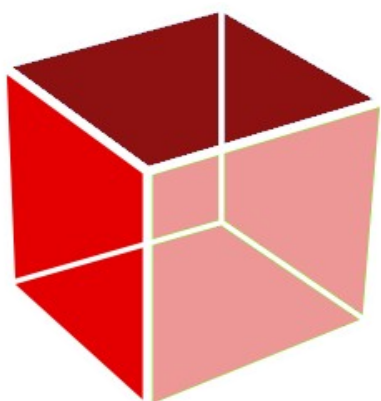


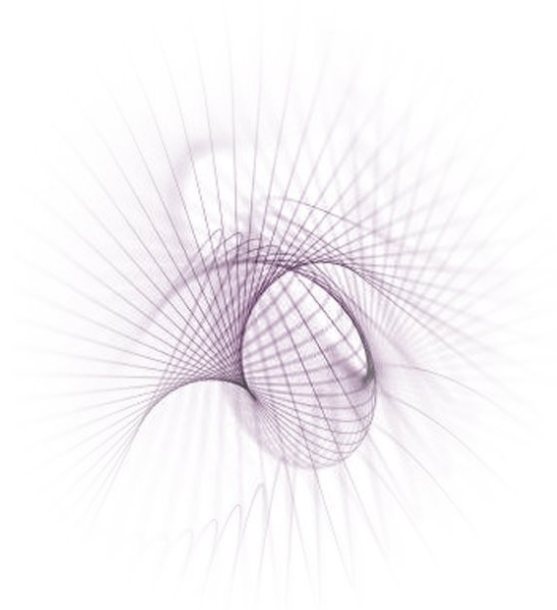
MEMORIA DEL PROYECTO



REDMETRICS



www.uoc.edu



Versión: 1.0

Fecha: 07/06/11



CONTROL DE FIRMAS Y VERSIONES

Versión	Fecha	Autor	Descripción
0.1	25/05/11	Carlos Parra Camargo	Estructura del documento.
0.2	27/05/11	Carlos Parra Camargo	Introducción y primer esbozo de origen del proyecto.
0.3	28/05/11	Carlos Parra Camargo	Origen del proyecto y primeras entradas de bibliografía.
0.4	29/06/11	Carlos Parra Camargo	Metodología, infraestructura y ciclo de vida del proyecto.
0.5	02/06/11	Carlos Parra Camargo	Redacción de conclusiones, se incluyen gráficas varias para facilitar la comprensión.
1.0	03/06/11	Carlos Parra Camargo	Repaso y ajuste de formato.

CONTROL DE DIFUSIÓN



Reconocimiento - CompartirIgual (by-sa): Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.



ÍNDICE DE CONTENIDO

1.-INTRODUCCIÓN.....	4
1.1.-¿QUÉ ES REDMETRICS?.....	4
1.2.-FICHA DEL PROYECTO.....	5
1.3.-PRINCIPIOS DE IMPLEMENTACIÓN.....	6
2.-ORIGEN DEL PROYECTO.....	7
3.-METODOLOGÍA DE DESARROLLO.....	8
4.-INFRAESTRUCTURA Y HERRAMIENTAS.....	9
5.-CICLO DE VIDA DE REDMETRICS.....	13
5.1.- PLANIFICACIÓN.....	13
5.1.1.-DEFINICIÓN DE OBJETIVOS.....	13
5.1.2.-ANÁLISIS DEL ESTADO DEL ARTE.....	13
5.1.3.-DEFINICIÓN DEL ALCANCE.....	14
5.1.4.-ELABORACIÓN DEL PLAN.....	15
5.2.- ANÁLISIS.....	16
5.2.1.-CREACIÓN DEL CATÁLOGO DE REQUISITOS.....	16
5.2.2.-CREACIÓN DEL ANÁLISIS FUNCIONAL.....	19
5.3.- DISEÑO.....	20
5.3.1.-DISEÑO DEL MODELO.....	21
5.3.2.-DISEÑO DE LA VISTA.....	22
5.3.3.-DISEÑO DEL CONTROLADOR.....	24
5.4.- IMPLEMENTACIÓN DEL PROTOTIPO.....	24
5.4.1.-DEFINICIÓN DEL PROTOTIPO.....	24
5.4.2.-IMPLEMENTACIÓN DEL PROTOTIPO.....	25
6.-CONCLUSIONES.....	26
7.-BIBLIOGRAFIA.....	28
ANEXO I: RELACIÓN DE GRÁFICAS.....	29
ANEXO II: RELACIÓN DE TABLAS.....	30





1.- INTRODUCCIÓN

1.1.- ¿QUÉ ES REDMETRICS?

Redmetrics es un proyecto de Software Libre cuya finalidad es la creación de una aplicación de gestión y control que facilite la planificación y seguimiento de proyectos de desarrollo de software.

La aplicación se fundamenta en las siguientes tres ideas:

- La **gestión de proyectos** de software contiene dos actividades bien diferenciadas: **planificación y seguimiento**.
- El **desarrollo de software es un proceso dinámico** que puede llegar a requerir varias planificaciones a lo largo del ciclo de vida de un proyecto.
- Tres de los ámbitos de estudio más relevantes en la planificación y seguimiento de un proyecto son el **tiempo**, el **esfuerzo** y el **coste**.

En definitiva, Redmetrics es un proyecto orientado a facilitar la realización de dos actividades claves en la gestión de proyectos (planificación y seguimiento) centrando el estudio en métricas de tiempo, esfuerzo y coste.

