

Consejos para el desarrollo de tu proyecto final de carrera

Francisco J. Abad Cerdá (fjabad@dsic.upv.es)

8 de octubre de 2010

Prólogo

Seguramente estás leyendo este documento porque vas a empezar a hacer tu proyecto final de carrera, y de alguna forma me has convencido para que te lo supervise. Pues bien, aquí te presento una serie de consejos y guías para hacerte la vida más fácil durante los próximos meses.

En primer lugar, aunque tengas muchas ganas de empezar a escribir código, a construir un prototipo, o a hacer lo que piensas que hay que hacer para completar tu proyecto, pisa el freno. Piensa que, independientemente del tema que hayas escogido, hay muchas personas muy inteligentes que ya han dedicado mucho tiempo a trabajar en dicho tema. No te tomes esto que te acabo de decir como que “ya está todo hecho”, sino como “cómo puedo mejorar lo que ya existe”. Obviamente “mejorar” es un concepto difuso: puede significar hacer que algo funcione más rápido, que funcione de un modo no convencional o novedoso, que sea más barato que el sistema original, etc. Por ello, el primer paso de tu proyecto final de carrera será ponerse al día (de hecho, este paso será común a cualquier proyecto que te propongas en tu vida profesional).

En terminología académica, ponerse al día se llama “estudiar el estado del arte”. Haberse leído el artículo correspondiente de la Wikipedia no cuenta, por lo que te voy a pedir que te leas completamente dos libros, a tu elección:

- Un libro sobre la herramienta que vayas a usar para desarrollar tu proyecto (ya sea sobre programación web, C++, Matlab, Java...).
- Un libro sobre la temática específica de tu proyecto (visión por ordenador, grafos de escena, videojuegos, control de procesos...).

Sobre qué libro en concreto elegir, ese será tu primer trabajo: busca qué opciones tienes disponible, y cuál es el más adecuado para tí, basándote en: fecha de publicación, popularidad (cuántas veces aparece en Google), si tiene una web asociada con correcciones o material adicional, críticas en Amazon, etc. La biblioteca de la UPV está surtida de toda clase de libros. Además, si no encuentras el que te interesa, puedes pedirles que lo compren (desde tu Intranet).

Obviamente, aparte de esos libros, tendrás que dedicar algún tiempo a buscar información en Internet. Busca tanto conocimiento nuevo que haya aparecido desde la edición del libro, hasta programas, bibliotecas, etc. en los que te puedas basar para construir tu solución. Si existe un programa parecido al que vas a desarrollar, estúdialo. Busca sus ventajas (que tu sistema debería mantener) y sus inconvenientes (que tu sistema debería solucionar).

A continuación se describe la estructura básica que debe seguir todo proyecto final de carrera en Informática. Consulta la normativa específica de tu centro con respecto al formato de la memoria, modo de entrega, etc.

Los títulos utilizados en cada apartado no tienen porqué ser los expuestos aquí. Se proponen únicamente como guía de lo que debe aparecer en cada uno. De hecho, son unos títulos horribles. Búscate los tuyos.

No incluyas excesivo código fuente en la memoria. Adjúntalo en un CD aparte. En la memoria puedes incluir trozos de código que te parezca interesante resaltar, pero no metas código simplemente para engordar el número de páginas. Hablando de número de páginas, ¿cuánto tiene que ocupar un PFC? A partir de 100 en total sobra. A partir de 200 es a todas luces excesivo. Piensa en unas 60 páginas de texto donde expliques tu aplicación. Los detalles, a los apéndices.

En el enlace siguiente:

<http://www.di.uniovi.es/~aquilino/ficheros/pfc/NormasPFC-EUITIO.pdf>

tienes un documento que da una serie de consejos útiles para los futuros ingenieros haciendo un proyecto final de carrera. Por favor, léelo tranquilamente.

Siguiendo uno de los puntos del documento anterior, deberás crear una página web (o un blog) donde describas el estado de tu proyecto. Mándame la dirección y trata de poblar la página web con toda la información generada por tu proyecto. **NO** me mandes material por correo electrónico (código fuente, documentos, imágenes, etc), cuélgalo de tu web y mándame el enlace.

Acostúmbrate a escribir todo lo que te parezca interesante en la página web porque seguramente te sirva a la hora de escribir la memoria del proyecto (desde que empiezas a trabajar hasta que tengas que escribir la memoria pasarán varios meses).

