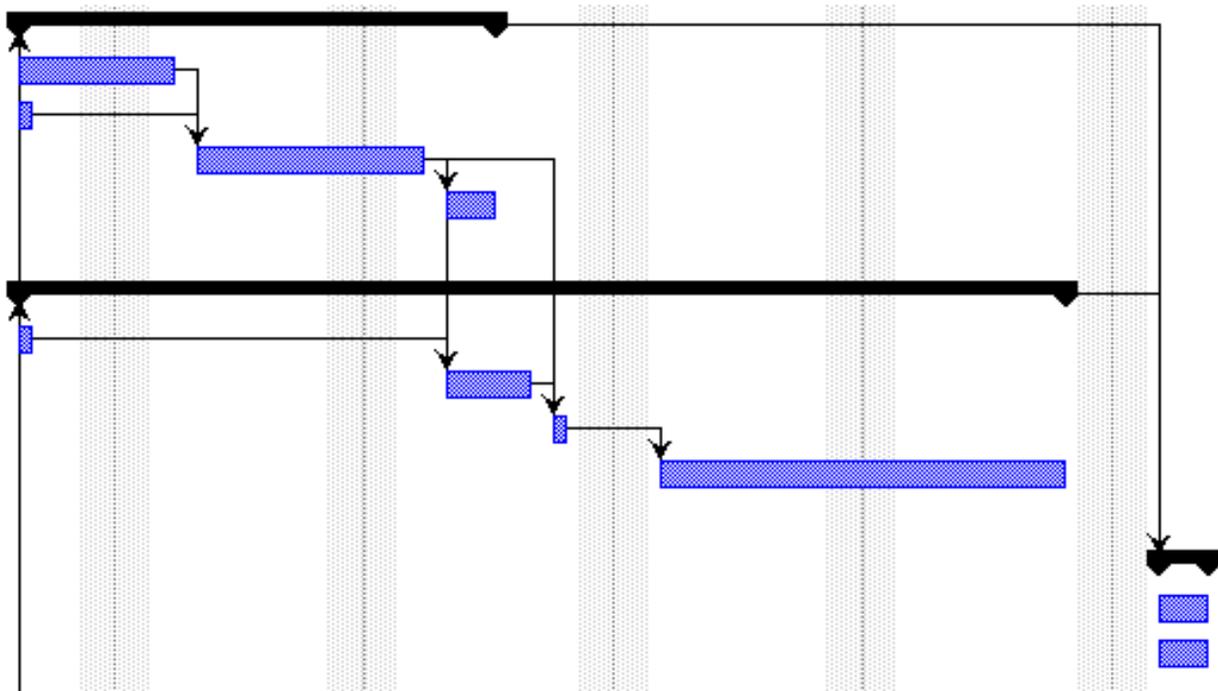
- a. Identificación de las Clases (¿De qué elementos se compone un proyecto?)
- b. Preparación del entorno de trabajo (Instalar gestor de base de datos)
- c. Diseño de la estructura de la Base de Datos (Implementar los elementos identificados)
- d. Introducción de datos de ejemplo (Introducir información que se pueda ver en la web)

### 2) Aplicación web

- a. Preparación del entorno de trabajo (instalar servidor web y php)
- b. Creación de la estructura básica de navegación (página index.html, hoja de estilos css)
- c. Conexión al BD (creación de una clase para comunicar con la BD)

- d. Creación de vistas (vista listado de proyectos y vista detalle de un proyecto)
- 3) Presentación
  - a. Presentación del proyecto en PowerPoint (diapositivas)
  - b. Video demostrativo del proyecto (diapositivas y uso)
- 4) Documentación
  - a. Plan de Trabajo
  - b. Documentación PAC2
  - c. Documentación PAC3
  - d. Documentación final del TFC

A continuación se presenta una visualización inicial donde pueden observarse las dependencias de las tareas en base a una primera duración estimada que será revisada posteriormente.



(fig.1 Dependencias de las EDT)

	⊞	Name	Duration	...	...	Predecessors
1	⊞	<b>Base de Datos Relacional</b>	25 days	...	...	19
2		Identificación de las Clases	10 days	...	...	
3		Preparación del entorno de trabajo	1 day	...	...	
4		Desarrollo de la estructura	10 days	...	...	2;3
5		Introducción de Datos de ejemplo	5 days	...	...	4
7	⊞	<b>Aplicación Web</b>	51 days	...	...	19
8		Preparación del entorno de trabajo	1 day	...	...	
9		Estructura básica	10 days	...	...	4;8
10		Conexión al BD	1 day	...	...	4;9
11		Creación de vistas	20 days	...	...	10
13	⊞	<b>Presentación</b>	3 days	...	...	1;7
14		PowerPoint	2 days	...	...	
15		Video demostrativo	3 days	...	...	
17	⊞	<b>Documentación</b>	142 days?	...	...	
18		Plan de Trabajo	15 days?	...	...	
19		PAC2	25 days	...	...	18
20		PAC3	21 days	...	...	19
21		Documentación final TFC	81 days?	...	...	20

(fig.2 Duración de las EDT)

## FASE DE PLANIFICACIÓN DEL PROYECTO

### PLAN DE GESTIÓN DEL PROYECTO

#### Objetivos y Resultados

- Desarrollar una Base de Datos capaz de almacenar proyectos. El resultado deseado será una estructura documentada y reutilizable en un gestor de base de datos.
- Generar datos de ejemplo para la Base de Datos. El resultado será un archivo importable con dichos datos que pueda utilizarse conjuntamente con la estructura previamente desarrollada.
- Desarrollar una interfaz web que permita visualizar los datos de un proyecto previamente introducido. El resultado deseado será una serie de formularios web que permitan visualizar y comparar los datos de ejemplo introducidos en la base de datos.
- Realizar una presentación audiovisual del funcionamiento. El resultado esperado será unas diapositivas y un video que ilustre la totalidad de la funcionalidad del producto.
- Finalizar la documentación del proyecto. El resultado deberá cubrir tanto el proceso como el producto realizado en un único documento de texto.

#### Hitos

A los hitos previamente identificados durante la fase de iniciación del proyecto, se añaden los hitos intermedios correspondientes a la ejecución del propio proyecto:

- PAC 1 entregada
- PAC 2 entregada
- PAC 3 entregada

- Entorno de trabajo preparado
- Base de Datos desarrollada
- Datos de ejemplo introducidos
- Interfaz Web de ejemplo desarrollada
- Presentación audiovisual elaborada
- Documentación final entregada

### Actividades

Las tareas en las que se descompondrá el proyecto, incluyendo el hito a alcanzar y el entregable producido, son:

- **Redactar la PAC 1** (Hito alcanzado: PAC 1 entregada; entregables: PAC1.pdf)
- **Redactar la PAC 2** (Hito alcanzado: PAC 2 entregada; entregables: PAC2.pdf)
- **Redactar la PAC 3** (Hito alcanzado: PAC 3 entregada; entregables: PAC3.pdf)
- **Preparar el entorno de trabajo** (Hito alcanzado: Entorno de trabajo preparado; entregables: ninguno)
- **Desarrollar la Base de Datos** (Hito alcanzado: Base de datos desarrollada; entregables: BD.sql)
- **Introducir los Datos de Ejemplo** (Hito alcanzado: Datos de ejemplo introducidos; entregables: Ejemplos.sql)
- **Desarrollar la Interfaz Web** (Hito alcanzado: Interfaz web de ejemplo desarrollada; entregables: Web.zip)
- **Elaborar la Presentación audiovisual** (Hito alcanzado: Presentación audiovisual elaborada; entregables: Presentacion.ppt, Presentacion.avi)

- **Redactar la Documentación Final** (Documentación final entregada; entregables: Presentacion.avi)

## Calendario

El proyecto contará en total con un total de 16 semanas, siendo inamovibles tanto el inicio (26/2/2015) como el final (18/6/2015).

Existen otras tres fechas requeridas que coinciden con los tres hitos de entrega de PAC:

1. **Plan de Trabajo** (18/3/2015)
2. **PAC2** (22/4/2015)
3. **PAC3** (21/5/2015)

## Riesgos

Se preverán los siguientes riesgos:

- Requisitos excesivos para los productos
- No se cumplen los calendarios establecidos en el proyecto
- Errores en los productos
- Cambios de requisitos del proyecto

Además, puede presentarse una oportunidad:

- Duración menor de las tareas respecto a aquellas planificadas

## Calidad

Se seguirá de modo orientativo los criterios generales establecidos en la norma ISO/IEC 9126 para la calidad del producto en Ingeniería del Software.

## PLAN DE GESTIÓN DEL ALCANCE

### Recogida de Requisitos

Necesidades que cubrirán los objetivos del Proyecto:

- **REQ-01:** La Base de Datos debe almacenar los datos de un proyecto  
Fases, actividades, recursos, personas... En general todos los datos pertinentes para un proyecto real.
- **REQ-02:** Deberán introducirse datos de ejemplo  
Se introducirán todos los datos de este proyecto y datos de ejemplo de dos proyectos más para permitir demostrar la función de proyectos relacionados.
- **REQ-03:** La interfaz web de ejemplo debe poder mostrar todos los datos de un proyecto  
Se mostrará en HTML toda la información recopilada para el requisito REQ-01 y sugerencias de proyectos relacionados en cada categoría.
- **REQ-04:** La interfaz web debe poder comparar dos proyectos  
Para cada proyecto la interfaz web ofrecerá sugerencias de otros proyectos que podrán consultarse en la misma u otra página (utilizando las funciones apropiadas del navegador web).
- **REQ-05:** La presentación debe mostrar toda la funcionalidad de la interfaz web y la BD  
Durante la presentación se hará una demostración del funcionamiento de la web.