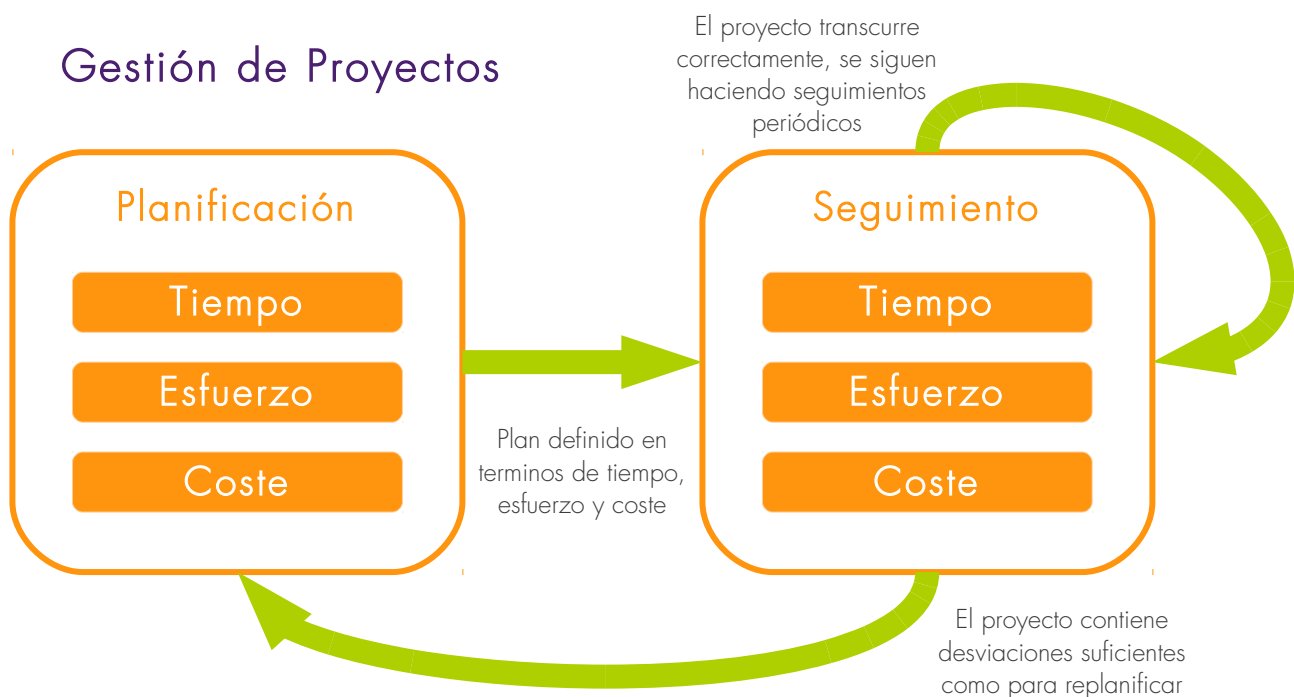


Gráfico 1: Flujo de actividades





El propio Plan de Proyecto define como principal objetivo el siguiente:

«Aportar en las soluciones de medición y análisis de la gestión de proyectos y grupos en el ámbito de la ingeniería de software y el software libre.»

1.2.- FICHA DEL PROYECTO

Nombre	Redmetrics
Descripción	Aplicación para la planificación y seguimiento de proyectos de desarrollo de software.
Fecha de creación	Febrero de 2011
Última versión liberada	Mayo de 2011
Licencia	GNU GPL v3
Plataforma	Linux, Windows, Mac (toda plataforma que soporte Ruby)
Autor	Carlos Parra Camargo
Página Web	http://code.google.com/p/redmetrics/
Contacto	<carlospc@gmail.com>
Logo	<div><p><i>Gráfico 2: El logo elegido para el proyecto es un cubo en perspectiva; representa la perfección de las métricas y las distintas perspectivas de una misma verdad.</i></p></div>

Tabla 1: Ficha resumen del proyecto





1.3.- PRINCIPIOS DE IMPLEMENTACIÓN

El proyecto se ha desarrollado siguiendo una serie de principios entre los que cabe destacar que Redmetrics está compuesto por **Software Libre** al 100%. De la misma forma, el proceso de desarrollo se ha apoyado en herramientas de software que, de la misma manera, son libres.

Redmetrics está implementado en su totalidad en Ruby por lo que cualquier extensión o modificación del proyecto deberá estar implementado en este lenguaje. Para facilitar la **legibilidad** del código, se han seguido las recomendaciones citadas por las *Ruby Style Guidelines*¹.

En los desarrollos llevados a cabo, se prima la orientación a objetos por delante de la programación estructurada. De esta manera, se pretende que el proyecto esté preparado para una mejor **escalabilidad** y tiene una organización más estructurada y reutilizable.

Por último, la **eficiencia y eficacia** también se han tenido más que presentes en el desarrollo. Se realizan multitud de consultas a la base de datos para la realización de las métricas por lo que se ha tenido especial cuidado en optimizar las consultas y en cachear los datos reutilizables entre distintas métricas.

¹ <http://www.caliban.org/ruby/rubyguide.shtml#style>





2.- ORIGEN DEL PROYECTO

Los centros de producción de software de hoy en día tienen multitud de herramientas y metodologías sobre las que elegir. Actualmente, las metodologías ágiles disfrutan de una gran popularidad y aceptación y están conviviendo con otros modelos de desarrollo de software como puede ser CMMI.

En esta línea, existe un interesante artículo del Software Engineering Institute titulado *CMMI or Agile: Why Not Embrace Both?*². Dicho artículo recoge los valores positivos de ambos modelos y da una serie de pautas para combinar las mejores prácticas. El estudio concluye destacando el liderazgo de CMMI en lo que a procesos se refiere y el éxito del método ágil por centrarse en las personas. Por contra, queda un tercer pilar desatendido, el pilar de la tecnología, donde ni CMMI ni las metodologías ágiles destacan en su definición o tratamiento.

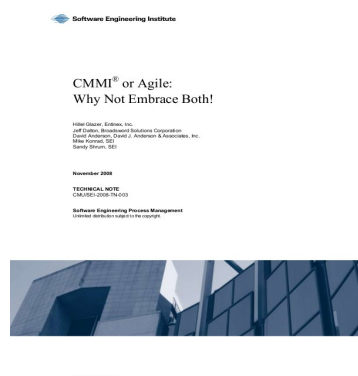


Gráfico 3: Portada del libro «CMMI or Agile: Why Not Embrace Both!»

Redmetrics surge, motivado por una carencia, como una herramienta tecnológica que pretende implementar un modelo de planificación y seguimientos que facilite su uso en las metodologías ágiles y que sea acorde a las directrices definidas por CMMI, concretamente las relativas a las dos siguientes Áreas de Proceso:

- *Project Planning Process Area*
- *Project Monitoring & Control Area*

Ambas Áreas pertenecen a la categoría de procesos *Project Management Category* (fuente: última versión del modelo CMMI-DEV³).

Redmetrics ha tenido en cuenta los principios de la metodología ágil con su planteamiento del dinamismo en la gestión de proyectos, permitiendo la generación de tantas planificaciones como se estimen oportunas.

En definitiva el origen del proyecto Redmetrics es la detección de una brecha tecnológica entre los procesos y las personas. Surge como respuesta tecnológica a una serie de actividades específicas de unos procesos pero teniendo en cuenta las necesidades de agilidad que requieren las personas.

² <http://www.sei.cmu.edu/library/abstracts/reports/08tn003.cfm>

³ <http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm>





3.- METODOLOGÍA DE DESARROLLO

La metodología de trabajo llevada a cabo en el proyecto se ha caracterizado por dos puntos fuertes: el **desarrollo en abierto** y una cuidadosa **gestión de la configuración** de todos los elementos del proyecto.



Gráfico 4: Sistema de control de versiones utilizado en Redmetrics

La utilización de un único repositorio centralizado⁴ (en este caso, subversion) ha facilitado la labor de creación y evolución de los distintos elementos que componen el proyecto.

Dado que el proyecto se ha compuesto por una gran parte de estudio de viabilidad, análisis y diseño, se ha tenido especial atención en el almacenamiento y versionado de toda la documentación generada (en el repositorio, toda la documentación se puede encontrar bajo el directorio doc/).

De la misma forma, se ha utilizado para el código un paradigma de especial cuidado a los distintos componentes de software. En esta línea, se ha trabajado mediante la utilización de una rama de desarrollo (trunk/) y otra rama para almacenar las congelaciones de la plataforma (tags/); de esta manera, toda evolución de código se ha realizado desde la rama de desarrollo y, cuando el desarrollo era lo suficientemente estable, se ha creado una nueva rama congelada para reflejar la liberación de una nueva versión.

Por otra parte, el desarrollo en abierto del proyecto puede dar lugar a que se encuentren fallos en el proyecto por parte de terceros. Este supuesto en el caso de este proyecto es algo totalmente positivo dado que contribuiría a mejorar la calidad del software. De igual manera, se podrían, incluso, recibir parches de código que pudieran incorporarse a la línea de desarrollo directamente, tras la revisión oportuna.

