

UNIVERSIDAD DE ZARAGOZA

TRABAJO DE LA ASIGNATURA

---

## Plan de gestión, análisis, diseño y memoria del proyecto

---

<i>Autor/es:</i>	<i>NIA/s</i>
Jorge Aznar López	721556
Ángel Cañal Muniesa	716205
Abel Chils Trabanco	718997
Yasmina Albero Escudero	721288
Óscar Fraca Ferrández	566416
Alexandru Ioan Oarga Hategan	718123
Jorge Pinilla López	715750

## Declaración de autoría

I, Jorge Aznar López

Ángel Cañal Muniesa

Abel Chils Trabanco

Yasmina Alberro Escudero

Óscar Fraca Ferrández

Alexandru Ioan Oarga Hategan

Jorge Pinilla López, declare that this thesis titled, «Plan de gestión, análisis, diseño y memoria del proyecto» and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

*«Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.»*

Dave Barry

UNIVERSIDAD DE ZARAGOZA

## *Resumen*

Faculty Name  
Escuela de Ingeniería y Arquitectura

### **Plan de gestión, análisis, diseño y memoria del proyecto**

por Jorge Aznar López  
Ángel Cañal Muniesa  
Abel Chils Trabanco  
Yasmina Albero Escudero  
Óscar Fraca Ferrández  
Alexandru Ioan Oarga Hategan  
Jorge Pinilla López

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too...

# Índice general

<b>Declaración de autoría</b>	<b>I</b>
<b>Resumen</b>	<b>III</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Resumen . . . . .	1
1.2. Objetivos . . . . .	1
<b>2. Organización del proyecto</b>	<b>2</b>
2.1. Equipo . . . . .	2
<b>3. Plan de gestión del proyecto</b>	<b>3</b>
3.1. Procesos . . . . .	3
3.1.1. Procesos de inicio de proyecto . . . . .	3
3.1.2. Procesos de ejecución y control del proyecto . . . . .	3
3.1.3. Procesos técnicos . . . . .	3
3.2. Planes . . . . .	4
3.2.1. Plan de gestión de configuraciones . . . . .	4
3.2.2. Plan de construcción y despliegue del software . . . . .	4
3.2.3. Plan de aseguramiento de la calidad . . . . .	4
3.2.4. Calendario del proyecto y división del trabajo . . . . .	4
<b>4. Análisis y diseño del sistema</b>	<b>6</b>
4.0.1. Análisis de requisitos . . . . .	6
4.0.2. Diseño del sistema . . . . .	6
<b>5. Memoria del proyecto</b>	<b>7</b>
5.0.1. Inicio del proyecto . . . . .	7
5.0.2. Ejecución y control del proyecto . . . . .	7
5.0.3. Cierre del proyecto . . . . .	7
<b>6. Conclusiones</b>	<b>8</b>
<b>A. Dedicación</b>	<b>9</b>
A.1. Dedicación . . . . .	9

# Índice de figuras

# Índice de cuadros

A.1. Dedicación a la práctica de los integrantes del grupo. . . . .	9
---	---

*For/Dedicated to/To my...*



## Capítulo 1

# Introducción

### 1.1. Resumen

Resumen del proyecto, propósito, alcance, objetivos, entregables e hitos principales. Alrededor de una página es suficiente.

Incluir una breve descripción de la estructura del resto del documento.

### 1.2. Objetivos

## Capítulo 2

# Organización del proyecto

### 2.1. Equipo

- Equipo del proyecto: integrantes del mismo, roles y responsabilidades. Qué hace dentro del proyecto cada miembro del equipo. Aunque es normal que todo el mundo haga varias cosas, también es importante que haya responsables definidos para las tareas importantes.
  - Es importante determinar un director o directora de proyecto.

## Capítulo 3

# Plan de gestión del proyecto

### 3.1. Procesos

Aquí se describe cómo se llevarán a cabo distintas tareas que hay que realizar en distintos momentos del proyecto.

#### 3.1.1. Procesos de inicio de proyecto

- Cómo se van a identificar y asignar recursos (p.ej. conseguir servidores en cloud o teléfonos móviles para pruebas, pero también registrarse para acceder a API que se quieran integrar o a herramientas online que se quieran usar etc.).
- Cómo se va a abordar la formación inicial de los miembros del equipo (revisar qué tecnologías se van a usar, qué componentes se van a integrar, con qué API hay que conectar y quiénes tienen que formarse, o auto-formarse, en todas esas cosas y de qué manera (hacer algún curso online, planificar algo de tiempo para auto-formación con tutoriales y documentación etc.).

