



Universidad Complutense

Facultad de Informática



eGorilla

Gestión de Riesgos

Asignatura: Ingeniería del Software

Curso Académico: 2008/2009

Grupo: 4º B

Índice

1.	INTRODUCCIÓN	5
2.	IDENTIFICACIÓN DE RIESGOS	6
2.1.	RIESGOS DE PROYECTO.	7
2.2.	RIESGOS DE PRODUCTO.	8
2.3.	RIESGOS DE NEGOCIO.	9
3.	ANÁLISIS DE RIESGOS	10
4.	PLANIFICACIÓN DE RIESGOS.....	12
5.	SEGUIMIENTO DE RIESGOS	15



FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

1. Introducción

El propósito del presente documento es dar una **visión general de los riesgos**, tanto conocidos como predecibles, que se prevé que pueden presentarse a lo largo del desarrollo del **proyecto P2P eGorilla**, junto con las medidas que se adoptarán para resolverlos y/o minimizar su impacto.

La gestión de riesgos se encarga de identificar y tratar, mediante la elaboración de planes, los riesgos que pueden aparecer a lo largo de un proyecto y minimizar su efecto en el mismo, para asegurar así el éxito de la aplicación.

Hemos de definir primero que es un riesgo. El cuál, es la probabilidad de que se dé una **circunstancia adversa**.

El proceso de gestión de riesgos conlleva las siguientes **fases**:

- **Identificación de riesgos.**
 - **Identificar** riesgos de proyecto, producto y negocio.
- **Análisis de riesgos.**
 - Evaluar la **probabilidad** y las **consecuencias** del riesgo.
 - Dar **prioridades** a los riesgos.
- **Planificación de riesgos.**
 - Elaboración de planes para **evitar** o **minimizar** los riesgos.
- **Seguimiento de los riesgos.**
 - Llevar una **vigilancia** de los riesgos durante el proyecto.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

2. Identificación de riesgos

La identificación de riesgos es la **primera etapa** de la gestión de riesgos. En este apartado se intentará obtener la lista de los principales riesgos que pueden afectar al proyecto. Es de vital importancia pues los riesgos afectan a los acontecimientos futuros e implican cambio, elección e incertidumbre.

La identificación del riesgo es un intento sistemático para especificar las amenazas al plan del proyecto (estimaciones, planificación temporal, carga de recursos, etc.). Identificando los riesgos conocidos y predecibles, se intentará dar un paso adelante para evitarlos cuando sea posible y controlarlos cuando sea necesario.

Para facilitar el análisis y la comprensión, se clasificarán los riesgos en:

- **Riesgos de proyecto:** relativos al calendario o a los recursos.
- **De producto:** relativos a la calidad o el rendimiento del software.
- **De negocio:** relativos a la organización que desarrolla el software.

Nótese que esta clasificación **no es cerrada** y un riesgo puede ubicarse en varias categorías.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

2.1. Riesgos de proyecto.

Amenazan el plan del proyecto. Si los riesgos se presentan es probable que la planificación se atrase y los costes aumenten.

- **Cambios en el grupo de desarrollo (altas y bajas):** Provocan dificultades de **organización** (**Comunicación** entre los distintos miembros del equipo) y de **planificación** (**Asignación de tareas, incumplimiento de hitos propuestos -> retrasos**)
- **Integración de las distintas partes de las que se compone el proyecto** llevadas a cabo por diferentes subgrupos dentro del grupo de desarrollo.
- **Organización de los miembros del grupo de desarrollo:** Con una estructura lineal en la que todo el mundo tiene el mismo poder de decisión se puede producir el desconcierto sobre quién y cuando tiene que realizar alguna tarea a no ser de que los todos los miembros actúen de buena fe y tomen la iniciativa.
- **Comunicación entre los distintos miembros del grupo de desarrollo.** Si la comunicación entre los diferentes miembros es insuficientes o no es buena puede ser el origen del fracaso de todo el proyecto.
- **Restricciones en el uso de laboratorios:** tanto en la no disponibilidad de software/hardware necesario, como de horarios de utilización. El rendimiento de los puestos de laboratorios también dificulta el desarrollo del proyecto dentro de la universidad.
- **Subestimación de la complejidad o el tamaño del proyecto:** Aparentemente un optimismo inicial sobre las grandes posibilidades que ofrece el proyecto puede llevar a los desarrolladores a abarcar más áreas de las que realmente, por diversas cuestiones, no pueden llegar a alcanzar.
- **Cambios importantes** respecto de los **requisitos iniciales**, puede provocar retrasos en las entregas e incluso la posibilidad de poder continuar con el proyecto estimado.
- **Formación insuficiente o poca experiencia del personal del equipo** en el **ámbito técnico del proyecto** (lenguaje java, protocolos de comunicaciones, etc.).
- **Formación insuficiente o inexperiencia del personal del equipo en el uso de los entornos/herramientas** de desarrollo que se van a utilizar en el proyecto.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