Siguiendo el modelo opensource, el desarrollo en abierto es una apuesta del proyecto Redmetrics para la mejorar su calidad y continuidad.

⁴ <http://redmetrics.googlecode.com/svn/>





4.- INFRAESTRUCTURA Y HERRAMIENTAS

A lo largo del desarrollo del proyecto, se ha utilizado distintos servicios y herramientas. A continuación se muestran una relación de los elementos más relevantes utilizados en el proceso de elaboración de Redmetrics.

En primer lugar, se muestran los servicios utilizados en el proyecto:

Servicio	Dirección	Descripción
 Google Project Hosting	http://code.google.com/p/redmetrics/	Página del proyecto para alojar y centralizar el desarrollo.
 Subversion	http://redmetrics.googlecode.com/svn/trunk/	Sistema de control de versiones.
 Campus UOC. «TFC-Ing. Software aula 1»	http://www.uoc.edu	Plataforma de la UOC utilizada para la coordinación con el tutor

Tabla 2: Infraestructura del proyecto

Por otra parte, a continuación se listan las herramientas más destacables del proyecto:





Herramienta	Versión	Descripción
 Redmine	1.1.3	Plataforma de gestión de proyectos base sobre la que se sostiene Redmetrics.
 Ruby on Rails	2.3.5	Plataforma de desarrollo de aplicaciones web sobre las que se basa Redmine y Redmetrics.
 Ruby	1.8.7	Lenguaje de programación utilizado.
 MySQL Server	5.1.49	Servidor de base de datos utilizado para implementar el modelo de la aplicación de Redmetrics.
 Vi IMproved	7.2	Editor de código utilizado.










 Ubuntu	10.10	Sistema operativo utilizado para el desarrollo. Tanto para la elaboración del código y documentación del proyecto como la puesta en marcha del servidor de pruebas.
 OpenOffice	3.2.1	Herramienta ofimática utilizada para la creación de documentos y diagramas.
 Chromium	11.0.696	Navegador utilizado para las pruebas del prototipo.
 Firefox	4	Navegador utilizado para las pruebas del prototipo.
 TaskJuggler	2.4.3	Herramienta libre de creación de Diagramas de Gantt utilizada en el proyecto.

Tabla 3: Herramientas del proyecto

Nótese que esta relación es simplemente una muestra representativa que permite reflejar el entorno sobre el que se ha desarrollado el proyecto. La relación exacta de aplicaciones, librerías y herramientas se encuentran en el laboratorio preparado para el desarrollo del proyecto.

En cualquier caso, dado que el proyecto se ha desarrollado cumpliendo los estándares que fijan





las plataformas usadas, es tolerante a cambios en el entorno (otros navegadores, nuevas versiones de ruby, ruby on rails, redmine, ...). De hecho, durante la elaboración de esta documentación, se ha publicado la nueva versión de Redmine (1.2) y las primeras pruebas apuntan a que Redmetrics es totalmente compatible con la nueva versión.



5.- CICLO DE VIDA DE REDMETRICS

En este apartado, se expondrán las distintas actividades llevadas a cabo en el proyecto incluyendo los resultados obtenidos en cada fase del ciclo de vida.

5.1.- PLANIFICACIÓN

El proyecto comenzó con la fase de planificación, en dicho periodo comenzaron a esbozarse las directrices que darían forma a Redmetrics.

Todos los elementos trabajados en esta fase quedaron reflejados en el documento «Plan de Trabajo del Proyecto Redmetrics».

PLAN DE TRABAJO DEL PROYECTO REDMETRICS

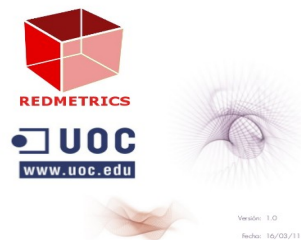


Gráfico 5: Portada del documento «Plan de Trabajo del Proyecto Redmetrics»

5.1.1.- DEFINICIÓN DE OBJETIVOS

Tras evaluar distintos objetivos y contrastar con el tutor la mejor opción, se decide definir un objetivo global que se describía en el plan de proyecto como:

«se pretende realizar un proyecto que aporte al software libre una mejora significativa en la gestión de los procesos comunes en la elaboración de desarrollos de proyectos de software »

Una vez definido este objetivo global, se pudo comenzar a estudiar el estado del arte del software libre para así poder dar una respuesta al alcance del proyecto.

5.1.2.- ANÁLISIS DEL ESTADO DEL ARTE

Dado que el foco de atención eran las herramientas de software libre orientadas a la gestión de proyectos que permitieran un control sobre el desarrollo de software, el campo de atención del estudio estaba muy acotado.

El análisis se centró en la búsqueda en forjas de reconocido prestigio como son:

- <http://sourceforge.net/>
- <http://freshmeat.net/>





- <https://launchpad.net/>

En todos los casos, el conjunto de estudio era muy pequeño, predominaban proyectos sin apenas actividad y estaban muy centrados en dos tipos de gestiones: tareas y diagramas de Gantt.

Es decir, el estado del arte revelaba que existían proyectos de propósito general en lo que a gestión de proyectos se refiere pero poco centrados en la gestión del desarrollo de software.

En esta línea, existen varias opciones de plataformas de propósito general para la gestión de proyectos, una opción emergente es Redmine, una aplicación web basada en Ruby On Rails y fácilmente extensible.

Por tanto, el análisis del estado del arte concluye con la poca profundidad de las aplicaciones de Software Libre en materia de gestión de proyectos de desarrollo de software pero existe una gran base de opciones de propósito general.



Gráfico 6: Redmine, una plataforma de propósito general para la gestión de proyectos

5.1.3.- DEFINICIÓN DEL ALCANCE

La primera decisión para acotar el alcance fue la utilización de Redmine como base del proyecto. Gracias a las funcionalidades que aporta Redmine y su sistema de plugins la elaboración de un proyecto más complejo centraría sus esfuerzos en los elementos que de verdad aportan valor al objetivo de Redmetrics.

En cuanto a la particularidad del sistema de gestión de proyectos de software, se concluyó que debería centrarse en dos tareas claves, la planificación de proyectos y el seguimiento de los mismos. Concretamente, alrededor de tres dimensiones bien diferenciadas:

1. Esfuerzos: dado que el desarrollo de software, principalmente, se traduce en horas invertidas en la elaboración del proyecto, es un bloque de información de obligado análisis.
2. Costes: podemos dividir los gastos de un proyecto en dos tipos: materiales y humanos. Ambos son claves para tener en cuenta el coste y, por tanto, la rentabilidad que pudiera llegar a tener un proyecto.





3. Tiempo: dada la gran casuística que puede llegar a existir en un proyecto de software, la gestión del tiempo con unas métricas adecuadas pueden ser muy útiles para observar y reaccionar adecuadamente.