### Definición Detallada del Alcance

#### *Entregables del Proyecto*

Los Entregables a producir durante el proyecto serán 9:

- **ENT-01 PAC1.pdf:** Documentación asociada a la Fases de Iniciación del Proyecto

- **ENT-02 PAC2.pdf:** Documentación asociada a la Fase de Planificación del Proyecto
- **ENT-03 PAC3.pdf:** Documentación asociada a la Fase de Ejecución, Seguimiento y Control del Proyecto
- **ENT-04 TFC.pdf:** Documentación final del Proyecto
- **ENT-05 BD.sql:** Archivo SQL con la estructura de la Base de Datos
- **ENT-06 Ejemplos.sql:** Archivo SQL con los datos de ejemplo
- **ENT-07 Web.zip:** Archivo con el interfaz web de ejemplo
- **ENT-08 Presentacion.ppt:** Diapositivas de presentación del proyecto
- **ENT-09 Presentacion.avi:** Video de presentación del proyecto

### *Limitaciones*

Tendremos en consideración 4 Limitaciones:

- **LIM-01:** el entregable ENT-01 deberá estar listo antes del 18/3/2015
- **LIM-02:** el entregable ENT-02 deberá estar listo antes del 22/4/2015
- **LIM-03:** el entregable ENT-03 deberá estar listo antes del 21/5/2015
- **LIM-04:** los entregables ENT-04 a ENT-09 deberán estar listos antes del 18/6/2015

### *Hitos*

Se establecerán 11 hitos:

- **HIT-01:** PAC1 entregada
- **HIT-02:** PAC2 entregada
- **HIT-03:** Entorno de desarrollo listo

- **HIT-04:** Análisis previo terminado
- **HIT-05:** Base de Datos Operativa
- **HIT-06:** Datos de ejemplo introducidos
- **HIT-07:** Web de ejemplo Operativa
- **HIT-08:** PAC3 entregada
- **HIT-09:** Presentación lista
- **HIT-10:** Documentación final lista
- **HIT-11:** TFC entregado

### Creación de la EDT

Analizando las EDT que se han establecido en el Plan de Gestión del Proyecto, procedemos a subdividir las en componentes menores:

- **EDT Base de Datos**
  - Análisis
  - Diseño
  - Preparación Gestor de Base de Datos
  - Programación
  - Introducción de Datos de Prueba
  - Test
  - Documentación
- **EDT Web**
  - Diseño
  - Preparación entorno de Desarrollo Web

- Programación
  - Conexión al DB
  - Creación de vistas y consultas
- Test
- Documentación
- **EDT Documentación**
  - Elaboración PAC1 (Plan de Trabajo)
  - Elaboración PAC2
  - Elaboración PAC3
  - Elaboración de la Documentación Final del TFC
- **EDT Presentación**
  - Elaboración de la presentación
  - Grabación del video

## PLAN DE GESTIÓN DEL TIEMPO

### Definición de las actividades

Se definirán 9 actividades principales:

- **ACT-01:** Redacción PAC 1  
Consiste en la redacción del plan de trabajo para este proyecto. En mi caso coincide con la parte Previa y de Iniciación del Proyecto. Deberá ser aprobado por el consultor antes de continuar.
- **ACT-02:** Redacción PAC 2  
Consiste en la redacción de la parte de Planificación del Proyecto para su valoración por parte del consultor dentro del marco de la EC.
- **ACT-03:** Redacción PAC 3  
Consiste en la redacción de la parte de Ejecución y Control para su valoración por parte del consultor dentro del marco de la EC.
- **ACT-04:** Preparación del entorno de trabajo  
Instalación de los elementos necesarios para trabajar. En mi caso son el gestor de base de datos (PostgreSQL), un servidor web (IIS) y el módulo CGI de php.
- **ACT-05:** Elaboración de la Base de Datos  
Analizar, diseñar y programar la estructura relacional necesaria para almacenar un proyecto.
- **ACT-06:** Generación Datos de Ejemplo  
Introducir en la Base de Datos los datos de este proyecto y otros datos de ejemplo.
- **ACT-07:** Elaboración Interfaz Web  
Programar con php un pequeño interfaz web capaz de conectar con la Base de Datos y ejecutar consultas para recuperar la información almacenada
- **ACT-08:** Elaboración de la Presentación audiovisual

Realizar un pase de diapositivas y grabar un video defendiendo mi proyecto para el tribunal

- **ACT-09:** Redacción Documentación Final

Redacción de una memoria final del TFC

### Secuenciación de las actividades

Comparando las actividades entre ellas (técnica de análisis de red) se pueden observar claras dependencias que establecerán el camino crítica de la secuencia de realización:

- La iniciación debe preceder a la planificación, por lo que ACT-01 Debe preceder a ACT-02
- La planificación debe preceder a la ejecución, por lo que ACT-02 debe preceder a ACT-04
- El entorno se preparará antes de iniciar el desarrollo, por lo que ACT-04 debe preceder a ACT-05
- Cuando la estructura esté lista, se generarán los datos, por lo que ACT-05 precederá a ACT-06
- Una vez haya datos para visualizar se elaborará el interfaz para visualizarlos, por lo que ACT-06 precederá a ACT-07
- Cuando el interfaz gráfico sea funcional, se procederá a documentar, por lo que ACT-07 precederá a ACT-03, ACT-08 y ACT-09

### Estimación de la duración de las actividades

Teniendo en consideración el esfuerzo estimado necesario para cada tarea, podemos planificar:

- **ACT-01:** 5 días
- **ACT-02:** 5 días
- **ACT-03:** 5 días
- **ACT-04:** 2 días

- **ACT-05:** 8 días
- **ACT-06:** 2 días
- **ACT-07:** 14 días
- **ACT-08:** 4 días
- **ACT-09:** 10 días

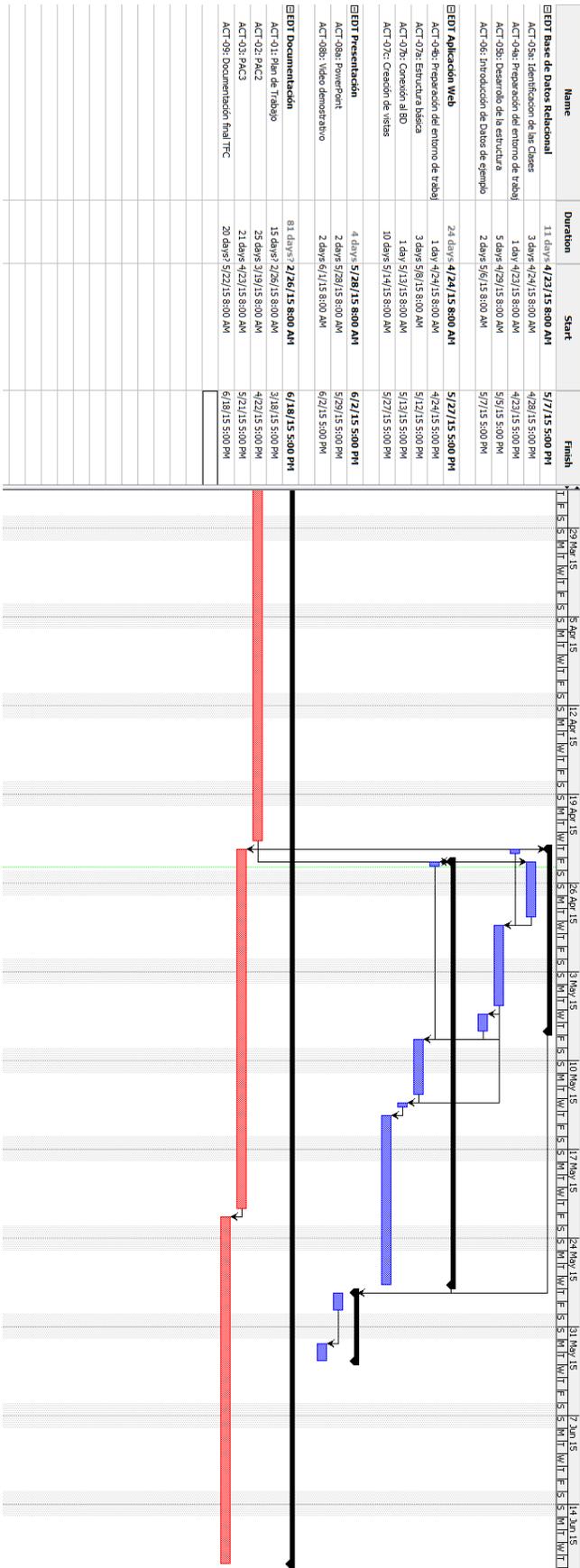
### Elaboración del calendario

Partiendo de la duración estimada, se puede elaborar un primer calendario. Nótese que en estas estimaciones no se han considerado los fines de semana, por lo que las tareas más largas contarán con dos días de buffer que permitirán ajustar mejor la carga de trabajo durante la semana.

- **ACT-01:** 5 días
- **HIT-01:** 18/3/2015
- **ACT-02:** 5 días
- **HIT-02:** 22/4/2015
- **ACT-04:** 2 días
- **HIT-03:** estimado 24/4/2015
- **ACT-05:** 8 días
- **HIT-04:** estimado 28/4/2015
- **HIT-05:** estimado 5/5/2015
- **ACT-06:** 2 días
- **HIT-06:** estimado 7/5/2015

- **ACT-07:** 14 días
- **HIT-07:** estimado 27/5/2015
- **ACT-03:** 5 días
- **HIT-08:** 21/5/2015
- **ACT-08:** 4 días
- **HIT-09:** estimado 2/6/2015
- **ACT-09:** 10 días
- **HIT-10:** estimado 12/6/2015
- **HIT-11:** 18/6/2015

Finalmente, utilizando estos datos se puede generar un gráfico Gantt donde representar visualmente el Plan de Gestión del tiempo.



(fig.3 Calendario)

## PLAN DE CALIDAD

### Criterios de Calidad

Debido a la dificultad de establecer unos criterios de calidad estandarizados en los Proyectos TIC, y aunque se utilizará de forma orientativa la norma ISO/IEC 9126, el principal criterio de calidad deberá ser forzosamente la Calidad Percibida por el Consultor tanto del producto, cómo de la documentación asociada.

### Declaración de Calidad

#### *Misión y Alcance*

La intención de este plan de calidad es realizar una declaración de intenciones, estableciendo los aspectos deseables que se perseguirán durante el desarrollo del producto, pero sin establecer métricas externas para su evaluación, quedando está restringida al campo subjetivo de la Calidad Percibida.

#### *Objetivos*

Siguiendo la ISO 9126, los objetivos de calidad que se perseguirán en este proyecto son:

- **Funcionalidad:** El producto deberá cumplir con toda la especificación funcional establecida durante el proyecto. Es decir, deberá hacer lo que se espera de él.
- **Fiabilidad:** Se garantizará mediante una batería de pruebas la estabilidad del producto generado. No se recortarán los tiempos de pruebas ni siquiera en caso de necesidad.
- **Usabilidad:** Tanto el código generado, como la estructura implementada deberá ser clara y comprensible para el usuario final.