2.2. Riesgos de producto.

Este tipo de riesgos amenazan la calidad y el rendimiento del software que hay que producir. Se han identificado los siguientes:

- Dificultades de **diseño y funcionamiento del protocolo eGorilla**.
- Problemas derivados de la **utilización de otros componentes** (bibliotecas de clases java, etc.) para el desarrollo del mismo.
- Dificultad para definir una **arquitectura adecuada y estable**.
- **Incompatibilidad** de la aplicación con las máquinas virtuales de java de diferentes sistemas operativos.
- **Permisos necesarios** para la ejecución correcta de la aplicación.
- **Recursos del equipo** no accesibles para la aplicación, como por ejemplo no poder utilizar puertos de comunicación, disco duro, etc.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

2.3. Riesgos de negocio.

Los riesgos de negocio amenazan la viabilidad del software a construir y a menudo ponen en peligro el producto a desarrollar.

- **Construir un producto o sistema excelente que no quiere nadie en realidad** (riesgo de mercado). Como consecuencia de la naturaleza del sistema a desarrollar, éste se vería perjudicado en el sentido de que es más eficiente y útil cuanto mayor número de usuarios intervenga.
- **Problemas legales** debido a que el presente proyecto se encuadra en un marco en el que existe un **vacío legal** en lo que concierne al intercambio de archivos por red. El 'quid' legal del P2P es su peculiar forma de funcionar: cuando se descarga un archivo, a la vez se está compartiendo, poniéndolo a disposición de otras personas.
- **Problemas de distribución** debido a que el servidor de distribución no funcione en un momento determinado.
- **Problemas de mantenimiento** de la aplicación P2P, pues muchos fallos críticos surgirán con la distribución de la primera versión y será necesario que sean solucionado de manera casi inmediata. Si no es así los usuarios buscaran otras aplicaciones que funcionen correctamente.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

3. *Análisis de riesgos*

El análisis de riesgos es la siguiente etapa en la gestión de riesgos. El objetivo que pretende es estimar, para cada riesgo identificado anteriormente, su probabilidad de ocurrencia y las consecuencias del mismo. En base a estos valores se podrá priorizar los riesgos para tratarlos consecuentemente.

La probabilidad del riesgo podrá ser:

- **Muy baja.**
- **Baja.**
- **Moderada.**
- **Alta.**
- **Muy alta.**

Las consecuencias del riesgo podrán ser:

- **Catastróficas.**
- **Serias.**
- **Tolerables.**
- **Insignificantes.**





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

A continuación se muestra la lista de los riesgos valorados, mostrándose en **negrita** los riesgos que se han considerado más importantes.

Riesgos	Probabilidad	Consecuencias
1. Cambios en el grupo de desarrollo.	Muy alta	Tolerables
2. Integración de las distintas partes de las que se compone el proyecto.	Alta	Tolerables
3. Organización de los miembros del grupo de desarrollo.	Alta	Tolerables
4. Comunicación entre los distintos miembros del grupo de desarrollo.	Muy alta	Tolerables
5. Restricciones en el uso de laboratorios	Muy alta	Tolerables
6. Subestimación de la complejidad o el tamaño del proyecto.	Moderada	Serias
7. Cambios importantes respecto de los requisitos iniciales.	Baja	Serias
8. Formación insuficiente o poca experiencia del personal del equipo en el ámbito técnico del proyecto.	Alta	Tolerables
9. Formación insuficiente o inexperiencia del personal del equipo en el uso de los entornos/herramientas de desarrollo.	Alta	Tolerables
10. Dificultades de diseño y funcionamiento del protocolo eGorilla.	Moderada	Serias
11. Problemas derivados de la utilización de otros componentes.	Baja	Tolerables
12. Dificultad para definir una arquitectura adecuada y estable.	Moderada	Serias
13. Incompatibilidad de la aplicación con las maquinas virtuales de java de diferentes sistemas operativos.	Baja	Tolerables
14. Permisos necesarios para la ejecución correcta de la aplicación.	Baja	Tolerables
15. Recursos del equipo no accesibles para la aplicación.	Baja	Tolerables
16. Construir un producto que nadie va a utilizar.	Moderada	Tolerables
17. Problemas legales derivados del vacío legal existente en lo relativo al P2P.	Baja	Seria
18. Problemas de distribución.	Baja	Tolerables
19. Problemas de mantenimiento.	Moderada	Tolerables





4. Planificación de riesgos

El propósito de la etapa de planificación de riesgos es **valorar cada riesgo identificado y proponer una estrategia para resolverlo**.