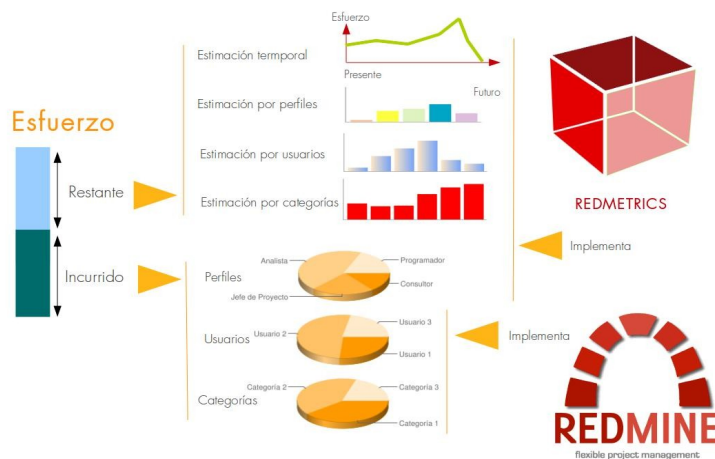


Gráfico 7: Primeros esquemas con el alcance de Redmetrics

Desde la perspectiva temporal, también se definió que estos elementos siempre deberían diferenciar dos perspectivas: la parte incurrida y la parte restante. De esta forma, cada vez que un gestor del proyecto realizara un análisis debería poder acceder a un histórico incurrido en esfuerzos, costes y tiempos del proyecto para, de este modo, tener la mejor información del estado del proyecto y poder plantear así nuevas planificaciones si fueran necesarias.

5.1.4.- ELABORACIÓN DEL PLAN

El trabajo de estudio y definición del alcance culminó en la elaboración de un Plan de Trabajo que, además de incluir todos los elementos anteriormente citados, aportaba una distribución temporal para cumplir los objetivos marcados.



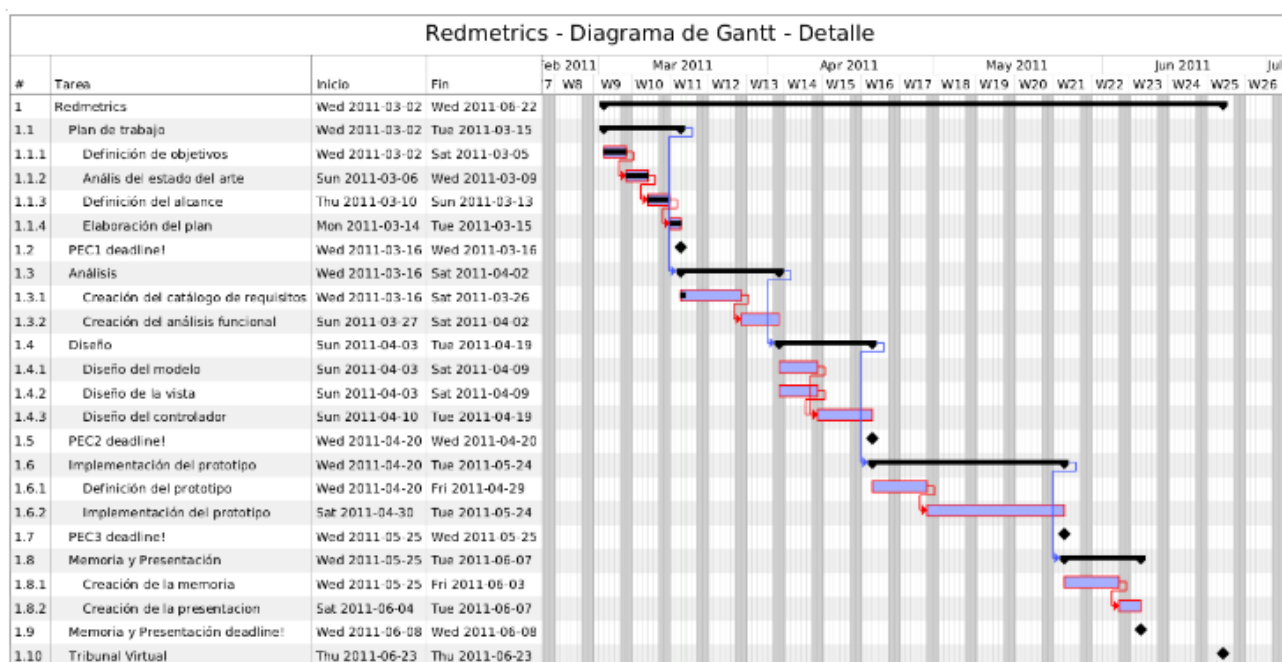


Gráfico 8: Diagrama de Gantt incluido en el Plan de Proyecto

5.2.- ANÁLISIS

El análisis del proyecto comenzó, una vez definidos los objetivos y el alcance general del proyecto, con la elaboración del catálogo de requisitos para posteriormente realizar un análisis funcional del proyecto.

Los elementos generados, tras ser entregados en su primera versión, fueron revisados y actualizados incluyendo las directrices recomendadas por el tutor del proyecto.

5.2.1.- CREACIÓN DEL CATÁLOGO DE REQUISITOS

Este documento comenzó con el desglose de objetivos del sistema, donde se sintetizaron en 9 objetivos los relevantes en el contexto del proyecto.

La elaboración del catálogo de requisitos comenzó la parte de requisitos funcionales con la definición de los actores del proyecto que, en este caso, no existe complejidad al ser únicamente 2: el responsable del proyecto y el participante del proyecto.

A continuación se definieron los diagramas de casos de uso donde se elaboraron los 5 diagramas que representaban los flujos clave del sistema propuesto.



Para concluir con los requisitos funcionales, se detallaron 18 casos de uso del sistema donde se describen detalladamente la denominación, actores, objetivos asociados, requisitos asociados, la descripción, precondition, postcondición, la secuencia normal y las excepciones que pudieran conllevar.

En cuanto a los requisitos no funcionales, se elaboraron un total de 12 requisitos que pretenden reflejar las necesidades horizontales que deben tener un proyecto informático de estas características (seguridad, modularidad, interactividad, eficacia, ...)

El catálogo de requisitos concluye con una matriz de trazabilidad, donde se pueden observar las relaciones entre los requisitos funcionales y los objetivos definidos en el sistema. Tras la elaboración de esta matriz y tras comprobar que todos los objetivos estaban cubiertos con uno o varios requisitos funcionales, se concluyó que el proyecto tenía un catálogo de requisitos acorde a las necesidades del proyecto.

CATÁLOGO DE REQUISITOS DEL PROYECTO
REDMETRICS

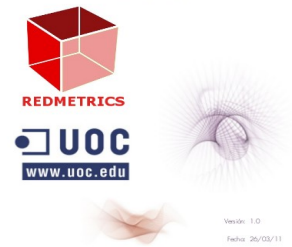


Gráfico 9: Portada del documento «Catálogo de Requisitos del Proyecto Redmetrics»



	OBJ-001. Permitir visualizar los esfuerzos invertidos por los participantes del proyecto.	OBJ-002. Gestionar las previsiones de esfuerzo restante para la finalización de proyecto.	OBJ-003. Permitir visualizar los gastos materiales y humanos realizados a lo largo del proyecto.	OBJ-004. Gestionar las previsiones de gastos restantes para terminar el proyecto.	OBJ-005. Permitir visualizar el tiempo invertido en la realización del proyecto.	OBJ-006. Gestionar las previsiones de tiempo restante para la finalización del proyecto.	OBJ-007. Administrar perfiles.	OBJ-008. Acceso controlado a la plataforma.	OBJ-009. Estudiar desviaciones entre el inicio y el estado actual.
RF-001. Acceder a la plataforma.								●	
RF-002. Salir de la plataforma.								●	
RF-003. Registrar esfuerzos.	●								
RF-004. Visualizar esfuerzos incurridos.	●								
RF-005. Registrar estimación inicial de esfuerzos.		●							●
RF-006. Registrar estimación actual de esfuerzos.		●							●
RF-007. Visualizar esfuerzos incurridos/restantes.	●	●							●
RF-008. Registrar gastos.			●						
RF-009. Visualizar gastos incurridos.			●						
RF-010. Registrar estimación inicial de gastos.				●					●
RF-011. Registrar estimación actual de gastos.				●					●
RF-012. Visualizar gastos incurridos/restantes.			●	●					●
RF-013. Registrar tiempo.					●				
RF-014. Visualizar tiempo incurrido.					●				
RF-015. Registrar estimación inicial de tiempo.						●			●
RF-016. Registrar estimación actual de tiempo.						●			●
RF-017. Visualizar tiempos incurridos/restantes.					●	●			●
RF-018. Administrar histórico de perfiles.							●		