- **Eficiencia:** Se buscará evitar la repetición de información en la Base de Datos aprovechando al máximo el aspecto relacional de la Base de Datos. La ausencia de repeticiones repercutirá en un menor tamaño de los proyectos, hablándose por tanto de una mayor eficiencia espacial.
- **Mantenibilidad:** Por la naturaleza del proyecto, el aspecto de la mantenibilidad que será tratado con mayor profundidad será la testeabilidad del código.
- **Portabilidad:** Debido a la separación estructural entre la Base de Datos y el interfaz web de prueba, la estructura subyacente podrá ser utilizada por otras aplicaciones. Igualmente, al tratarse de una Base de Datos relacional, deberá ser fácilmente escalable.

### *Enfoque*

Para asegurar la calidad de los productos, se establecerán los pertinentes procesos de seguimiento y control de la calidad (testing), que contarán con sus propias tareas en las EDT de desarrollo y que deberán probar todos los requisitos funcionales establecidos.

En caso de no satisfacerse estos, se activarán los procesos de no conformidad y corrección a los que seguirá una nueva fase de test. En ningún caso se recortará tiempo de pruebas.

## PLAN DE GESTIÓN DE RIESGOS

### Clasificación de los Riesgos

#### *Categorías de riesgos*

Se contemplarán 4 categorías de riesgos. Cada riesgo podrá pertenecer a una o más categorías

- CAT-01: Riesgos producidos por la propia planificación
- CAT-02: Riesgos producidos por factores externos
- CAT-03: Riesgos que afectan al esfuerzo
- CAT-04: Riesgos que afectan a la duración

#### *Tipo de tolerancias admitidas*

Los riesgos de CAT-04 no serán aceptables durante la fase de cierre, ya que producirán el fracaso total del proyecto. Los que se produzcan en otras fases, si bien podrán en peligro el proyecto, deberán ser mitigados ajustando la planificación acordemente.

En general se intentará evitar los riesgos de CAT-01, reduciéndose los riesgos posibles a CAT-02.

#### *Métodos de seguimiento y control*

Todos los riesgos identificados serán monitorizados durante la fase de ejecución del proyecto. Si nuevos riesgos u oportunidades son detectados, serán incorporadas mediante los pertinentes informes de riesgos.

## Identificar los Riesgos

Se preverán 4 riesgos y una oportunidad:

- **RSG-01 Requisitos Excesivos:** Puede producir una excesiva carga de trabajo en las EDT de desarrollo que puede llevar a que no se cumplan los calendarios de entrega (ver RSG-02).  
Categorías: CAT-01, CAT-03
- **RSG-02 No se cumplen los calendarios:** Si se produce durante la fase de ejecución, podrá responderse correctamente. Si se produce en la entrega final significará el fracaso total del proyecto. Categorías: CAT-02, CAT-04
- **RSG-03 Errores en los productos:** Pueden influir negativamente sobre la calidad percibida y aumentar los tiempos de test en las EDT de desarrollo. Categorías: CAT-02, CAT-03
- **RSG-04 Cambios de requisitos:** Puede provocar a su vez RSG-01 y RSG-02, si se producen durante las EDT de desarrollo o documentación. Categorías: CAT-02, CAT-03
- **OPR-01 Menor duración de las tareas:** Esta oportunidad es el equivalente positivo al riesgo RSG-02, en el que el tiempo programado será excesivo para la realización de la tarea.  
Categorías: CAT-02, CAT-04

## Análisis Cualitativo de los Riesgos

- **RSG-01:** Probabilidad Media, Impacto Alto
- **RSG-02:** Probabilidad Alta, Impacto Alto
- **RSG-03:** Probabilidad Baja, Impacto Medio
- **RSG-04:** Probabilidad Baja, Impacto Bajo
- **OPR-01:** Probabilidad Baja, Impacto Alto

## Planificación de la Respuesta

- **RSG-01:** Se intentará evitar durante las EDT de análisis. Si se descubre durante las EDT de desarrollo, se intentarán mitigar utilizando un sistema de control de cambios.
- **RSG-02:** Si conduce a no realizar la entrega final, se evitará incurriendo en RSG-04 para intentar producir OPR-01, si fuese necesario. En general se evitará cumpliendo y ajustando el calendario durante la Fase de ejecución del proyecto.
- **RSG-03:** Se evitará mediante las EDT de test. A su vez, las EDT de test no podrán ser reducidas.
- **RSG-04:** Se evitará estableciendo un sistema de control de cambios que impida que se produzcan a su vez RSG-01 y RSG-02. Si se aceptan cambios, estos deberán favorecer que se produzca OPR-01.
- **OPR-01:** Se intentará mejorar el calendario para intentar que se produzca. Si no es posible ajustar el calendario voluntariamente, se aceptará si se produce.

## FASE DE EJECUCIÓN DEL PROYECTO

### INFORME DE LANZAMIENTO

En el momento del lanzamiento (kickstart) del proyecto, no ha sido necesario ni revisar el alcance ni la planificación de este. Igualmente, durante el seguimiento del proyecto en la fase de ejecución todos los objetivos se han mantenido iguales (y de hecho se ha cumplido con los objetivos), siendo la única nota discordante el hecho de que los datos de ejemplo de los proyectos adicionales son quizás demasiado breves.

Posteriormente a la realización del informe de lanzamiento, debido a una incidencia ha sido necesario revisar la planificación del proyecto, si bien no ha sido necesario modificar el alcance de este. Para mayor detalle, consultar las secciones de gestión de incidencias, gestión de cambios y gestión de calendario un poco más adelante.

## GESTIÓN DE INCIDENCIAS

Durante esta fase se ha producido una sola incidencia, como ya se ha visto en la revisión de la planificación del proyecto. Esta incidencia ha producido un retraso ya que el desarrollo de la aplicación no ha partido cuando era planificado y no se ha generado el entregable ENT-03, es decir el documento para la PAC3.

- **INC-01: Retraso**

Se ha producido una parada total en el desarrollo del TFC entre el 22/04/2015 y el 05/05/2015.

Tal como se había previsto en el riesgo RSG-02, se ha debido realizar un cambio (tal como predecía el RSG-04) para salvar el proyecto (véase CMB-01). Por suerte, esta incidencia sólo provocará cambios en el calendario y no será necesario revisar el alcance del proyecto.

## GESTIÓN DE CAMBIOS

Debido a la incidencia anteriormente tratada, debemos generar una serie de cambios en el proyecto. Una vez gestionadas, se han realizado los cambios apropiados en los entregables.

- **CMB-01:** Cancelación del entregable ENT-03

Debido al salto en la planificación, el ENT-03 (redacción de la PAC3) no ha podido realizarse dentro de los tiempos establecidos. Para no comprometer el resto del proyecto, el contenido del ENT-03 será incorporado directamente en el entregable ENT-04 (Documentación final). Con este cambio, también queda anulada la actividad ACT-03, de redacción de la PAC3 debiendo absorberse la carga de trabajo dentro de la actividad ACT-09 (redacción de la memoria final). El Hito HIT-08 tampoco será alcanzable, por lo que se eliminará de la programación. Este cambio se ha aprobado con fecha 05/05/2015.

- **CMB-02:** Modificación del entregable ENT-05 y cancelación del entregable ENT-06

Para agilizar la entrega del TFC, reduciendo el número de archivos a entregar y evitar posibles confusiones, se generará un solo archivo SQL con la estructura y los datos del proyecto, por lo que el entregable ENT-06 (Ejemplos.sql) viene cancelado y su contenido incorporado en el ENT-05 (BD.sql) y no varía ninguna actividad. Este cambio se ha aprobado con fecha 17/06/2015.

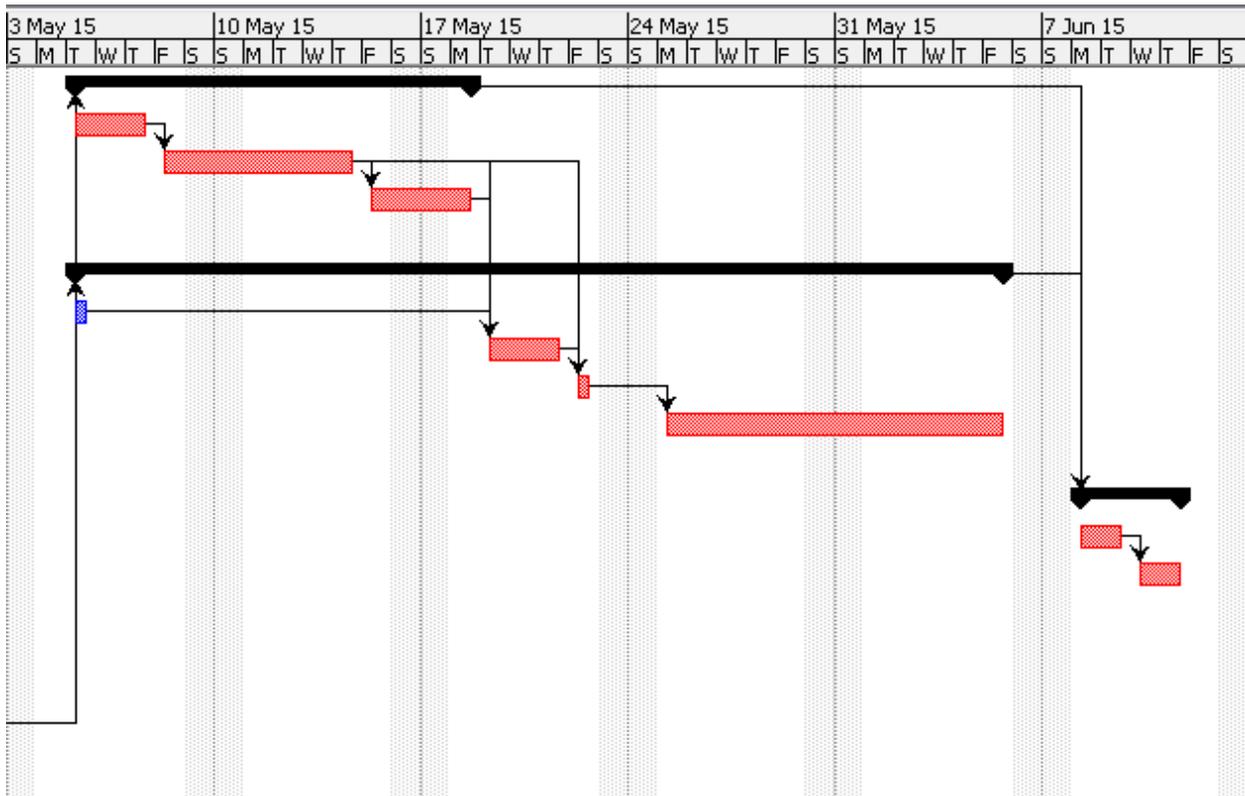
## GESTION DEL CALENDARIO

Partiendo del calendario original una vez tratadas las incidencias y los cambios vemos que debemos ajustar las fechas tanto de los Hitos como de las actividades

