

TRABAJO DE CURSO EVOLUCIÓN Y GESTIÓN DE LA CONFIGURACIÓN

GRADO EN INGENIERÍA DEL SOFTWARE ETSI INFORMÁTICA

UNIVERSIDAD DE SEVILLA



6 de octubre de 2016

CONTROL DEL DOCUMENTO

REGISTRO DE CAMBIOS EN EL DOCUMENTO			
VERSIÓN	MOTIVO	REALIZADO POR	FECHA
1.0	Creación	D. Benavides	12 de Septiembre de 2014
1.1	Publicación de borrador	D. Benavides	22 de Septiembre de 2014
2.0	Cambiado nº de componentes	D. Benavides	29 de Septiembre de 2014
3.0	Versión 2015/2016	D. Benavides	Septiembre de 2015
4.0	Versión 2016/2017	D. Benavides	Octubre de 2016

Índice general

1. Introducción	4
1.1. Objetivos generales	4
1.2. Tareas para la realización	4
2. Equipos	6
2.1. Conformación	6
2.2. Número de componentes	6
2.2.1. Coordinador del equipo	6
2.2.2. Reparto de tareas del equipo	7
2.3. Fecha límite de inscripción	7
2.4. Gestión de conflictos	7
3. Temática	9
3.1. Objetivo del trabajo	9
3.2. Elección del proyecto	9
3.2.1. Proyecto real	9
3.2.2. Proyecto de cursos anteriores	9
3.3. Evolución y cambios en el proyecto	10
3.4. Actividad del proyecto	10
4. Entregables	12
4.1. Documento del proyecto	12
4.2. Máquina virtual	15
4.3. Diario del equipo	16
4.4. <i>Milestones</i> intermedios	16
4.5. Fecha para la entrega	16

1 Introducción

1.1. Objetivos generales

El objetivo general del trabajo es que el equipo ponga en práctica y observe cómo se ponen en funcionamiento en un proyecto real todos los conceptos teórico-prácticos que se vean en la asignatura y profundice en ellos **todo lo que su motivación lo lleve**.

Este es un trabajo en el que, para llevarlo a cabo de manera satisfactoria, será muy útil tener buenas dosis de entusiasmo e interés en la temática del trabajo. En este sentido, se da un amplio margen de libertad para elegir la temática, el proyecto, las herramientas, la tecnología,... por lo tanto, **¡elige lo que más te apasione!**.

De los entregables se espera tengan un acabado profesional. La puntualidad, rigor y seriedad en las entregas serán valores irrenunciables sin los que no podrá llevarse el trabajo adelante.

1.2. Tareas para la realización

De manera general las tareas para la realización del trabajo serán los siguientes:

- Conformar e inscribir el equipo (ver Sección 2).
- Elegir el proyecto a realizar (ver Sección 3.2).
- Establecer con claridad la organización y coordinación del equipo y el mecanismo de comunicación tanto interno como externo que se va a tener en el grupo.
- Ir estudiando los aspectos vistos en clases y observando cómo el proyecto elegido trata dichos aspectos y cómo se van a hacer en el proyecto a realizar e ir documentando gradualmente esos aspectos. Ir pensando en alternativas, mejoras, etcétera. No se trata de saber replicar lo que ya está hecho. Se trata de que se puedan buscar mejores soluciones o alternativas.
- Ir elaborando un entorno con las herramientas que se usan o se podrían usar en el

proyecto. Crear una máquina o varias máquinas virtuales para ello.

- planificar las iteraciones tanto internas como los *milestones* externos.
- Elaborar una documentación profesional siguiendo las indicaciones de este documento.
- Elaborar una o varias máquinas virtuales con todo el entorno a entregar.
- Elaborar un diario del grupo.
- Empaquetar todo para la entrega y entregar.

2 Equipos

2.1. Conformación

El trabajo debe ser realizado en equipo. Se hará un taller para ver afinidades de personas y disponibilidades horarias para conformar los equipos un día en clases de teoría al inicio del cuatrimestre. En todo caso este taller no es vinculante y servirá únicamente para promover la creación del equipo. **Es responsabilidad de cada alumno/a buscar un equipo** que le interese y se acerque a sus expectativas e intereses. Sólo en casos extremos, se acudirá al coordinador de la asignatura para que le asigne un equipo a un alumno/a.

Se recomienda que se busquen en el equipo que sean afines en intereses, horarios y forma de trabajo.

2.2. Número de componentes

Habrán un máximo de 10 equipos por cada grupo de clases, es decir, 10 equipos por la mañana, y 10 equipos por la tarde.