#### 3.1.2. Procesos de ejecución y control del proyecto

- Cómo se llevarán a cabo las comunicaciones internas, el registro de las decisiones tomadas en reuniones, la redacción de las actas etc.
- Cómo se van a determinar las tareas a realizar y el reparto de las mismas a integrantes del equipo en el día a día.
- Cómo se abordarán los temas de gestión del equipo (moral, resolución de disputas...).
- Qué se va a hacer respecto a medidas de progreso y monitorización del estado del proyecto (qué se mira/mide, cada cuánto tiempo, qué se hace si se detectan problemas de rendimiento o avance insuficiente o desviaciones respecto al plan inicial...).
- Cómo se hará la entrega de resultados.

#### 3.1.3. Procesos técnicos

- Describir los métodos, herramientas y técnicas necesarios tanto para construir el software (p.ej. herramientas de desarrollo), desplegarlo, probarlo (todos los necesarios para dar soporte a los planes descritos en la sección 3.2).

## 3.2. Planes

### 3.2.1. Plan de gestión de configuraciones

- Convenciones de nombres (documentos) y estándares de código.
- Responsable o responsables de las distintas actividades (puesta en marcha, apoyo al equipo, revisión de commits, copias de seguridad, control de las versiones entregadas a cliente...).
- Recursos: repositorios de control de versiones (cuáles, cuántos, permisos de acceso a los mismos) y sistema de gestión de incidencias.
- Procedimiento para realizar cambios al código fuente y los documentos técnicos: workflow de control de versiones utilizado, cuándo/cómo se permiten realizar commits al repositorio compartido, si tienen que ser aceptados por alguien previamente o no, qué hay que anotar en el sistema de gestión de incidencias, quién decide el estado de las incidencias, en qué estados puede estar una incidencia etc.

### 3.2.2. Plan de construcción y despliegue del software

- Cómo se construye e integra el software: si hay scripts de construcción automatizada o no (en ese caso qué se usa, y cómo se garantiza que todos los participantes compilan igual y con las mismas dependencias), qué se incluye en la construcción (descarga y actualización de dependencias, compilación, ejecución de tests automáticos...) y cada cuánto se construye (compila, integra, prueba) el sistema completo, cómo se configuran los computadores de los desarrolladores.
- Cómo se despliega el software más allá de las máquinas de desarrollo: contenedores, máquinas virtuales, servidor en cloud etc. y cómo se configuran esos entornos (rutas, usuarios y contraseñas, puertos y otros elementos).

### 3.2.3. Plan de aseguramiento de la calidad

- Estándares de código y otros (se pueden definir guías para la documentación de diseño y otros documentos del proyecto).
- Actividades de control de calidad del código que se realizarán: revisiones de código por pares, revisiones de requisitos o diagramas UML por pares, tipos de tests automáticos o manuales que se llevarán a cabo.

### 3.2.4. Calendario del proyecto y división del trabajo

- Diagrama de Gantt que recoja las tareas a realizar. Tened en cuenta que trabajáis con dos iteraciones y por tanto que hay una entrega intermedia y una final, y reflejarlo en este diagrama. Tened en cuenta que es normal que lo tengáis que actualizar conforme avance el proyecto (cuándo y cómo establezcáis en la sección 3.1.2).
- Debe quedar claro qué requisitos van a estar completados en la primera iteración y cuáles en la segunda. Es posible que para la primera iteración no se planifique completar ningún requisito, pero en ese caso tiene que planificarse qué se hará y que faltará por hacer para cada requisito.

- División del trabajo en partes (los módulos del software a desarrollar, pero también la documentación, el diseño gráfico, instalaciones o despliegues, pruebas manuales etc.) y reparto de los mismos entre el equipo de desarrollo, al menos a alto nivel (el reparto de labores concretas en el día a día no se detalla aquí, pero hay que explicar bajo qué criterios y quién/cómo se hace en la sección 3.1.2). Debe haber una correspondencia con las tareas que aparecen en el diagrama de Gantt (que no necesariamente tiene que ser una relación 1 a 1).
  - Verificar que esta división del trabajo cubre todos los requisitos.

## Capítulo 4

# Análisis y diseño del sistema

### 4.1. Análisis de requisitos

Completar y detallar los requisitos preliminares incluidos en la propuesta técnica y económica. Recordad que los requisitos deben ser completos, concretos, medibles cuando tenga sentido y lo menos ambiguos posible. También es importante que estén identificados para facilitar su trazabilidad.