- **ACT-01:** 5 días
- **HIT-01:** 18/3/2015
- **ACT-02:** 5 días
- **HIT-02:** 22/4/2015
- **ACT-04:** 2 días
- **HIT-03:** 5/5/2015
- **ACT-05:** 8 días
- **HIT-04:** 13/5/2015
- **HIT-05:** 13/5/2015
- **ACT-06:** 2 días
- **HIT-06:** 15/5/2015
- **ACT-07:** 14 días
- **HIT-07:** 30/5/2015
- **ACT-08:** 4 días
- **HIT-09:** 5/6/2015
- **ACT-09:** 10 días
- **HIT-10:** 15/6/2015
- **HIT-11:** 18/6/2015

Como podemos ver, aun se puede terminar el proyecto dentro de plazo, pero como se ha visto, la carga de trabajo de la ACT-09 será mayor de lo estimado originalmente.

Finalmente, utilizando estos datos se puede generar un gráfico Gantt actualizado donde representar visualmente el Plan de Gestión del tiempo actualizado.



(fig.4 Calendario actualizado)

## FASE DE CIERRE DEL PROYECTO

Llegados a este punto, se puede proceder al cierre del proyecto. Los entregables serán enviados al consultor mediante las herramientas apropiadas del campus virtual y corresponderá al tribunal la aceptación del proyecto.

Como nota para la validación del alcance, a este punto considero que se han cumplido con todos los objetivos establecidos, si bien siempre hay espacio para la mejora como se verá en las conclusiones y la autoevaluación más adelante.

## CICLO DE DESARROLLO DEL PRODUCTO

### FASE DE ANÁLISIS

Durante esta fase he aplicado las técnicas propias de Ingeniería del Software y modelado de sistemas.

#### Análisis de Requisitos

##### *Comprensión del Dominio*

¿Qué es un proyecto? Se puede definir un proyecto como un esfuerzo temporal diseñado para generar un único producto o servicio.

Así, la gestión de proyectos es el proceso y actividad de planear, organizar y controlar recursos para alcanzar los objetivos.

La primera parte de este trabajo la he dedicado a identificar y comprender cuales son los elementos que componen un proyecto. El modelo de trabajo que he adoptado en este TFC presupone que cualquier proyecto se compone de las siguientes partes:

- Datos generales del proyecto: tales como su nombre, descripción, portafolio si pertenece a uno.
- Fases: las fases por las que se va pasando el proyecto.
- Hitos: los hitos que se han programado para el proyecto.
- Actividades: las actividades a realizar dentro del proyecto.
- Entregables: los productos a generar durante el proyecto.

- Riesgos: los riesgos identificados para el proyecto.
- Cambios: los cambios propuestos para el proyecto
- Incidencias: issues ocurridos durante el desarrollo del proyecto.
- Recursos: los recursos asignados al proyecto.
- Personas: personas asociadas al proyecto, tales como PM, stakeholders, trabajadores, etc...
- Documentos: otros documentos asociados al proyecto

### *Recolección y Definición de Requisitos*

#### Requisitos funcionales

- **RF-01: Recuperar listado de proyectos.** El interfaz web de prueba debe ser capaz de recuperar un listado de todos los proyectos almacenados en la Base de Datos.
- **RF-02: Mostrar detalle del proyecto.** El interfaz web de prueba debe ser capaz de mostrar todos los datos de un proyecto.
- **RF-03: Proyectos relacionados.** El interfaz web de prueba debe ser capaz de mostrar proyectos relacionados a un proyecto determinado.
- **RF-04: Almacenamiento de proyectos.** Un proyecto almacenado en la Base de Datos debe contener todas las partes individuadas durante la fase de comprensión del dominio.

#### Requisitos no funcionales del producto

- **RNFP-01:** La Base de Datos debe ser capaz de almacenar un número ilimitado de proyectos.
- **RNFP-02:** Se desea mantener la integridad referencial de la Base de Datos, es decir, un proyecto no podrá contener referencias a personas, recursos, etc. que no se encuentren introducidas en

el sistema. La eliminación de una de estas referencias, pasa primero por la modificación de todos los proyectos que contengan referencias a estas.

- **RNFP-03:** Facilidad de uso para el usuario final.
- **RNFP-04:** Portabilidad del producto.

#### Requisitos no funcionales organizacionales

- **RNFO-01:** La Base de Datos Relacional será implementada en SQL
- **RNFO-02:** La aplicación de prueba será desarrollada para ser visualizada en un navegador web, por lo que se deberá utilizar un lenguaje de servidor para generar el HTML resultante.

#### *Especificación de Requisitos*

**Requisito:** RF-01

**Condiciones:** Ninguna

**Descripción detallada:**

1. El sistema debe poder conectar a la Base de Datos
2. El sistema debe poder ejecutar una Query que devuelva todos los proyectos introducidos en el sistema
3. El sistema debe poder mostrar cada uno de los proyectos para que el usuario pueda elegir uno
4. El sistema debe desconectar de la Base de Datos al terminar

**Requisito:** RF-02

**Condiciones:** Se ha elegido un código de proyecto

**Descripción detallada:**

1. El sistema debe poder conectar a la Base de Datos
2. El sistema debe poder ejecutar una Query que devuelva todos los datos del proyecto
3. El sistema debe poder mostrar todos los datos del proyecto elegido
4. El sistema debe desconectar de la Base de Datos al terminar

**Requisito:** RF-03

**Condiciones:** Se ha elegido un código de proyecto

**Descripción detallada:**

1. El sistema debe poder conectar a la Base de Datos
2. El sistema debe poder ejecutar una Query que devuelva proyectos con los mismos componentes que el proyecto seleccionado
3. El sistema debe poder mostrar cada uno de los proyectos relacionados para que el usuario elija uno
4. El sistema debe desconectar de la Base de Datos al terminar

**Requisito:** RF-04

**Condiciones:** Ninguna

**Descripción detallada:**

La Base de Datos debe poder almacenar los siguientes datos de un proyecto

- Datos generales del proyecto.
- Fases.
- Hitos.
- Actividades.
- Entregables.
- Riesgos.
- Cambios.
- Incidencias.
- Recursos.
- Personas.
- Documentos.

**Requisito:** RNFP-01

**Condiciones:** Ninguna

**Descripción detallada:**

La Base de Datos deberá poder almacenar un número ilimitado de entidades. Se entenderá por “ilimitado” una cantidad determinada por un tipo de dato “Integer”.

**Requisito:** RNFP-02

**Condiciones:** Ninguna

**Descripción detallada:**

Cada tabla deberá tener una clave primaria única para identificar cada elemento. Las entidades externas a la tabla deberán referenciar siempre elementos existentes en el sistema, utilizando claves externas para el control de integridad referencial.

**Requisito:** RNFP-03

**Condiciones:** Ninguna

**Descripción detallada:**

Se seguirán criterios de usabilidad y accesibilidad en el diseño de la interfaz web de prueba. Se evitarán opciones inútiles o confusas para el usuario final.

**Requisito:** RNFP-04

**Condiciones:** Ninguna

**Descripción detallada:**

El producto final deberá ser portable. Para ello, no se utilizarán llamadas a librerías externas, sistema operativo, etc. Se utilizará un lenguaje de programación portable e interpretable, cuyo intérprete sea multiplataforma (php) y un gestor de Base de Datos de código abierto también multiplataforma (postgresql).

**Requisito:** RNFO-01

**Condiciones:** Ninguna

**Descripción detallada:**

Se utilizará SQL (postgresql) para el desarrollo de la Base de Datos.

**Requisito:** RNFO-02

**Condiciones:** Ninguna

**Descripción detallada:**

El código en HTML generado debe ser conforme a los estándares establecidos.

### *Resolución de Conflictos*

No se han encontrado conflictos entre los requerimientos recolectados y analizados.

### *Priorización*

Se dará prioridad a los requisitos funcionales y RNFP-02. Los requisitos no funcionales restantes se pueden considerar implícitos para la estrategia de desarrollo elegida.

### *Verificación de Requisitos*

Al no existir otros Stakeholders que satisfacer con estos requisitos, considero que los requisitos recogidos se corresponden con las necesidades del producto y los requisitos establecidos en el marco del proyecto.

Igualmente, no se encuentran inconsistencias o contradicciones entre estos. Son realistas, fácilmente implementables y serán verificables.

## Modelado del Sistema

### *Descripción Dinámica*

El conjunto de estados del sistema se denomina S. Disponemos de 2 estados:

- S0 = Estado inicial. El sistema deberá mostrar el listado de proyectos introducidos en la Base de Datos
- S1 = Vista Proyecto. El sistema mostrará toda la información relacionada con un proyecto especificado.

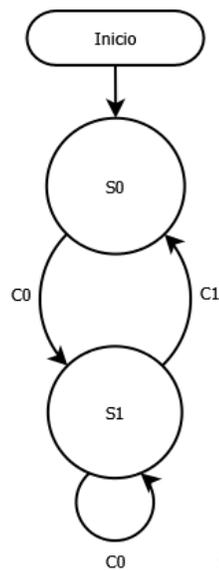
El conjunto de condiciones del sistema se denomina C. Disponemos de dos condiciones:

- C0 = Suministrar código de un proyecto.
- C1 = Utilizar la función de volver atrás para retroceder al último estado anterior.