Gráfico 10: Matriz de trazabilidad del proyecto Redmetrics



5.2.2.- CREACIÓN DEL ANÁLISIS FUNCIONAL

Una vez concluido el catálogo de requisitos, se comenzó el desarrollo del análisis funcional del proyecto. La finalidad de esta fase era la obtención de un desglose pormenorizado del sistema entrando especialmente en el análisis GAP, dado que existe una fuerte relación con el software base, Redmine.

Uno de los componentes más relevantes que incluye este análisis es la arquitectura de información del sistema, donde se analizan las 10 unidades de información básicas que se manejarán desde Redmetrics y se expone un diagrama para facilitar la comprensión de la arquitectura.

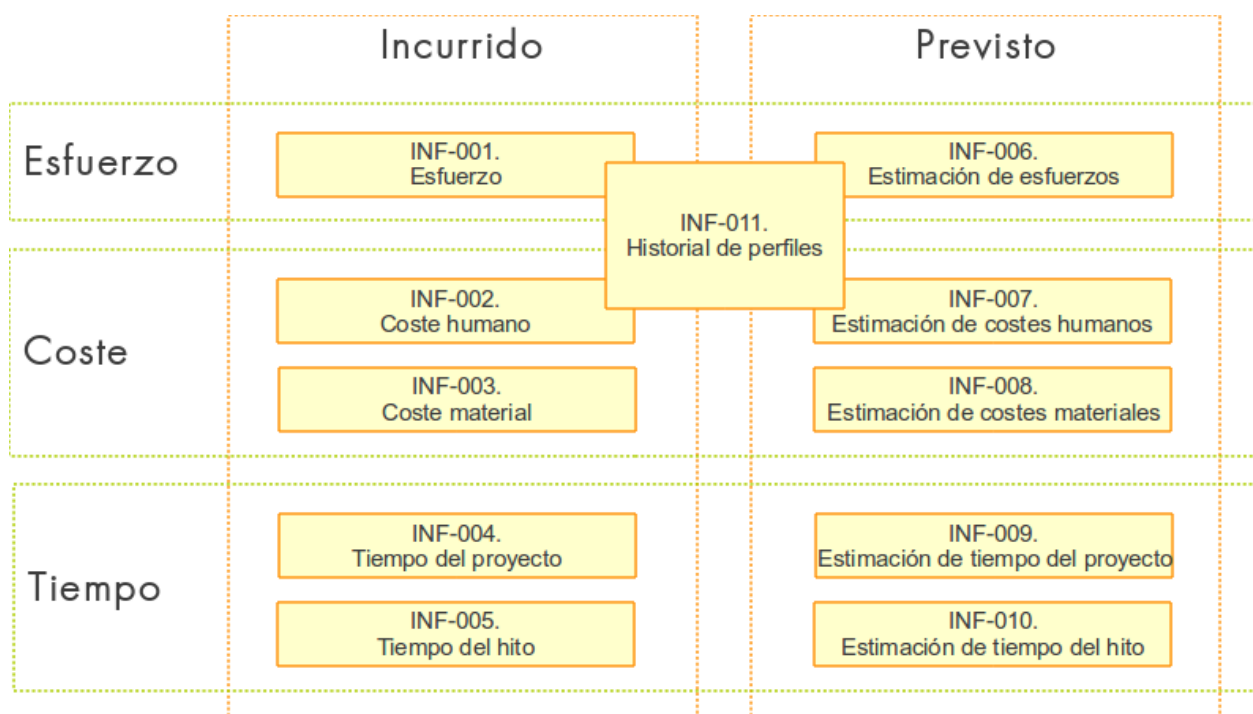


Gráfico 11: Arquitectura de Información del proyecto Redmetrics

Dado que el origen del sistema es acercar el mundo de los procesos a las personas, en el análisis funcional se incluyó un apartado específico para la usabilidad y accesibilidad de Redmetrics, en dicho desarrollo, se definieron las pautas que se utilizarían en esta materia en el resto del proyecto.

Con el objetivo de estructurar el desarrollo, se realizó una descomposición del sistema en partes, donde se concluyó que el sistema debería tener un módulo de datos, un módulo de estimaciones y un tercer módulo de perfiles.



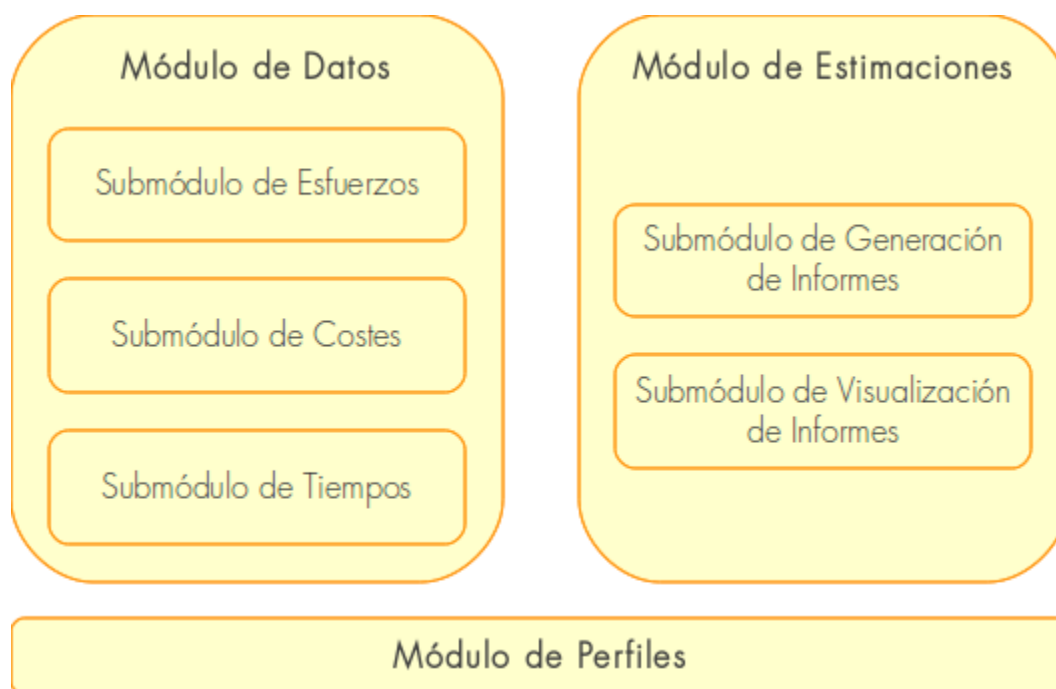


Gráfico 12: Desglose de componentes del proyecto Redmetrics

Por último, el análisis más importante se centró en estudiar las funcionalidades de Redmine y las competencias de Redmetrics. Este análisis detallaba las necesidades del proyecto, la funcionalidad ofrecida por Redmine y las carencias detectadas en la plataforma base. De esta forma, se definió el alcance específico, funcionalmente hablando, de los módulos que debían desarrollarse desde cero o modificarse, al ya existir en Redmine, para adaptarse a las necesidades de Redmetrics.