### 4.2. Diseño del sistema

- Diagramas arquitecturales (de módulos, de componentes y conectores, de distribución), patrones de diseño y estilos arquitecturales que se aplicarán. Las interfaces (de módulos y de componentes) son especialmente importantes. También lo son los protocolos de comunicación entre componentes.
- Tecnologías elegidas (lenguajes de programación, componentes que se integrarán, API web externas con las que se conectará etc.).
- Otros aspectos técnicos de interés (p.ej. si hay base de datos si va a ser SQL o NoSQL, si hay una API Web va a ser RESTful o no, si algunas de las operaciones van a ser asíncronas o no, si va a ser una aplicación móvil o de escritorio será nativa o se van a usar tecnologías web, cómo se van a considerar los requisitos de seguridad o de prestaciones, cómo y dónde se harán las instalaciones y despliegues etc.)

Hay que justificar todas las decisiones de diseño. Esto exige contestar a dos preguntas sobre cada decisión: ¿qué alternativas se barajaron? y ¿por qué se eligió una y no las otras?

## Capítulo 5

# Memoria del proyecto

ESTE CAPÍTULO NO SE RELLENA EN LA PRIMERA ENTREGA En este capítulo se describirá cómo se ha llevado a cabo el proyecto, qué cambios se han hecho respecto a la versión inicial, imprevistos surgidos, etc.

### 5.1. Inicio del proyecto

Describir cómo transcurrió esta fase del proyecto, especialmente los resultados de llevar a cabo los procesos descritos en la sección Procesos de inicio del proyecto.

### 5.2. Ejecución y control del proyecto

Describir cómo transcurrió esta fase del proyecto, especialmente los resultados de llevar a cabo los procesos descritos en la sección Procesos de ejecución y control del proyecto y en la sección Procesos técnicos. No olvidar:

- Cómo se ha realizado el reparto de trabajo entre miembros del equipo. Cómo ha transcurrido la comunicación interna.
- Cómo se ha medido el progreso del proyecto. Cómo se sabía el trabajo realizado, el trabajo pendiente y lo que estaba haciendo cada persona.
- Los ajustes realizados cuando se detectaron divergencias frente al calendario inicial (ajustes en el trabajo y/o ajustes en el calendario). Si se han identificado las causas de estas divergencias, explicarlas.
- Adecuación de las herramientas y tecnologías empleadas. Si ha habido que cambiar alguna decisión de diseño o de tecnología, y por qué.
- Funcionamiento de los procesos de control de versiones del código, construcción y despliegue. ¿Ha habido problemas con las integraciones? ¿Problemas con los despliegues? ¿Se han perdido cosas por errores humanos? ¿Cómo se han abordado estas tareas?
- Pruebas del software. ¿Se han podido cumplir las ideas que se tenían al respecto?

### 5.3. Cierre del proyecto

Al menos:

- Comparar las estimaciones iniciales (tamaño, esfuerzos, costes) con los resultados finales, analizar los resultados y tratar de expresar algunas lecciones aprendidas.

- Lecciones aprendidas sobre herramientas y tecnologías.
- Recopilar los esfuerzos dedicados al proyecto por cada uno de los participantes: horas trabajadas y actividades realizadas por cada persona.



## Capítulo 6

# Conclusiones

ESTE CAPÍTULO SOLO SE RELLENA EN LA ENTREGA FINAL Además de conclusiones personales (razonadas) sobre el transcurso del proyecto realizado, es importante plantear ideas para mejorar los procesos llevados a cabo: si hubiera que iniciar un nuevo proyecto inmediatamente usando una metodología de gestión basada en procesos, ¿qué cambios haríais respecto a los procesos que habéis seguido durante este proyecto? ¿Qué cosas está claro que haríais de otra forma? ¿Qué cosas seguiríais haciendo más o menos igual?

## Apéndice A

# Dedicación

### A.1. Dedicación

CUADRO A.1: Dedicación a la práctica de los integrantes del grupo.

Integrante	Dedicación (h)	Partes desarrolladas
Ángel Cañal	40	▪ A
Abel Chils	40	▪ A
Jorge Aznar	40	▪ A
Yasmina Albero	40	▪ A
Óscar Fraca	40	▪ A
Alexandru Oarga	40	▪ A
Jorge Pinilla	40	▪ B
<i>Ambos</i>	—	▪ Diagramas de secuencia de diseño prácticas