

# Índice general

<b>1. Descripción y Planificación del Proyecto</b>	<b>1</b>
1.1. Descripción Funcional del Sistema . . . . .	1
1.2. Metodología de Desarrollo . . . . .	2
1.3. Requisitos de Alto Nivel del Sistema . . . . .	2



# Índice de figuras



# Índice de cuadros



# Capítulo 1

## Descripción y Planificación

El presente capítulo describe «a completar»

### Contents

<b>1.1. Descripción Funcional del Sistema . . . . .</b>	<b>1</b>
<b>1.2. Metodología de Desarrollo . . . . .</b>	<b>2</b>
<b>1.3. Requisitos de Alto Nivel del Sistema . . . . .</b>	<b>2</b>

### 1.1. Descripción Funcional del Sistema

En la siguiente sección se describe el ámbito funcional del sistema, en otras palabras, las características que se espera que tenga la aplicación deseada. Definirlas de modo correcto es muy importante, ya que los requisitos de alto nivel dependen directamente y si estos son incorrectos, obtendríamos una aplicación inútil que no cumple con lo requerido.

El proyecto consiste en desarrollar una aplicación que, utilizando la red local instalada en los laboratorios de la Facultad de Ciencias, sea capaz principalmente de, desde un computador distinguido, el del profesor, controlar el uso del resto de los computadores, los de los alumnos, que hagan de la red. Está especialmente orientado a denegar accesos no deseados durante pruebas evaluables, evitando por ejemplo, la realización de un test pudiendo consultar dudas en internet, sin necesitar desconectar físicamente la red.

Esto es deseable puesto que en ocasiones es necesario que los alumnos dispongan de ciertos archivos al comenzar, y recoger los archivos resultantes al final de la prueba, con esta aplicación se permite, aprovechando la conexión existente entre computadores, el envío y recogida de estos archivos, garantizando transferencias correctas, de un modo más cómodo y sencillo, que el profesor vaya con un dispositivo de almacenamiento recogiendo los resulta-

dos individualmente cuando un alumno finaliza.

El componente que se ejecutará en el computador del profesor ha de ser capaz de permitir conectarse a todos los alumnos que vayan a realizar la prueba, y como se ha comentado, al principio de la misma, si se desea, enviar un archivo dado, a los alumnos conectados a modo de enunciado, y notificar al resto de computadores el inicio de la prueba y que, por tanto, el acceso a la red ya no está permitido, del mismo modo, cuando cada alumno decide que ha acabado, en caso de que entreguen ficheros como resultado, recogerlos y almacenarlos en el computador del profesor, en un directorio seleccionado por él, sin necesidad de intervención del mismo, además de volver a permitir el acceso a la red al computador del alumno que concluye la prueba.

Cuando el profesor decide iniciar una prueba puede optar por establecer una hora límite para la finalización automática, o no, en ambos casos se permite concluirlo manualmente por el profesor.

Además, se guardarán registros de cada prueba con el fin de tener una noción de lo que está ocurriendo en la prueba actual o en alguna pasada, por ejemplo, la hora a la que empieza, a la que finaliza, cuándo acaba un alumno, si se recogen ficheros de resultados, dónde se almacenan.

Las funcionalidades más importantes presentadas al usuario en el componente a ejecutar por el alumno son dos, conectarse al computador del profesor y enviar el archivo de resultados al decidir finalizar la prueba.

Tanto el profesor como los alumnos dispondrán de interfaces gráficas sencillas e intuitivas para facilitar lo máximo posible el uso de la aplicación.

## 1.2. Metodología de Desarrollo

## 1.3. Requisitos de Alto Nivel del Sistema

En esta sección se enumeran los requisitos de alto nivel del sistema. Un requisito es una propiedad que debe ser exhibida por un software para resolver un problema particular. Los requisitos han de tener ciertas características, entre ellas ser:

- *No ambiguos*, no puede haber varias interpretaciones para un requisito puesto que se puede optar por una solución no deseada por el cliente.
- *Entendible*, se comprende fácilmente el significado.
- *Verificable*, es necesario que existan técnicas para comprobar que cada requisito es construido correctamente.



El conjunto de requisitos ha de ser, entre otras cosas:

- *Completo*, de tal forma que todo lo que deba hacer la aplicación esté recogido.
- *Consistente*, no deben existir conflictos entre requisitos.
- *No redundante*, es decir, que un problema sólo lo resuelva un único requisito.

Requisitos de alto nivel:

1. Profesor - Comenzar prueba
2. Profesor - Enviar enunciado
3. Profesor - Establecer hora fin prueba
4. Profesor - Finalizar prueba
5. Denegar acceso a red al comenzar
6. Permitir acceso a red al finalizar
7. Comprobación de integridad de los archivos transmitidos
8. Crear y guardar logs en el computador profesor
9. Alumno - Conectarse a profesor
10. Alumno - Enviar resultado