Las estrategias de resolución de riesgos pueden estar **orientadas a evitar el riesgo** (reduciendo su probabilidad de ocurrencia), **a minimizarlo** (reduciendo sus consecuencias), **y/o a desarrollar un plan de actuación** para afrontarlo en caso de que se produzca.

A continuación se muestran los riesgos junto con las estrategias que se han establecido.

Riesgos	Estrategia
1. Cambios en el grupo de desarrollo.	<ul style="list-style-type: none">Para las altas se les mantendrá informados desde el principio del avance del proyecto para que el tiempo de adaptación y formación sea mínimo. Generar documentación suficiente para que puedan ponerse al día. Se les dará permisos al repositorio del proyecto.Para las bajas compartir conocimiento. Los aspectos clave del proyecto deben ser conocidos por más de una persona. Se revisará que todo el trabajo realizado por la persona que se va a dar de baja esta documentado y se exigirá a la persona que se va a dar de baja que informe del trabajo pendiente para poder ser continuado por otra persona. <p>Se procederá a la reasignación de tareas en función del estado del proyecto.</p>
2. Integración de las distintas partes de las que se compone el proyecto.	<p>Se procederá a definir correctamente las interfaces de cada módulo de la aplicación. Si existe conflicto con las interfaces se procederá al retoque de los diseños de los módulos implicados en el conflicto para una rápida adaptación común de los mismos. Los encargados de cada módulo afectado por la nueva integración de un nuevo módulo definirán las partes a modificar en sus módulos de mutuo acuerdo antes de realizar ningún cambio.</p>
3. Organización de los miembros del grupo de desarrollo.	<p>La figura del jefe de proyecto mantendrá actualizada la lista de tareas personales de cada uno de los miembros del grupo de forma diaria para gestionar de una forma más segura los posibles cambios que se puedan llegar a producir en el número de personas del grupo de trabajo. El jefe dividirá el trabajo en diferentes grupos atendiendo a la arquitectura del sistema, los cuales tendrán una organización horizontal.</p>
4. Comunicación entre los distintos miembros del grupo de desarrollo.	<p>Se procederá a indicar las tareas individuales de cada individuo dentro de un subgrupo definido de forma clara y rigurosa con un cierto tiempo de antelación para que esa nueva reasignación afecte lo menos posible al rendimiento de la persona o personas afectadas. Para las comunicaciones se hará uso del Google Group creado para el proyecto y que reside en la dirección http://groups.google.es/group/egorilla.</p>





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

5. Restricciones en el uso de laboratorios	El software/hardware que no esté disponible será aportado por los miembros del equipo. Las restricciones de horario se salvarán con trabajo extra fuera del laboratorio.
6. Subestimación de la complejidad o el tamaño del proyecto.	Seguimiento especial del proyecto, especialmente en la etapa de análisis y en menor aunque no menos importante en la etapa de diseño . Se tendrán como principales objetivos a desarrollar el funcionamiento básico de la aplicación.
7. Cambios importantes respecto de los requisitos iniciales.	Especial cuidado en la especificación de requisitos y validación del diseño con ellos. Con posterioridad sólo se permitirán cambios que sean imprescindibles y estén muy justificados . Se prestará especial cuidado en el diseño de la arquitectura de la aplicación, distribuyendo la funcionalidad en diferentes módulos, para evitar el mayor daño posible frente a los posibles cambios futuros.
8. Formación insuficiente o poca experiencia del personal del equipo en el ámbito técnico del proyecto.	Compartición de conocimientos y/o experiencia entre el equipo de desarrollo así como consulta de manuales, libros, páginas web, etc. relativos al marco en el que se encuentra la aplicación.
9. Formación insuficiente o inexperiencia del personal del equipo en el uso de los entornos/herramientas de desarrollo.	Compartición de conocimientos y/o experiencia entre el equipo de desarrollo así como consulta de manuales, libros, páginas web, etc. relativos al marco en el que se encuentra la aplicación.
10. Dificultades de diseño y funcionamiento del protocolo eGorilla.	Consultar información en Internet, páginas de e-mule, sourceforge, etc. relativas a la aplicación a desarrollar. Se emularán los conceptos abstractos del funcionamiento del protocolo mediante reuniones colectivas.
11. Problemas derivados de la utilización de otros componentes.	Uso bibliotecas de clases especializadas alternativas (y de libre distribución) si resultasen necesarias. Evitar en la mayoría de lo posible hacer uso de bibliotecas específicas de los IDEs como NETBEANS.
12. Dificultad para definir una arquitectura adecuada y estable.	Hacer énfasis especial en la etapa de definición de la arquitectura mediante la implicación de todos los miembros del proyecto siendo consensuada por todos ellos y siendo validada en última instancia por el jefe de proyecto.
20. Incompatibilidad de la aplicación con las máquinas virtuales de java de diferentes sistemas operativos.	Investigar que versiones de la máquina virtual de java existe para los diferentes SO del mercado. Si en algún sistema operativo no existe la versión 1.6 de la JVM realizar una migración de la aplicación a la última versión de ese SO quitando las funcionalidades que no soporte de la misma. Informar a los usuarios de ese SO de las funcionalidades omitidas para su SO. Si el proceso fuera muy costoso informar al usuario que la aplicación no es compatible con su SO.
21. Permisos necesarios para la ejecución correcta de la aplicación.	Para la ejecución correcta de la aplicación detectar cuales son los privilegios necesarios e informar al usuario de los mismos a la hora de realizar la instalación.
22. Recursos del equipo no accesibles para la aplicación.	Detectar que recursos mínimos son necesarios para la ejecución correcta de la aplicación e informar al usuario de los mismos.
23. Construir un producto que nadie va a utilizar.	Hacer énfasis especial en el rendimiento y prestaciones de la aplicación . Así como en características nuevas que no ofrezcan los demás productos del mercado.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