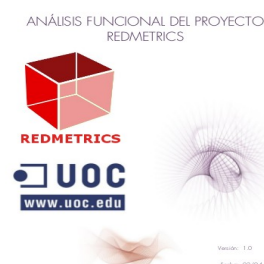


Gráfico 13: Portada del documento «Análisis Funcional del Proyecto Redmetrics»

5.3.- DISEÑO

Esta fase comienza tras la elaboración del Análisis Funcional y tiene como objetivo dar un diseño detallado del modelo, vista y controlador de Redmetrics.

Al igual que los elementos generados en análisis, el documento de diseño, tras ser entregado en su primera versión, fue revisado y actualizado incluyendo las directrices recomendadas por el tutor del proyecto.

5.3.1.- DISEÑO DEL MODELO

Gracias al análisis de la arquitectura de la información realizado previamente, se realizó un modelo completo, en su tercera forma normal, de las necesidades de Redmetrics. Tras ello, se estudio el modelo de datos que aporta Redmine y se creo un segundo modelo, denominado «Modelo segmentado», para mostrar los elementos que ya aportaba Redmine en su modelo o, dicho de otro modo, los elementos que debería incluir el propio modelo de Redmetrics.

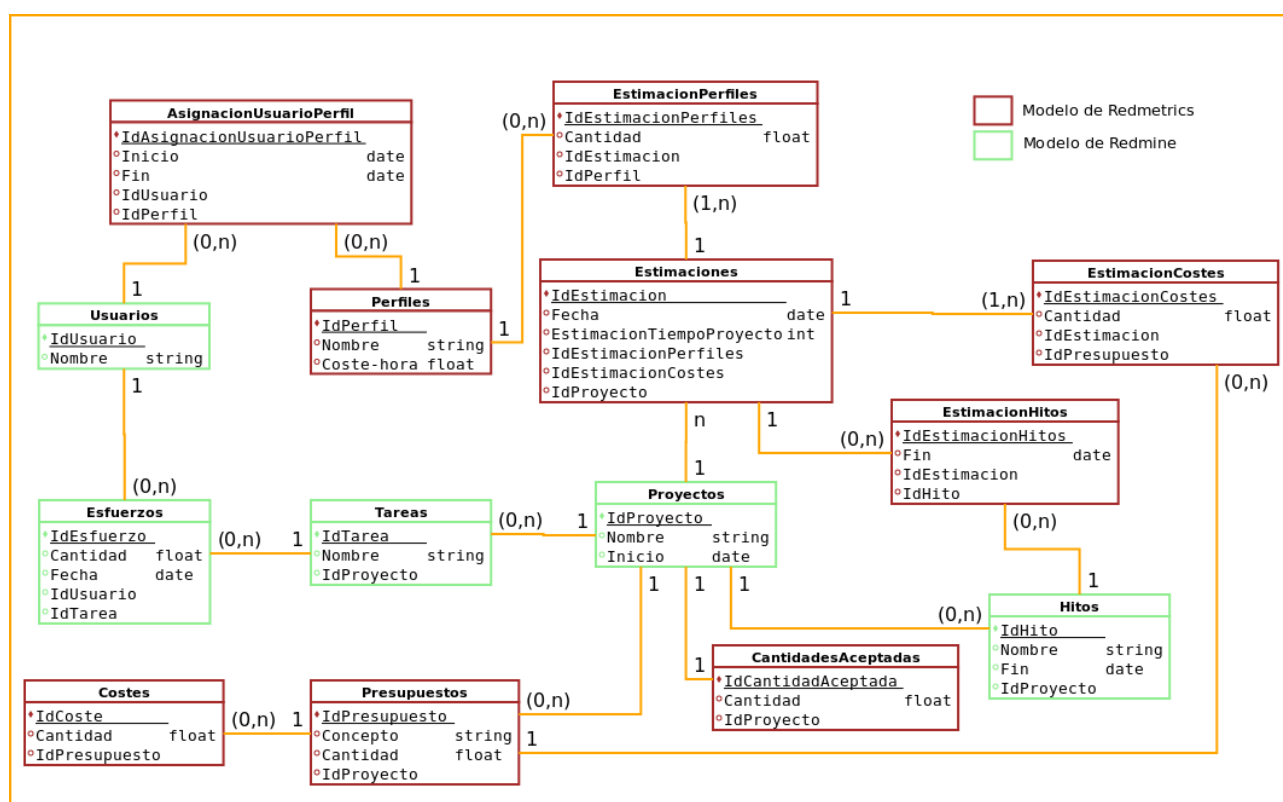


Gráfico 14: Modelo de Entidad-Relación de Redmetrics

Para garantizar que las tablas (concretamente las referentes a usuarios, esfuerzos, tareas, proyectos e hitos) se podían reutilizar se realizó un análisis de equivalencias con el modelo, estudiando atributo por atributo y contrastando que eran homólogos a los campos que Redmetrics necesitaba.

Por tanto, en esta fase, se definió y el modelo y se corroboró la compatibilidad con el modelo de Redmine.



5.3.2.- DISEÑO DE LA VISTA

En el diseño de la vista, se definieron las pantallas más relevantes que se iban a reutilizar de Redmine y, se desarrollaron unos prototipos de 6 pantallas necesarias para Redmetrics.

El prototipado de estas pantallas específicas de Redmetrics incluyen formularios, presentación de tablas, textos de ayuda y distintos elementos para facilitar la interacción de un responsable de proyectos con la herramienta de gestión de proyectos.

Estimación actual

Fecha

Esfuerzos

☒ Perspectiva Perfiles
☐ Perspectiva Usuarios

Horas restantes	2011-04	2011-05	2011-06	2011-07	2011-08	2011-09	2011-10	2011-11	2011-12	2012-01	2012-02	2012-03	V	Total
Jefe de Proyecto														0 horas
Analista Programador														0 horas
Programador Senior														0 horas

Costes

Gastos Previstos

Partidas	Presupuesto
Partida 1	100€
Partida 2	200€
Partida 3	300€
Total	600€

[Modificar](#) | [Eliminar](#)
[Modificar](#) | [Eliminar](#)
[Modificar](#) | [Eliminar](#)

[Nueva partida](#)

Tiempos

Fecha prevista para el cierre del proyecto

Hitos Previstos

Hitos	Fecha de Fin
Hito 1	31/07/2011
Hito 2	15/09/2011

[Modificar](#) | [Eliminar](#)
[Modificar](#) | [Eliminar](#)

[Nuevo hito](#)

Gráfico 15: Prototipo de la pantalla «Estimación actual» definida en el diseño de la vista del proyecto Redmetrics