La función de transición de estados se denomina F, de tal modo que  $F(S_i, C_j) = S_k$ , y cuyo comportamiento viene definido por la siguiente tabla:

Estado Actual	Condición de Entrada	Próximo estado
S0	C0	S1
S0	C1	S0
S1	C0	S1
S1	C1	Estado anterior

Podemos verlo de manera más visual en el Diagrama de Transición de estados de la figura 5.



(fig.5 Diagrama de estados)

## Técnicas de Obtención de Requisitos utilizadas

He iniciado por una investigación de antecedentes, donde he consultado tanto aplicaciones existentes de gestión de proyectos como otros acercamientos al almacenamiento de proyectos en Base de Datos. Para ello he consultado la documentación especificada en la bibliografía así como el repositorio institucional de la UOC. Al no contar con ningún trabajo previo en este campo, toda la información consultada se considera externa al proyecto. Si bien esto me ha preparado para otros enfoques, la perspectiva ha sido bastante limitada y quizás incluso desactualizada, ya que cuando se habla de gestión de proyectos en muchos casos la información resulta excesivamente genérica.

Una vez estudiados los antecedentes, he procedido a la recogida inicial mediante una tormenta de ideas. Partiendo de los antecedentes, he dejado volar mi imaginación y he combinado las ideas previas en los requisitos finales intentando responder siempre a unas preguntas de partida: ¿Qué se considera necesario para almacenar un proyecto? ¿Qué datos se necesitan mostrar al usuario? ¿Cómo se relacionan dos proyectos entre sí?

Posteriormente, he detallado todos los casos de uso y he pasado a una fase de prototipado (programación exploratoria) donde se han visto el resto de necesidades no cubiertas por el prototipo evolutivo que he ido incorporando de manera iterativa al proyecto final.

## Casos de uso

**Caso de Uso:** CU1 - Listado de Proyectos

**Actor principal:** Usuario

**Ámbito:** Interfaz Gráfica de Prueba

**Intereses:** El usuario quiere ver la lista de todos los proyectos introducidos en la Base de Datos

**Precondición:** Ninguna

**Escenario Principal de Éxito:** El usuario abre la página inicial de la interfaz gráfica de prueba. El sistema muestra el listado de proyectos introducidos para que el usuario elija uno.

**Escenarios Alternativos:** Si no hay proyectos el listado estará vacío.

**Caso de Uso:** CU2 – Detalle del Proyecto

**Actor principal:** Usuario

**Ámbito:** Interfaz Gráfica de Prueba

**Intereses:** El usuario quiere ver todos los detalles de un proyecto introducido en la Base de Datos así como los proyectos relacionados en cada categoría del proyecto.

**Precondición:** El usuario ha elegido un proyecto del listado inicial (CU1) o un proyecto relacionado desde otro detalle proyecto (CU2)

**Escenario Principal de Éxito:** El usuario accede a esta página desde el listado inicial o desde los proyectos relacionados de otro proyecto previamente abierto. El sistema muestra todos los datos introducidos en la Base de Datos para el proyecto seleccionado. Debajo de cada categoría se muestran enlaces a los proyectos relacionados con el proyecto actual.

**Escenarios Alternativos:** Si no existen proyectos relacionados, no aparecerán enlaces a otros proyectos.

**Caso de Uso:** CU3 – Buscar Proyectos relacionados

**Actor principal:** Sistema

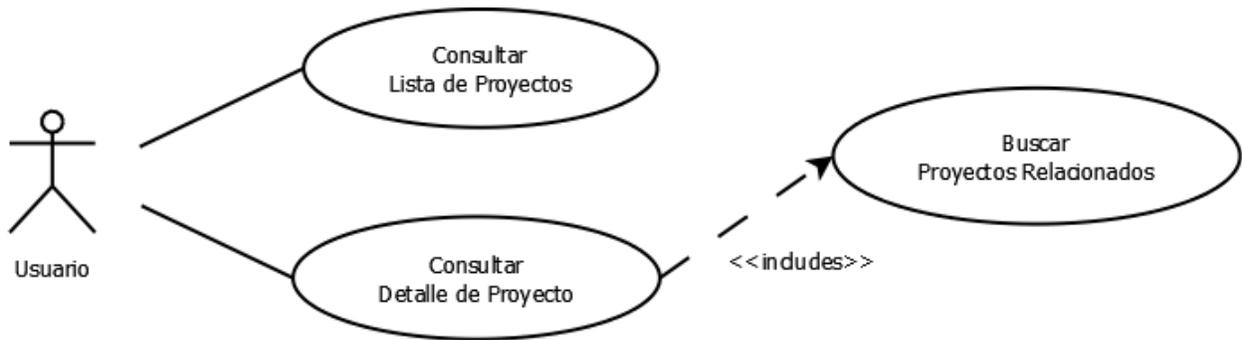
**Ámbito:** Detalle del proyecto

**Intereses:** El sistema mostrará otros proyectos que comparten características con el actual.

**Precondición:** El usuario ha elegido un proyecto del listado inicial (CU1) o un proyecto relacionado desde otro detalle proyecto (CU2)

**Escenario Principal de Éxito:** El sistema consultará la Base de Datos buscando proyectos relacionados para cada elemento del proyecto actual. Si existen proyectos relacionados, se mostrarán enlaces a estos.

**Escenarios Alternativos:** Si no existen proyectos relacionados en ninguna categoría, no aparecerán enlaces a otros proyectos.



(fig.6 Diagrama de casos de uso)

## FASE DE DISEÑO

En esta fase se han traducido los requisitos adquiridos y especificados en la fase anterior en una representación del software que se acerca al producto final.

La idea preliminar era utilizar una aproximación al patrón MVC (*Modelo Vista Controlador*) para la aplicación de prueba, si bien esto se ha realizado en parte, separando el Modelo que gestiona las comunicaciones con la Base de Datos, no ha llegado a diseñarse un controlador totalmente separado de la Vista como hubiese sido deseable en una aplicación real en producción. El motivo principal ha sido que la sencillez de la interfaz gráfica de prueba desaconsejaba la separación de vista y controlador en clases separadas ya que aumentaba sensiblemente la complejidad del diseño.

Finalmente, el patrón utilizado en el diseño de la interfaz gráfica de prueba se aproxima más al patrón proxy, en el que una única instancia de la clase que gestiona la comunicación con la Base de Datos se encarga de realizar de intermediario con la susodicha BD.

### Diseño preliminar

#### *Diseño de datos*

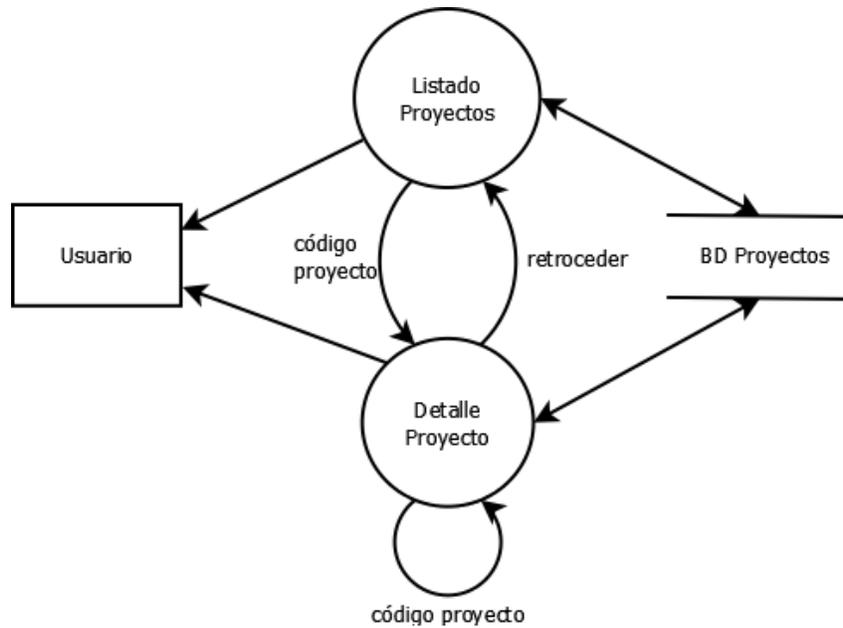
Para satisfacer los requisitos RF-04 y RNFP-02, se utilizarán 23 tablas para almacenar cada proyecto en la Base de Datos:

- **Actividades:** Esta tabla contiene los datos pertinentes de las actividades de los proyectos.
- **Actividad\_Proyecto:** Tabla relacional vinculando actividades con un proyecto concreto.
- **Cambios:** Esta tabla contiene los datos pertinentes a los cambios de los proyectos.

- **Cambio\_Proyecto:** Tabla relacional vinculado cambios a un proyecto concreto.
- **Documentos:** Tabla que contiene los documentos asociados a un proyecto.
- **Entregables:** Tabla que contiene los entregables de los proyectos
- **Entregable\_Proyecto:** Tabla relacional vinculando entregables a un proyecto concreto.
- **Fases:** Esta tabla contiene las distintas fases de los proyectos.
- **Fase\_Proyecto:** Tabla relacional vinculado fases a un proyecto específico.
- **Hitos:** Esta tabla contiene los hitos para utilizar en los proyectos.
- **Hito\_Proyecto:** Tabla relacional vinculando hitos a un proyecto concreto.
- **Incidencias:** Esta tabla contiene las incidencias para los proyectos.
- **Incidencia\_Proyecto:** Tabla relacional vinculando incidencias a un proyecto específico.
- **Personas:** Esta tabla contiene los datos de las personas para utilizar en los proyectos.
- **Persona\_Rol\_Proyecto:** Tabla relacional vinculando personas a un proyecto con un rol específico.
- **Portafolios:** Esta tabla contiene la información de los distintos portafolios en los que se agrupan los proyectos.
- **Proyectos:** Esta tabla contiene la información base de los distintos proyectos.
- **Recursos:** Esta tabla contiene la información de los recursos para utilizar en los proyectos.
- **Recurso\_Proyecto:** Tabla relacional vinculando un recurso a un proyecto concreto.
- **Riesgos:** Esta tabla contiene la información de los riesgos para utilizar en los distintos proyectos.
- **Riesgo\_Proyecto:** Tabla relacional vinculando riesgos con un proyecto específico.
- **Roles:** Esta tabla contiene los distintos roles que pueden tener las personas en un proyecto.
- **TiposRecurso:** Esta tabla contiene los distintos tipos a los que puede pertenecer un recurso.