En tus comunicaciones conmigo:

- Ten en cuenta que tengo muchas cosas en la cabeza. Identifícate en el correo que mandes y usa el asunto del correo electrónico como resumen de lo que hay en el cuerpo del correo.
- No escribas los correos electrónicos como si fueran SMS. Termina las palabras, usa el corrector ortográfico y no incrementes la frecuencia de aparición de la letra 'k' en el castellano.

Índice general

Prólogo	I
Índice general	III
1. Introducción	1
2. Antecedentes	3
3. Análisis	5
4. Diseño	7
5. Resultados	9
6. Conclusiones	11
A. Manual de usuario	13
B. Presentación	15
Bibliografía	17

Capítulo 1

Introducción

Ten en cuenta quién va a leer la memoria: posiblemente tú, yo, y el tribunal. Además, el tribunal no se la leerá entera. Por ello, hazles un favor: trata con especial cariño la introducción, el análisis y las conclusiones.

- Presentación del problema.
- Objetivos del proyecto. En este punto deberías describir qué problema pretendes resolver con tu aplicación. También deberías establecer unos objetivos medibles (p.e., que la aplicación funcione en tiempo real, o que permita hacer cierto trabajo con un mínimo de interacción con el usuario).
- Descripción de alto nivel de las bondades de la solución propuesta.
- Descripción de la estructura del resto de la memoria del proyecto.

Capítulo 2

Antecedentes

- Descripción detallada de los sistemas existentes más conocidos en el área de la aplicación
- Cada sistema que describas tendrá, al menos, una referencia como esta [1]. En el capítulo de bibliografía, al final, se listarán los detalles de todos los trabajos referenciados (incluyendo las direcciones web en caso de estar *on-line*).
- Además, a lo largo del texto del resto de la memoria, debes incluir las citas a los trabajos en los que te hayas apoyado para resolver cada problema (por ejemplo, si usas un algoritmo publicado, o si usas una función de una biblioteca).
- Dedicar el primer mes del proyecto a buscar información. Es muy importante que, antes de empezar a escribir código, tengas claro qué quieres hacer y cómo quieres hacerlo.
- No vuelvas a inventar la rueda. Busca algoritmos, librerías, artículos, libros, etc. que implementen o describan la solución a alguno de los problemas de tu proyecto. Una parte importante del trabajo de cualquier ingeniero es apoyarse en el conocimiento existente para conseguir mejores resultados. De esta forma, las metas de tu proyecto podrán ser más ambiciosas, y podrás desarrollar algo realmente útil.
- Como resumen de este capítulo: lee, lee, lee y lee. Te debes convertir en un experto en el área de aplicación de tu proyecto, y eso sólo se puede hacer estudiando el trabajo de otros expertos.
- *Éste es el primer capítulo que deberías escribir.*

Capítulo 3

Análisis

- Descripción detallada de la solución, paso a paso, sin hacer referencia concreta a la implementación ni a tecnologías específicas.
- Descripción del flujo de trabajo dentro de la aplicación. Deberás describir cómo fluye la información por tu sistema.
- Descripción de los algoritmos a utilizar.
- Aquí deberás tomar decisiones que darán o quitarán flexibilidad a la aplicación (por ejemplo, si tu programa tiene una salida gráfica, especificarás aquí si permites una vista perspectiva, una vista ortográfica, o ambas).
- Aquí establecerás una lista de requisitos que debe cumplir tu aplicación. Básicamente, este capítulo describe el *qué*. El *cómo* se describe en el capítulo siguiente,

Capítulo 4

Diseño

- Descripción de los módulos, clases principales, etc (a ser posible, con esquemas y diagramas utilizando algún lenguaje de descripción conocido, como UML).
- Metodología de desarrollo.
- Aquí deberás explicar en detalle cómo funciona tu aplicación. Recuerda: no incluyas código.
- Una forma de ayudarte a diseñar la aplicación es mediante los Casos de uso. Este método de diseño simula distintas interacciones entre el usuario y los distintos módulos del sistema.
- Descripción de la interfaz de usuario. Normalmente este punto se pasa por alto en el desarrollo de un proyecto final de carrera, pues parece que no tiene tanta importancia como el resto. Deberías prestar especial atención a la interfaz de usuario, especialmente en aplicaciones relacionadas con Gráficos por computador.
- Para estudiar si tu aplicación tiene un buen interfaz de usuario o no, pídele a algún amigo o familiar que la use. Explícale por encima para qué sirve tu aplicación y pídele que haga algo con ella. Observa sus pasos. Fíjate qué hace y dónde pierde más tiempo. Luego, reorganiza los controles o rediseña la interfaz teniendo en cuenta los problemas de usabilidad que hayas encontrado.