Costes

Humanos		Estimación 20/04/2011	Actual
Incurrido	Jefe de Proyecto	400€	500€
	Analista Programador	600€	800€
	Programador Senior	1.200€	1.500€
	Proporciones		
Restante	Jefe de Proyecto	1.400€	1.300€
	Analista Programador	2.500€	2.300€
	Programador Senior	400€	100€
	Proporciones		
Previsto	Jefe de Proyecto	1.800€	1.800€
	Analista Programador	3.100€	3.100€
	Programador Senior	1.600€	1.600€
	Proporciones		
	Total		

Materiales		Estimación 20/04/2011	Actual
Incurrido	Presupuesto 1	40€	50€
	Presupuesto 2	60€	80€
	Presupuesto 3	120€	150€
	Proporciones		
Restante	Presupuesto 1	140€	130€
	Presupuesto 2	250€	230€
	Presupuesto 3	40€	10€
	Proporciones		
Previsto	Presupuesto 1	180€	180€
	Presupuesto 2	310€	310€
	Presupuesto 3	160€	160€
	Proporciones		
	Total		

Gráfico 16: Prototipo de la vista «Estado de proyecto. Costes» definida en el diseño de la vista del proyecto Redmetrics



5.3.3.- DISEÑO DEL CONTROLADOR



Gráfico 17: Portada del documento «Diseño del Proyecto Redmetrics»

Por último, la fase del diseño se concluyó con el diseño del controlador. Esta tercera parte es el nexo de unión entre la capa del modelo de datos y la de presentación, donde se almacena toda la lógica del negocio de la aplicación.

Se definieron en total, entre 6 pantallas, 16 funciones que requerían estar alojadas en la capa del controlador. Estas funciones describían las necesidades específicas de Redmetrics.

Este diseño, al igual que la parte del modelo y de la vista, quedaron recogidos en el documento «Diseño del Proyecto Redmetrics».

5.4.- IMPLEMENTACIÓN DEL PROTOTIPO

Esta última fase del proyecto tenía como objetivo la elaboración de un prototipo que garantizara la viabilidad del sistema. Dado que el proyecto realiza un análisis y diseño de un proyecto de ciertas dimensiones, esta fase se limitaba a escoger un subconjunto de funcionalidades que pudieran ser probadas en un prototipo. Para ello, se realizó primero un esfuerzo en la definición del prototipo para, posteriormente, realizar su implementación.

5.4.1.- DEFINICIÓN DEL PROTOTIPO

La definición del prototipo tuvo en cuenta que el tiempo para la implementación era de 3 semanas desde la finalización de las especificaciones, por ello se centró en escoger las funcionalidades que menor complejidad entrañaban en su elaboración.

Para facilitar la comprensión de las especificaciones, se elaboró una relación de los casos de uso previstos del sistema con su nivel de implementación en el sistema, dando lugar a 7 casos de uso completamente implementados 11 casos de uso parcialmente implementados.

En este contexto, se entendió clave la reutilización de todo el código de Redmine, así como de otros plugins, que pudieran ayudar a facilitar la implementación del sistema definido. En esta línea, la arquitectura de plugins de Redmine y la comunidad que hay detrás de la misma ayudó en gran medida.





5.4.2.- IMPLEMENTACIÓN DEL PROTOTIPO

El prototipo se llevó a cabo en el tiempo estipulado y generó un conjunto de funcionalidades suficientes como para probar la viabilidad del sistema.

Para facilitar la comprensión de la implementación del prototipo implementada, se ha creado un vídeo en la plataforma Youtube⁵. En dicho vídeo se podrán observar varios casos de uso del sistema incluyendo la presentación de pantallas de informes, uno de los elementos más útiles para un usuario de Redmetrics.

Redmetrics - Demo

c4rlospc

7 videos

Subscribe

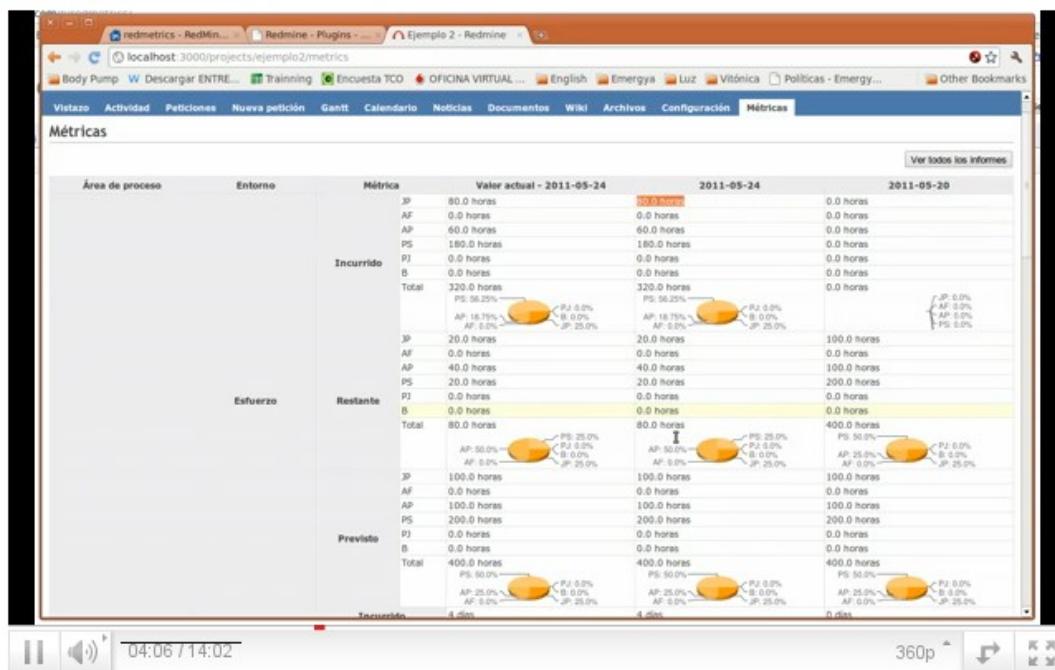


Gráfico 18: Vídeo de demostración de Redmetrics que puede encontrarse en <http://www.youtube.com/watch?v=btpNPCtCV3c>

Todo el código implementado siguió, tal y como se explicaba en este mismo documento en las metodologías de trabajo, unas guías de referencia que han facilitado un desarrollo estructurado de tres semanas, dando lugar a dos versiones, 0.1 y 0.2.

En cuanto al presente Trabajo Fin de Carrera, Redmetrics concluye su primera etapa con un primer esbozo de funcionalidades implementadas y con las garantías de que el análisis y diseño del sistema tienen una orientación adecuada.

