Gestión de equipos

El objetivo de este documento es establecer en el equipo una mecánica de trabajo que nos permita satisfacer una desiderata cualitativa de buen diseño software.

Concretamente, los factores principales a los que estamos atendiendo son:

* Fomentar la fluidez de la comunicación entre los integrantes del grupo, usando para ello herramientas que nos permitan dejar constancia de toda la información importante discutida.
* Garantizar una distribución satisfactoria de las tareas, que evite la repetición innecesaria de trabajo y ayude a los integrantes del equipo a identificar tareas pendientes y encargarse de que se vean cumplidas en los plazos apropiados. Para satisfacer este requisito necesitamos poder consultar en qué está trabajando cada uno de nuestros compañeros en cada momento.
* Fomentar el control de calidad a través de revisiones continuas.
* Otras gilipolleces

A continuación procedemos a realizar un análisis de las características del proyecto y equipo, culminando con una propuesta de solución adecuada a los resultados de dicho análisis.

## Análisis de factores Mantei del proyecto

### Tamaño del proyecto

Estimamos que en codificación el proyecto rondará las 10,000 LDC[[1]](#footnote-2).

Esto lo coloca dentro de los proyectos pequeños-medios.

### Dificultad

Las dificultades técnicas del proyecto no son demasiadas. Sin embargo, la falta de experiencia del equipo[[2]](#footnote-3) puede suponer un lastre, especialmente en lo referente a la gestión del proyecto.

Trataremos de paliar esta debilidad con una gestión de equipos rigurosa y clara.

### Duración

El proyecto se alargará temporalmente durante 6 meses. Una vez entregado no habrá responsabilidades de mantenimiento, que pasará a ser una preocupación del cliente.

Esto nos lleva a clasificar el proyecto como un proyecto de corta duración.

### Modularidad

El proyecto se descompone en varios entregables que son analizados en detalle en la planificación. Clasificamos el proyecto con un nivel de modularidad media.

### Fiabilidad

El sistema va a ser utilizado en la red interna de una empresa, luego no va a estar en una posición vulnerable a ataques de agentes externos. Hay cierta preocupación por evitar que los empleados trastoquen el sistema, pero esto se refleja en un diseño que no les invita a abusar de sus posibilidades (distintos niveles de usuario, restricciones de edición en proyectos a los que no perteneces, etc).

Además, los clientes son conscientes de la falta de experiencia del equipo, y sus expectativas están calibradas acordemente.

Por todo ello concluimos que la fiabilidad no es una prioridad de diseño.

### Fecha de entrega

### Comunicación requerida

Como es norma general en un proyecto de ingeniería del software, la comunicación entre los componentes del equipo es fundamental.

Esta necesidad de comunicación entre los miembros del equipo se ve acentuada por la falta de experiencia del equipo. Durante el proyecto tendremos que experimentar con modelos de trabajo nuevos, descubriendo a través de la práctica qué funciona y qué no.

Trataremos de agilizar el proceso en la medida de lo posible fomentando la comunicación efectiva entre los miembros del equipo, en combinación con un registro exhaustivo de las actividades realizadas para poder realizar un análisis de datos eficaz.

## Análisis de la composición del equipo

### Tamaño del equipo

Nuestro equipo cuenta con una plantilla de 6 miembros. Juzgamos que el número corresponde a un tamaño de equipo medio-pequeño.

Atendiendo al tamaño del equipo, creemos conveniente una subdivisión en parejas que se hagan cargo de las tareas. Esto facilita la asignación de responsabilidades y nos permite adoptar naturalmente mecánicas de pair programming (extendidas no sólo a codificación sino a desarrollo de artefactos).

### Especialización

Todos los integrantes del equipo son estudiantes en curso de Segundo de Informática, y comparten una base de conocimientos formada por las asignaturas cursadas hasta el momento.

Esto se traduce en las siguientes habilidades técnicas relevantes para el proyecto:

* Programación estructurada, modular y orientada a objetos (C++ y Java).
* Gestión de proyectos a nivel iniciación.
* Diseño algorítmico y verificación nivel básico.

Durante el transcurso del proyecto trataremos de formar a los integrantes para que desarrollen las siguientes habilidades (entre paréntesis algunas tecnologías con las que deberemos familiarizarnos):

* Gestión de repositorios (GitHub)
* Gestión de proyectos nivel intermedio (PivotalTracker)

La falta de especialización entre los distintos miembros de grupo nos permite optar por una gestión flexible, aunque dificulta el reparto de responsabilidades.

### Disponibilidad temporal y espacial

Cada miembro del equipo se estima que dedique al proyecto 4 horas por semana, 3 durante las vacaciones de invierno.

Existe la posibilidad de hacer reuniones presenciales, aunque se prefiere en general el trabajo remoto, exceptuando las horas de trabajo en clase.

Por ello vamos a usar usar tecnologías que permitan dejar constancia de las conversaciones tenidas (Issues de GitHub, chat de PivotalTracker). Las horas de trabajo en clase serán aprovechadas para planificar y votar medidas que afecten al grupo.

## Solución propuesta

Según el modelo de Mantei el pequeño tamaño, la baja fiabilidad, los plazos cortos de fechas de entrega y la alta necesidad de comunicación son todos alicientes para optar por un modelo **descentralizado democrático**.

1. En un proyecto anterior hecho por el autor, un gestor de mail con funcionalidades de organización parecidas a las requeridas, se alcanzaron las 4700 LDC. Dicho proyecto no contaba con interfaz gráfica, y contamos con que este proyecto sea más complicado, así que hemos optado por un factor 2 comparativo, llevándonos a 4700\*2 ~= 10,000 LDC. [↑](#footnote-ref-2)
2. Véase el análisis de especialización [↑](#footnote-ref-3)