### Diseño arquitectónico

El usuario interactuará con la Base de Datos utilizando la interfaz gráfica de prueba, que implementará los dos casos de uso definidos en este documento utilizando para ello dos vistas separadas.



(fig.7 Diagrama de flujo de datos)

### Diseño de la interfaz gráfica de usuario

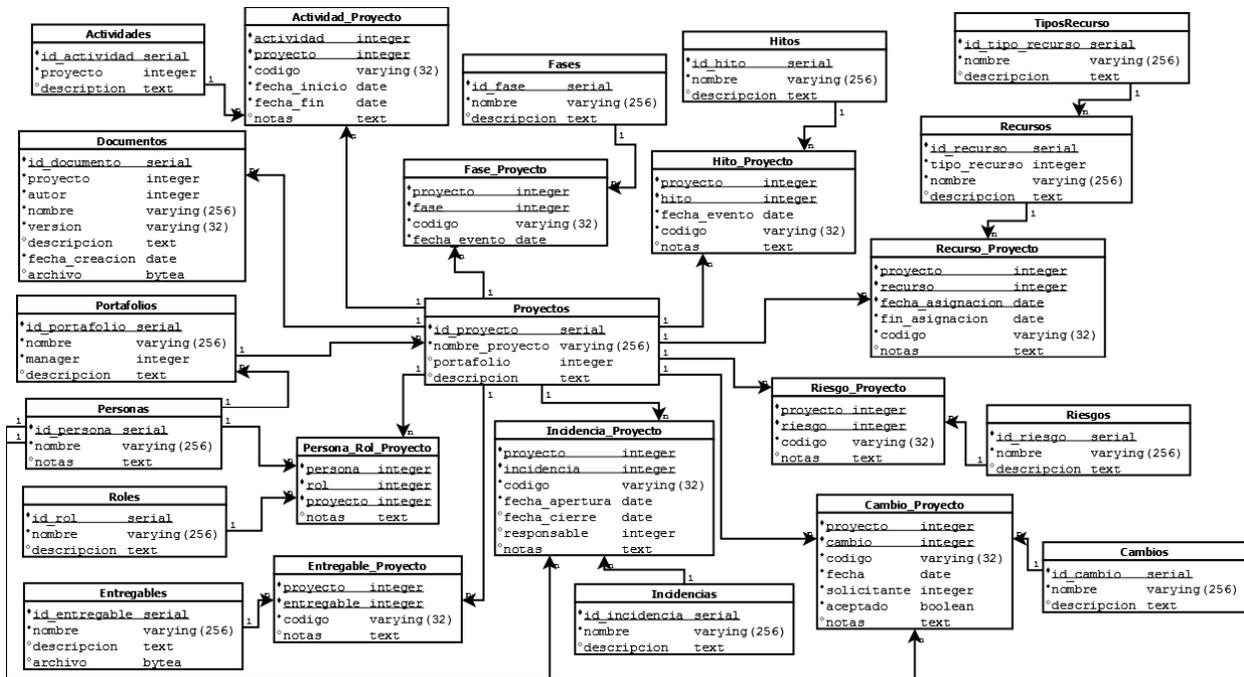
La interfaz se compone de dos vistas separadas:

- **Vista de Listado:** en la que viene mostrado un listado con todos los proyectos almacenados en el sistema.
- **Vista de Proyecto:** vista de detalle en la que vienen mostrado todos los datos de un proyecto específico.

Diseño detallado

Diseño detallado de datos

La Base de Datos se compone de 23 tablas conteniendo la totalidad de la información pertinente a un proyecto:



(fig.8 Estructura de la Base de Datos)

Tablas

Aquí se enumeran las tablas a implementar con sus campos. Las claves primarias están subrayadas, se han creado foreign keys en todos los campos referenciados para mantener la integridad referencial cumpliendo con el requisito RNFP-02.

- "Actividad\_Proyecto"

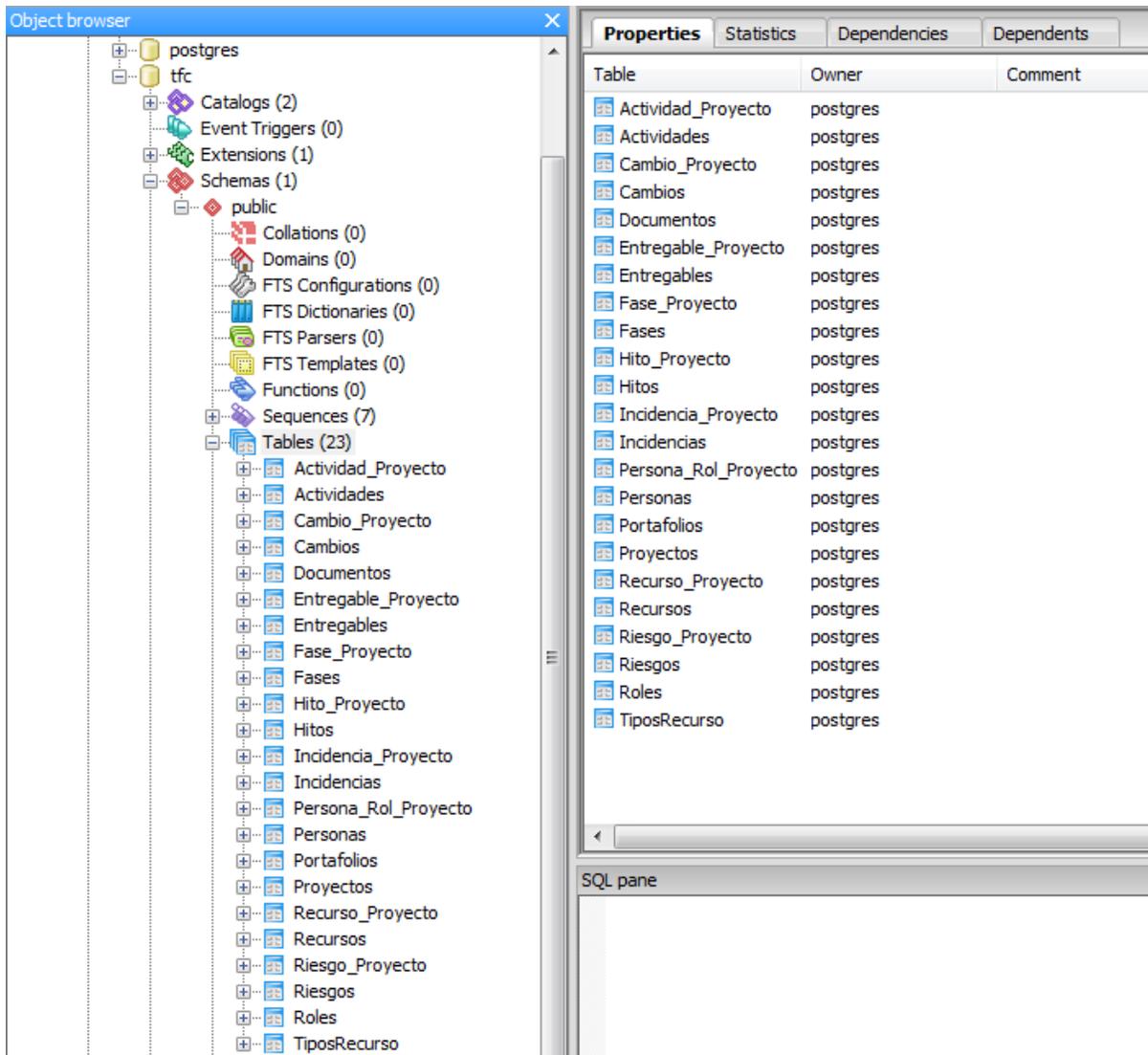
- actividad integer NOT NULL
  - proyecto integer NOT NULL
  - codigo character varying(32) NOT NULL
  - fecha\_inicio date NOT NULL
  - fecha\_fin date NOT NULL
  - notas text
- 
- "Actividades"
    - id\_actividad serial NOT NULL
    - descripcion text
- 
- "Cambio\_Proyecto"
    - cambio integer NOT NULL
    - proyecto integer NOT NULL
    - codigo character varying(32) NOT NULL
    - fecha date NOT NULL
    - solicitante integer NOT NULL
    - aceptado boolean DEFAULT false NOT NULL
    - notas text
- 
- "Cambios"
    - id\_cambio serial NOT NULL,
    - nombre character varying(256) NOT NULL,
    - descripcion text
- 
- "Documentos"
    - id\_documento serial NOT NULL
    - autor integer NOT NULL
    - nombre character varying(256) NOT NULL

- version character varying(32)
- descripcion text
- fecha\_creacion date NOT NULL
- archivo bytea,
- proyecto integer NOT NULL
  
- "Entregable\_Proyecto"
  - proyecto integer NOT NULL
  - entregable integer NOT NULL
  - codigo character varying(32) NOT NULL
  - notas text
  
- "Entregables"
  - id\_entregable serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text
  - archivo bytea
  
- "Fase\_Proyecto"
  - id\_proyecto integer NOT NULL
  - id\_fase integer NOT NULL
  - fecha\_evento date NOT NULL
  - notas text
  - codigo character varying(32) NOT NULL
  
- "Fases"
  - id\_fase serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text

- "Hito\_Proyecto"
  - id\_hito integer NOT NULL
  - id\_proyecto integer NOT NULL
  - fecha date NOT NULL
  - codigo character varying(256) NOT NULL
  - notas text
  
- "Hitos"
  - id\_hito serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text
  
- "Incidencia\_Proyecto"
  - incidencia integer NOT NULL
  - proyecto integer NOT NULL
  - codigo character varying(32) NOT NULL
  - fecha\_apertura date NOT NULL
  - fecha\_cierre date
  - responsable integer
  - notas text
  
- "Incidencias"
  - id\_incidencia serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text
  
- "Persona\_Rol\_Proyecto"