⁵ <http://www.youtube.com/watch?v=btpNPCtCV3c>



6.- CONCLUSIONES

A continuación se citan las conclusiones del Trabajo Fin de Carrera, Redmetrics:

- Gracias al estudio del arte realizado en la primera etapa del proyecto, podemos concluir que en cuanto a herramientas de gestión de proyectos de software aún hay mucho campo para innovar. Existen multitud de metodologías emergentes que apenas tienen herramientas tecnológicas especializadas.
- El control y seguimiento del proyecto se centra en las dimensiones de esfuerzo, coste y tiempo pero no son las únicas, hay otra dimensión muy relevante que podría incluirse como bloque de control: la calidad. Inicialmente no se contempló como dimensión dentro de Redmetrics por no aumentar la complejidad del alcance pero, tras las primeras pruebas, se concluye que sería muy interesante incluir dicha dimensión. De esta forma, un responsable de proyecto tendría en una pantalla unificada métricas referentes a cuatro puntos claves en el desarrollo de un proyecto de software: esfuerzo, tiempo, coste y calidad.
- En el desarrollo del proyecto se ha tenido especial cuidado en el cumplimiento de los plazos de entrega y el seguimiento del plan establecido. Si bien es cierto que la planificación panorámica entregada se ha cumplido correctamente en lo que a plazos de entrega se refiere, hay ciertas tareas dentro de un mismo bloque (por ejemplo, la creación de los distintos elementos que conformaban el documento de diseño) donde han sido difícilmente respetables las fechas internas. Esto puede significar que el grado de detalle de la planificación era muy excesivo para un proyecto de una única persona o que no se definieron bien las interrelaciones entre los distintos componentes.
- El proceso de elaboración del catálogo de requisitos y la elaboración del análisis funcional son vitales para la realización de un proyecto de ingeniería informática. Sin dichos trabajos previos, el grado de comprensión del sistema y la coherencia de los módulos no tendrían un nivel suficiente. En este caso, se ha utilizado constantemente la capacidad de abstracción y de construcción de arriba-abajo para detallar planes cada vez más definidos (Plan de Trabajo → Catálogo de Requisitos → Análisis Funcional).
- A lo largo del diseño, se ha profundizado en mayor medida en la tecnología. Se concluye que un buen dominio en la tecnología que se vaya a desarrollar facilita la realización de un diseño de calidad. La utilización del paradigma Modelo/Vista/Controlador ha facilitado en gran medida la definición por bloques de las necesidades de Redmetrics; es altamente recomendable seguir este paradigma.
- El prototipo implementado ha cumplido su objetivo de probar la viabilidad del sistema gracias a la simplificación del subconjunto de funcionalidades que debía cumplir. En





proyectos de cierta dimensión, es una buena estrategia que debe seguirse y, de esta forma, permitiría visualizar a los usuarios un primer esbozo de lo que el sistema pretende ser. De esta manera, las reacciones, tanto positivas como negativas, aparecerían antes y el desarrollo podría reorientarse en una etapa más temprana del proyecto.

- La reutilización de componentes de software es un gran acierto siempre y cuando se realice un análisis detallado de las funcionalidades que aporta y las que quedan por implementar por parte del nuevo sistema que se pretende crear. Gracias a la reutilización de Redmine, se ha podido crear un sistema en forma de plugin que no podría haberse realizado de otro modo. Es decir, la reutilización permite construir aplicaciones más complejas y más específicas de una forma más económica en cuanto a esfuerzo y tiempo invertido se refiere.
- La utilización de un repositorio de control de versiones ha facilitado en todo momento el control de la documentación y el código. Sin dicho repositorio o el histórico que contiene el mismo, el desarrollo habría sido mucho más artesanal, sin posibilidad de volver a versiones anteriores o sin la confianza que aporta un sistema de copias de seguridad.
- La infraestructura de Google Project Hosting ha aportado mucha ayuda en la estructura del proyecto. Se concluye que la utilización de una forja especializada en el desarrollo de software puede aportar en el planteamiento del proyecto así como en la difusión a terceros.
- El Software Libre aporta un ecosistema de trabajo y de desarrollo de software de alta calidad. Todo el proyecto ha sido realizado utilizando Software Libre y, concretamente, sin coste alguno en licencias siendo totalmente legal. El nivel de depuración de las herramientas utilizadas era muy alto, siempre versiones estables y, por tanto, han facilitado que el desarrollo de Redmetrics haya sido una realidad. El Software Libre es una alternativa viable.
- La usabilidad del proyecto es muy baja en el prototipo, no se dedicó suficiente esfuerzo en la formación en tecnologías que mejoren la capa de presentación (como pueden ser AJAX) y, en consecuencia, los formularios aún son muy simples. De hecho, se concluye que es una de las principales prioridades en las sucesivas evoluciones de Redmetrics.





7.- BIBLIOGRAFIA

A continuación se cita la bibliografía utilizada:

Título	Autor	Dirección
Ruby Style Guidelines	Google	http://www.caliban.org/ruby/rubyguide.shtml#style
CMMI or Agile: Why Not Embrace Both!	Software Engineering Institute of Carnegie Mellon	http://www.sei.cmu.edu/library/abstracts/reports/O8tn003.cfm
CMMI for Development, Version 1.3	Software Engineering Institute of Carnegie Mellon	http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm
CMMI for Development: Guidelines for Process Integration and Product Improvement, Third Edition	Software Engineering Institute of Carnegie Mellon	http://www.sei.cmu.edu/library/abstracts/books/0321711505.cfm
Guía de Desarrollo de Redmine	Redmine Community	http://www.redmine.org/projects/redmine/wiki/Guide
Materiales y Fuentes del TFC – Ingeniería del Software	Antoni Pérez Navarro, Alfons Bataller Díaz, Roser Beneito Montagut, Nita Sáenz Higuera, Rut Vidal Oltra.	http://www.uoc.edu

Tabla 4: Relación bibliográfica





ANEXO I: RELACIÓN DE GRÁFICAS

ÍNDICE DE GRÁFICAS

Gráfico 1: Flujo de actividades.....	4
Gráfico 2: El logo elegido para el proyecto es un cubo en perspectiva; representa la perfección de las métricas y las distintas perspectivas de una misma verdad.....	5
Gráfico 3: Portada del libro «CMMI or Agile: Why Not Embrace Both!».....	7
Gráfico 4: Sistema de control de versiones utilizado en Redmetrics.....	8
Gráfico 5: Portada del documento «Plan de Trabajo del Proyecto Redmetrics».....	13
Gráfico 6: Redmine, una plataforma de propósito general para la gestión de proyectos.....	14
Gráfico 7: Primeros esquemas con el alcance de Redmetrics.....	15
Gráfico 8: Diagrama de Gantt incluido en el Plan de Proyecto.....	16
Gráfico 9: Portada del documento «Catálogo de Requisitos del Proyecto Redmetrics».....	17
Gráfico 10: Matriz de trazabilidad del proyecto Redmetrics.....	18
Gráfico 11: Arquitectura de Información del proyecto Redmetrics.....	19
Gráfico 12: Desglose de componentes del proyecto Redmetrics.....	20
Gráfico 13: Portada del documento «Análisis Funcional del Proyecto Redmetrics».....	20
Gráfico 14: Modelo de Entidad-Relación de Redmetrics.....	21
Gráfico 15: Prototipo de la pantalla «Estimación actual» definida en el diseño de la vista del proyecto Redmetrics.....	22
Gráfico 16: Prototipo de la vista «Estado de proyecto. Costes» definida en el diseño de la vista del proyecto Redmetrics.....	23
Gráfico 17: Portada del documento «Diseño del Proyecto Redmetrics».....	24
Gráfico 18: Vídeo de demostración de Redmetrics que puede encontrarse en http://www.youtube.com/watch?v=btpNPCtCV3c	25





ANEXO II: RELACIÓN DE TABLAS

ÍNDICE DE TABLAS

Tabla 1: Ficha resumen del proyecto.....	5
Tabla 2: Infraestructura del proyecto.....	9
Tabla 3: Herramientas del proyecto.....	11
Tabla 4: Relación bibliográfica.....	28