Ni se te ocurra introducir en este apartado de la memoria la lista de clases de la aplicación, con sus métodos, variables miembro y detalles varios. Añádelos como un apéndice.

Capítulo 5

Resultados

- Desarrollo de uno o varios ejemplos completos.
- Deberás describir los datos de entrada y los resultados obtenidos.
- También deberás evaluar tu sistema. Un sistema, dependiendo de su objetivo, se puede evaluar en función de: su velocidad (p.e., en *frames* por segundo), memoria que utiliza, usabilidad, etc. Idealmente deberías comprobar que has alcanzado los objetivos descritos en el capítulo 1.

Capítulo 6

Conclusiones

- Descripción del desarrollo del proyecto (duración, problemas encontrados, etc).
- Conclusiones del trabajo realizado.
- Trabajos futuros.

Apéndice A

Manual de usuario

- Descripción de la forma de usar la aplicación.

Apéndice B

Presentación

El día de la presentación seguramente lo pasarás mal. Estarás delante de un tribunal compuesto por tres profesores muy malos a los que no conoces y que te van a hacer preguntas difíciles y que quieren que sufras el máximo posible.

No te preocupes por la presentación. Si has llegado tan lejos, significa que posiblemente tú serás el que más sepa del tema de tu PFC en la sala, por lo que no deberías tener problemas para pasar el mal trago victoriosamente.

El día de la presentación procura estar por lo menos media hora antes en el lugar convenido para hacerte una idea de la sala de la presentación, te asegures de que todo funciona, etc.

Con respecto a la transparencias de la presentación, varios consejos:

1. La primera transparencia mostrará: el título de tu trabajo, tu nombre, el nombre de tu director, la fecha y el centro.
2. Cíñete al tiempo estipulado. Cada transparencia equivale más o menos a un minuto de presentación (por lo tanto, si tienes 20 minutos, prepara 20 transparencias).
3. No intentes meter todo el material del proyecto en la presentación para demostrar todo el trabajo que has hecho. Posiblemente te pasarás de tiempo, o tendrás que hablar tan rápido que nadie se entere de nada.
4. Lo más importante en una presentación es describir adecuadamente el problema, la solución y las conclusiones de tu trabajo. Los detalles están en la memoria del proyecto, y si hay alguien interesado en el tribunal te puede preguntar en el turno de preguntas.
5. Intenta usar muchas imágenes, vídeos, diagramas, etc. y minimiza el uso de texto. No copies párrafos enteros de la memoria a la presentación.

6. Puedes usar alguna animación en la presentación, pero no abuses. Evita hacer filigranas con el Powerpoint. **Evita los colores estridentes.** Usa una de las plantillas existentes y la paleta de colores por defecto (en esto, el Powerpoint 2007 es mucho mejor que los anteriores).
7. Por supuesto, pasa el corrector ortográfico a tu presentación. Las faltas de ortografía quedan muy mal en pantalla grande.

Con respecto a tu defensa ante el tribunal:

1. Prepárate la presentación varias veces. Conmigo harás por lo menos un ensayo, pero suéltale el rollo a algún amigo, familiar, etc.
2. Cuando estés delante del tribunal, no te muevas excesivamente, no bailes, o te muevas de un sitio a otro de la habitación. Tampoco te interpongas entre el tribunal y la pantalla del proyector.
3. Puedes usar algún laser o puntero para llamar la atención del tribunal sobre algún punto específico de la transparencia, pero tampoco quemes la pantalla abusando del laser.
4. Evita el lenguaje coloquial. Aunque estemos entre amigos, una defensa de PFC es algo serio.
5. El tribunal estará hojeando, leyendo y estudiando la memoria mientras que presentas. En esos momentos te arrepentirás de no haber comprobado que el documento que entregaste se imprimió bien. El viejo truco de que el Word se ha cargado tu memoria no funciona. Revisa el formato después de haberla impreso y asegúrate de que es legible, tiene unos márgenes adecuados y se puede leer bien.

Salvando las distancias, te puedes hacer una idea de cómo se hace una buena presentación buscando “apple keynotes” en Google. Mira un par de presentaciones de Steve Jobs y toma nota (no hace falta que te pongas el jersey negro de cuello alto).

Bibliografía

- [1] Sameer Agarwal, Ravi Ramamoorthi, Serge Belongie, and Henrik Wann Jensen. Structured importance sampling of environment maps. *ACM Transactions on Graphics*, 22(3):605–612, July 2003.