- id\_persona integer NOT NULL
  - id\_rol integer NOT NULL
  - id\_proyecto integer NOT NULL
  - notas text
- 
- "Personas"
    - id\_persona serial NOT NULL
    - nombre character varying(256)
    - notas text
- 
- "Portafolios"
    - id\_portafolio serial NOT NULL
    - nombre character varying(256) NOT NULL
    - manager integer NOT NULL
    - descripcion text
- 
- "Proyectos"
    - id\_proyecto serial NOT NULL
    - nombre\_proyecto character varying(256) NOT NULL
    - portafolio integer
    - descripcion text
- 
- "Recurso\_Proyecto"
    - recurso integer NOT NULL
    - proyecto integer NOT NULL
    - fecha\_asignacion date NOT NULL
    - fin\_asignacion date NOT NULL
    - codigo character varying(32) NOT NULL
    - notas text

- "Recursos"
  - id\_recurso serial NOT NULL
  - tipo\_recurso integer NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text
  
- "Riesgo\_Proyecto"
  - riesgo integer NOT NULL
  - proyecto integer NOT NULL
  - codigo character varying(32) NOT NULL
  - notas text
  
- "Riesgos"
  - id\_riesgo serial NOT NULL
  - nombre character varying(256)
  - descripcion text
  
- "Roles"
  - id\_rol serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text
  
- "TiposRecurso"
  - id\_tipo\_recurso serial NOT NULL
  - nombre character varying(256) NOT NULL
  - descripcion text



(fig.9 Base de datos vista desde pgAdmin III)

### Consultas SQL

Aquí se enumeran las principales consultas SQL realizadas por la interfaz web para recuperar la información de la Base de Datos:

Listar todos los proyectos:

- `SELECT id_proyecto, nombre_proyecto FROM "Proyectos"`

Datos de un portafolio con id = X:

- `SELECT pt.nombre, pr.nombre FROM "Portafolios" pt, "Personas" pr WHERE pr.id_persona=pt.manager AND pt.id_portafolio = X`

Datos de un proyecto con id = X:

- `SELECT nombre_proyecto, portafolio, descripcion FROM "\"Proyectos\" WHERE id_proyecto = X`
- `SELECT f.nombre, f.descripcion, fp.codigo, fp.fecha_evento, fp.notas FROM "Fase_Proyecto" fp, "Fases" f WHERE f.id_fase = fp.id_fase AND fp.id_proyecto = X`
- `SELECT h.nombre, h.descripcion, hp.codigo, hp.fecha, hp.notas, h.id_hito FROM "Hito_Proyecto" hp, "Hitos" h WHERE h.id_hito = hp.id_hito AND hp.id_proyecto = X`
- `SELECT a.nombre, a.descripcion, ap.codigo, ap.fecha_inicio, ap.fecha_fin, ap.notas, ap.actividad FROM "Actividad_Proyecto" ap, "Actividades" a WHERE a.id_actividad = ap.actividad AND ap.proyecto = X`
- `SELECT e.nombre, e.descripcion, ep.codigo, ep.notas, e.archivo, ep.entregable FROM "Entregable_Proyecto" ep, "Entregables" e WHERE e.id_entregable = ep.entregable AND ep.proyecto = X`
- `SELECT d.nombre, d.version, d.descripcion, d.fecha_creacion, p.nombre, d.archivo FROM "Documentos" d, "Personas" p WHERE d.autor = p.id_persona AND d.proyecto = X`
- `SELECT r.nombre, r.descripcion, rp.codigo, rp.notas, rp.riesgo FROM "Riesgo_Proyecto" rp, "Riesgos" r WHERE r.id_riesgo = rp.riesgo AND rp.proyecto = X`
- `SELECT c.nombre, c.descripcion, cp.codigo, cp.fecha, p.nombre, cp.aceptado, cp.notas, cp.cambio FROM "Cambio_Proyecto" cp, "Cambios" c, "Personas" p WHERE c.id_cambio = cp.cambio AND cp.solicitante = p.id_persona AND cp.proyecto = X`

- SELECT i.nombre, i.descripcion, ip.codigo, ip.fecha\_apertura, ip.fecha\_cierre, p.nombre, ip.notas, ip.incidencia FROM "Incidencia\_Proyecto" ip, "Incidencias" i, "Personas" p WHERE i.id\_incidencia = ip.incidencia AND ip.responsable = p.id\_persona AND ip.proyecto = X
- SELECT r.nombre, r.descripcion, tr.nombre, tr.descripcion, rp.codigo, rp.fecha\_asignacion, rp.fin\_asignacion, rp.notas, rp.recurso FROM "Recurso\_Proyecto" rp, "Recursos" r, "TiposRecurso" tr WHERE r.id\_recurso = rp.recurso AND r.tipo\_recurso = tr.id\_tipo\_recurso AND rp.proyecto = X
- SELECT p.nombre, r.nombre, r.descripcion, prp.notas, p.notas, prp.id\_persona FROM "Persona\_Rol\_Proyecto" prp, "Roles" r, "Personas" p WHERE r.id\_rol = prp.id\_rol AND p.id\_persona = prp.id\_persona AND prp.id\_proyecto = X

Para encontrar los proyectos relacionados se utiliza una estructura de consulta en la que se devuelven los proyectos únicos y distintos al proyecto actual en los que se cumple la condición establecida. Por ejemplo, para recuperar los proyectos en los que han participado las mismas personas que en el proyecto X haríamos:

- SELECT DISTINCT p.id\_proyecto, p.nombre\_proyecto FROM "Proyectos" p, "Persona\_Rol\_Proyecto" prp WHERE p.id\_proyecto = prp.id\_proyecto AND p.id\_proyecto != X AND prp.id\_persona IN (lista de personas que han trabajado en el proyecto X)

### *Abstracciones procedimentales*

Para el diseño del interfaz web de prueba se han utilizado las siguientes abstracciones procedimentales que deberán ser traducidas a funciones reales en la fase de codificación:

- Conectar a la BD: Es la función encargada de la conexión con la Base de Datos. Se le debe poder suministrar dirección de la BD, usuario y password.

- Ejecutar Query: Es la función encargada de ejecutar una Query en el DB y procesar su resultado. Recibirá la Query y devolverá el resultado de esta.
- Representar: Es una función encargada de representar en HTML los datos que se le pasen.

### *Implementación de los Casos de Uso*

#### Caso de Uso CA1

Inicio

Generar cabecera de la página HTML

Conectar con la BD

Ejecutar Query para recuperar el listado de Proyectos

Mientras haya Proyectos en la lista

    Representar un enlace con su código y su nombre

Fin Mientras

Desconectar de la BD

Cerrar documento HTML

Fin

#### Caso de Uso CA2

Inicio

Generar cabecera de la página HTML

Conectar con la BD

Ejecutar Query para recuperar todos los datos del proyecto

Representar los datos en pantalla

Cerrar documento HTML

Fin

### Caso de Uso CA3

#### Inicio

Conectar con la BD

Ejecutar Query para recuperar otros proyectos con los mismos elementos que el actual

Mientras haya Proyectos en la lista

Representar un enlace con su código y su nombre

Fin Mientras

Desconectar de la BD

#### Fin

### *Diseño detallado de la interfaz gráfica de usuario*

Se compone de dos vistas, VW-1 que corresponde a CA1 y VW-2 que corresponde a CA2 e incluye CA3. He optado por la simplicidad ya que el objetivo es sólo ejemplificar las posibles aplicaciones de la Base de Datos, por lo que en este paso no se especificarán colores, estilos, etc. quedando estos pospuestos a la fase de codificación.

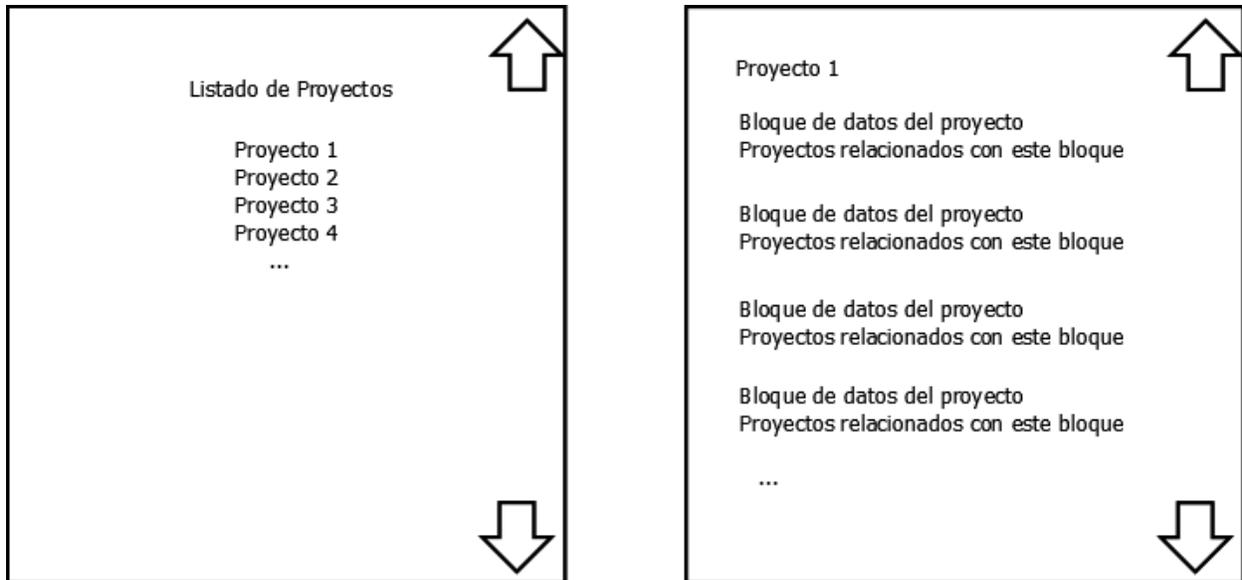
La interfaz web es una página con las dos vistas mencionadas. Una VW-1 mostrando la lista de proyectos y otra VW-2 con los detalles para un proyecto específico. De la vista de detalle se puede pasar a otros proyectos relacionados. Para acceder a ellas, hay que cargar el archivo index.php (incluido en el entregable `tfc_web.zip`) desde un navegador web.

La navegación parte del listado de proyectos introducidos en la Base de Datos. Una vez seleccionado el proyecto deseado la vista cambia a la de detalle (ver figura 5). Allí, los datos del proyecto se dividen en las siguientes secciones:

- Datos generales del proyecto
- Datos del portafolio (sólo si el proyecto pertenece a uno)
- Fases
- Hitos
- Actividades
- Entregables
- Documentos
- Riesgos
- Cambios
- Incidencias
- Recursos
- Personas

Debajo de cada sección pertinente, el sistema sugerirá proyectos relacionados que comparten elementos de dicha sección con el actual. Haciendo clic sobre uno de estos proyectos, pasaremos a la vista de detalle de estos.