	Creación de diferentes campañas publicitarias del producto (pagina web,...).
24. Problemas legales derivados del vacío legal existente en lo relativo al P2P.	Mantenerse informado de las novedades que se produzcan en este ámbito para ver cómo podrían afectar al sistema.
25. Problemas de distribución.	Asegurarse de tener un servidor de distribución estable y si los costes lo permiten contratar un segundo servidor de apoyo.
26. Problemas de mantenimiento.	Tener un equipo especializado en mantener, actualizar y resolver todos los problemas surgidos con la aplicación. Crear una web o correos de administración con los que el cliente pueda contactar para exponer sus quejas o posibles mejoras de la aplicación. Con ello aseguramos un canal de retroalimentación de la aplicación.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

5. Seguimiento de riesgos

Por último esta etapa se encarga de **evaluar periódicamente** los riesgos identificados, actualizando, para cada riesgo, su probabilidad y sus consecuencias. El seguimiento de riesgos se realizará sobre las **reuniones periódicas de proyecto**, poniéndose énfasis en los riesgos clave que puedan afectar más a la evolución del proyecto.

Los riesgos que más se han detectado y por tanto actualizando sus probabilidades y consecuencias han sido los siguientes:

- **Altas y bajas de nuevos miembros** al equipo a mitad del proyecto. En total se han producido **5 altas (Sergio, Daniel, Qiang, José Miguel e Iván)**, de las cuales **2 abandonaron el proyecto (Daniel y Sergio)** junto con otro miembro original del grupo de desarrollo (**Fran**). Se ha producido una baja intermitente de **Víctor Adaíl Ferrer**, durante el mes de Abril, por lo que su trabajo ha sido cambiado a la ayuda del diseño de comunicaciones de la aplicación y revisión de los documentos del proyecto.
- **Restricciones de material y de horario en el uso de laboratorios.** Se ha tenido una reunión con los técnicos de los laboratorios, para conocer más a fondo cuales eran las limitaciones de los mismos. Las conclusiones de la reunión son que las redes inalámbricas solo permiten el acceso a internet, dejando bloqueados todo tipo de puertos. La red Ethernet dentro del laboratorio y con los equipos del laboratorio tiene menos restricciones que la red inalámbrica pues los puertos de uso no específico están abiertos y disponibles.
- **Mala planificación del proyecto** provocada por el continuo cambio en el número de miembros del grupo de desarrollo. Ello ha provocado el incumplimiento de algunos hitos importantes en la planificación del proyecto y la continua reasignación de tareas de los miembros del grupo de desarrollo. Gracias a la incorporación como jefe de proyecto de Javier Salcedo se han solucionado los problemas de planificación existentes anteriormente.
- **Dificultades de comunicación** entre los miembros de los subgrupos que realizan el desarrollo del proyecto para realizar tareas de coordinación entre ellos. Se ha experimentado una mejor comunicación entre los diferentes grupos aunque debido a la complejidad del gestor de red sigue habiendo algunos problemas, que se están solucionando con reuniones extra entre los implicados.





FACULTAD DE INFORMÁTICA	UNIVERSIDAD COMPLUTENSE DE MADRID	
Ingeniería del Software	Curso 2008/2009	Grupo: 4º B

- **Dificultades de integración** de los distintos módulos que componen la aplicación producidas precisamente por esos problemas de comunicación lo que ha repercutido sobretudo en que algunos módulos han restado o han cambiado el planteamiento inicial de otros que estaban relacionados con el (GestorDeRed y GestorDeArchivos).