Para volver al menú inicial, podemos utilizar la función de volver atrás del navegador o cargar de nuevo la página sin especificar un proyecto como parámetro.



(fig.10 Diseño de Interfaz Gráfica de Usuario)

## Calidad del software

Para garantizar la calidad del producto final, antes de pasar a la fase de codificación se ha realizado una revisión técnica formal (RTF) donde se ha determinado que:

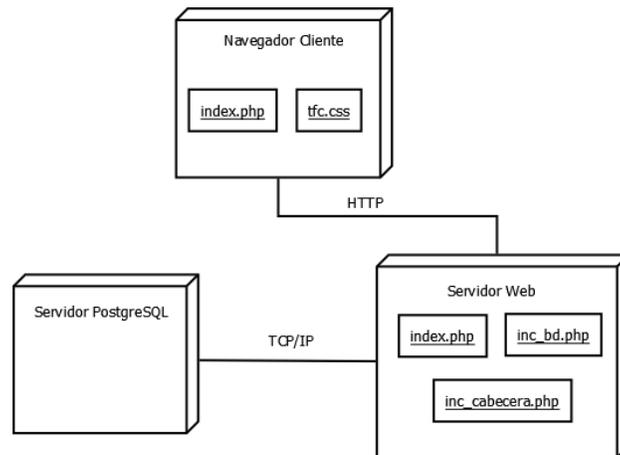
- El diseño propuesto es jerárquico
- El diseño propuesto es modular, con funciones independientes y reutilizables
- El diseño propuesto tiene representaciones distintas de datos y procedimientos
- El diseño propuesto conduce a interfaces que reducen la complejidad de las conexiones con el exterior

## FASE DE CODIFICACIÓN

Durante esta fase se ha producido el código fuente final, interpretable por el ordenador, partiendo de los documentos de diseño desarrollados en la fase anterior.

El producto de este proceso ha sido por un lado la estructura de la Base de Datos en SQL (ver entregable adjunto BD\_tfc.sql) y por otro el código fuente para la página web de prueba (ver entregable tfc\_web.zip).

Podemos ver el despliegue final del proyecto, en tres nodos separados (navegador, servidor y base de datos) una vez codificado en el diagrama de la figura 11. Nótese que los tres nodos pueden encontrarse en sistemas completamente separados, lo que da una idea más de las potencialidades de un sistema real.



(fig.11 Diagrama de despliegue)

Por último, la interfaz web de prueba vista desde el navegador se puede observar en la siguiente imagen una vez aplicada la hoja de estilo `tfc.css` que se ha utilizado para formatear el texto resultante.



(fig.12 Vistas del interfaz web de prueba)

## FASE DE PRUEBAS

Durante esta fase se ha testado el código fuente final, generado en la fase anterior. Las pruebas realizadas en esta fase han sido complementarias a las pruebas informales que se han realizado durante el propio desarrollo y que han ido conduciendo a las correcciones anteriormente señaladas.

### Pruebas de Caja Blanca

#### *Pruebas Unitarias*

Además del control informal que he ido realizando durante el desarrollo de los módulos, al final del desarrollo de cada uno de estos se ha procedido a realizar una pequeña batería de pruebas de caja blanca:

- **Módulo de Conexión a la BD:** Se han probado secuencialmente las funciones de conexión al BD, ejecución de query y desconexión sin encontrarse problemas destacables. (**PRB-01**)

- **Módulo de Visualización:** Se ha generado la página sin conectar a la BD comprobando que el documento final era correctamente formateado con todas las secciones correctamente generadas. **(PRB-02)**

### *Pruebas de Integración*

Una vez terminado el desarrollo de los módulos, se ha procedido a su incorporación al programa final mediante pruebas de integración descendientes. Debido al pequeño tamaño de la interfaz gráfica de ejemplo, una vez probados ambos módulos por separado, ha bastado integrar el módulo de conexión a la BD para realizar estas pruebas. **(PRB-03)**

### *Pruebas de Caja Negra*

#### *Pruebas de Validación y aceptación*

Las más sencillas de las pruebas funcionales realizadas. Después de cargar los datos de prueba en la Base de Datos, se ha comprobado que la interfaz gráfica de prueba mostrase los datos correctos utilizando para ello un entorno de producción y el código fuente final. Tal como pasó en las pruebas de integración, el resultado ha sido satisfactorio. **(PRB-04)**

#### *Pruebas de Regresión*

Al desarrollarse por separado los módulos, y no haberse producido cambios desde la validación funcional de estos, no ha sido necesario realizar pruebas de regresión durante este proyecto.

### *Pruebas de Prestaciones*

Debido al tamaño de la interfaz gráfica de prueba, las únicas pruebas de prestaciones realizadas han sido informales. La página se genera correctamente en tiempos aceptables con una carga normal de trabajo. Debido a las limitaciones del proyecto no se han podido realizar pruebas de escalabilidad en la que el interfaz debiese manejar miles de proyectos.

## FASE DE MANTENIMIENTO

El software producido en la fase de desarrollo deberá ser mantenido. Si bien hasta este momento no se han encontrado errores (y por tanto no se ha producido aun un mantenimiento correctivo), ni se han producido cambios en el entorno que justifiquen un mantenimiento adaptativo, en el capítulo conclusiones, en el apartado de propuesta de mejora y líneas de futuro se podrán encontrar propuestas de mantenimiento perfectivo a realizar en un futuro.

## DOCUMENTACIÓN DEL PRODUCTO FINAL

### INSTALACIÓN

#### Base de Datos

En un Gestor de Base de Datos PostgreSQL previamente configurado, basta importar la estructura y datos del sistema utilizando el archivo “BD\_tfc.sql” y la Base de Datos será disponible para utilizar.

#### Interfaz Web de Prueba

Para que la interfaz web de prueba pueda conectar con la Base de Datos, habrá previamente que modificar la dirección, el usuario y la password en la línea 10 del archivo inc\_bd.php con los datos de conexión correctos.

Acto seguido, en un servidor web previamente configurado y con soporte para php5, disponer los archivos inc\_bd.php, inc\_cabecera.php, index.php y tfc.css de forma que sean accesibles desde un navegador web. Cargar desde el navegador web el archivo index.php.

## CONCLUSIONES

Como conclusión, me gustaría recalcar que a pesar de los problemas encontrados durante la Fase de Ejecución del Proyecto se han cumplido los objetivos.

Obviamente, debido al reducido tiempo a disposición para este proyecto, algunos aspectos no han sido tratados como sería necesario en un producto que fuese a lanzarse al mercado. Especialmente el Interfaz web de prueba sólo nos permite entrever el verdadero potencial de una base de datos como esta.

Igualmente, al no ser un producto en producción, los datos de prueba introducidos son reducidos y no se corresponden con proyectos reales, lo que hubiese permitido dotar de mayor realismo a la aplicación web de prueba.

## PROPUESTA DE MEJORA Y LÍNEAS DE FUTURO

Tal como se ha dicho, la Base de Datos puede ser optimizada. Nuevos campos pueden incorporarse para almacenar aún más información útil para el Project Manager. También puede ser que si esta herramienta entra en producción, con el uso se descubran nuevas necesidades a incorporar o algunas de las tablas ya presentes se demuestren poco útiles.

Igualmente, desarrollar una aplicación real que utilizase esta estructura y que fuese totalmente funcional, ya fuera del marco de este TFC produciría una herramienta que, en mi opinión, sería de obligado uso para todos los profesionales de la Gestión de Proyectos.

Otro aspecto en el que se puede profundizar es en la creación de nuevas consultas de mayor complejidad para encontrar nuevas relaciones entre proyectos que podrían pasar desapercibidas con la aplicación actual. Por ejemplo ¿Y si el problema no es un recurso o una persona, sino una persona utilizando un recurso concreto?

Generar un cuerpo mayor de datos iniciales modelo (tipos de recursos, más roles, etc. ) facilitaría el uso para los usuarios potenciales y ayudaría a estandarizar algo más los proyectos resultantes almacenados.

Por último, mi última propuesta sería liberar el código final resultante como Open Source para permitir la reusabilidad de la que hice gala al inicio de este documento.

## AUTOEVALUACIÓN

Aunque no puedo estar totalmente satisfecho del resultado final de mi proyecto, ya que, siendo una persona perfeccionista, me hubiese gustado trabajarlo mucho más, todo el desarrollo conceptual de la Base de Datos me ha servido para afianzar mis conocimientos de la Gestión de Proyectos e introducirme de lleno en el mundo del Project Management y me ha resultado interesantísimo.

Así, durante la realización de este trabajo he podido adquirir nuevas competencias y practicar aquellas que ya tenía, todo ello dentro de un marco temporal establecido en el que he tenido que planificar y elaborar un proyecto complejo.

También quiero recalcar que considero un proyecto como este no sólo es viable, además puede ser realmente útil para un profesional de la gestión de proyectos. Mi intención sería continuar a trabajarlo en mi tiempo libre y sobretodo utilizarlo para la gestión de mis proyectos.

En resumen, he adquirido una visión más profunda del mundo de la gestión de proyectos trabajando en una herramienta de enorme potencial y he alcanzado los objetivos que me había propuesto.

## RECURSOS UTILIZADOS

- Para el desarrollo de la base de datos, se ha utilizado PostgreSQL 9.4 utilizando el interfaz de gestión pgAdmin III
- Para el interfaz web de ejemplo se ha utilizado PHP 5.3 como módulo CGI, y la página generada cumple con los estándares definidos de HTML y CSS
- El ordenador de desarrollo utilizado disponía de Windows 7 Enterprise con Internet Information Services 7 como servidor web
- El software utilizado ha sido Microsoft Word 2013, Powerpoint 2013 para las diapositivas, Dia para los diagramas y Project Libre para la planificación temporal

## BIBLIOGRAFÍA

- Gestión de Proyectos (material docente de la UOC) – José Ramón Rodríguez y Pere Mariné Jové
- Gestión de proyectos informáticos: métodos, herramientas y casos – José Ramón Rodríguez  
(2005) editorial UOC.
- A Guide to the Project Management Body of Knowledge (Fifth Edition) – PMI (2013)