Los equipos de trabajo deben estar conformados por **5 personas** salvo circunstancias excepcionales. Si en el transcurso de la asignatura algún miembro del equipo abandona, este abandono debe ser notificado al coordinador de la asignatura y quedar reflejado en el diario del equipo constando la firma y aprobación de los restantes miembros del equipo y si fuera posible de la persona que abandona el equipo. Si después de distribuir a todos/as los alumnos/as se quedase un equipo con un número distinto de 5 personas, la incidencia se resolverá junto con el delegado de la clase y el coordinador de la asignatura.

2.2.1. Coordinador del equipo

Habrán al menos un coordinador del equipo y como máximo dos que serán quienes lleven la coordinación en el equipo en cuanto a tiempos, tareas y gestiones. También

llevará las tareas de coordinación con el resto de equipos. No se debe confundir ser *coordinador* con ser *jefe* ¹.

Los coordinadores de los distintos equipos deben comunicarse entre ellos para aclarar cualquier cuestión de coordinación que sea necesaria entre los equipos.

2.2.2. Reparto de tareas del equipo

Todos los componentes del equipo deben ser desarrolladores en el proyecto. No se puede dividir el equipo para que unos hagan desarrollo y otros hagan documentación por ejemplo. Para poder poner en práctica los conocimientos debe todo el mundo ser desarrollador/a y trabajar en paralelo. Se podrá cargar más en una persona u otra la parte de desarrollo pero en ningún caso se podrá quitar a alguien de esta labor.

2.3. Fecha límite de inscripción

La fecha límite para inscribir los equipos será el **último viernes de Octubre**. Para inscribir los equipos se usará el portal de grupos disponible en: <https://opera.eii.us.es/egc>. En todo caso esa es la fecha límite para inscribir el equipo pero desde antes debe estar conformado para poder ir trabajando en el proyecto.

2.4. Gestión de conflictos

Se tendrá una actitud positiva en el equipo de trabajo pero eso no quitará que en algún momento se pueda dar y seguramente se de algún tipo de conflicto. En caso de conflicto “grave” se acudirá cuánto antes al coordinador de la asignatura para que sepa ayudar a reconducir o resolver el conflicto de modo que no perjudique a ninguna de las partes. Tipos de conflictos frecuentes:

Conflicto: Algún miembro del equipo no trabaja, no acude a las reuniones, no contesta a los mensajes, no presta interés en el equipo, etcétera.

Acción a realizar: Poner un límite prudente a esta situación y no dejar pasar el tiempo. Hacer constar en acta la situación y comunicarlo al coordinador para, en su caso, dar de baja al miembro del equipo o intentar mediar en el conflicto.

¹Ver, por ejemplo, este enlace como aclaración: https://projetsii.informatica.us.es/attachments/download/990/Trabajo_en_equipo_correcciones_y_apuntes_.pdf

Conflicto: Se tienen puntos de vista distintos sobre determinados temas del trabajo.

Acción a realizar: Esto es muy frecuente en un equipo de trabajo por lo que es una situación que deben superar y sólo en caso extremo acudirán al coordinador para que, en su caso, medie en el conflicto.

Conflicto: Hay niveles distintos de interés/habilidades en el equipo y eso hace que haya partes que se hagan mejor y otras peor.

Acción a realizar: El coordinador del equipo debe intentar solventar esta situación y, en todo caso, será algo frecuente en el trabajo por lo que las personas con más interés deben atraer el interés de las personas que tengan menos para así acompañarlas, motivarlas y hacerlas aprender pues enseñando también se aprende. El liderazgo no es solo saber trabajar bien y exigir que el resto lo haga.

3 Temática

3.1. Objetivo del trabajo

El objetivo del trabajo es que el grupo observe, entienda y decida sobre los procesos, técnicas y herramientas relacionados con los temas de la asignatura. Para ello trabajará a lo largo del cuatrimestre con objeto de abordar un proyecto en grupo que cumpla con los requisitos que se enuncian en este documento. Se trata de que el grupo trabaje con código ya desarrollado y proponga cambios sobre el proyecto que redunden en alguna mejora del mismo.

3.2. Elección del proyecto

El grupo podrá elegir entre dos opciones: *i*) trabajar sobre un proyecto real *ii*) trabajar sobre un proyecto desarrollado en el curso anterior.

3.2.1. Proyecto real

Al inicio del curso estarán disponibles en la WIKI algunas propuestas de proyectos para trabajar sobre el sistema real. Cabe señalar que en caso de elegir esta propuesta se puede optar también a participar en el concurso de software libre ¹.

3.2.2. Proyecto de cursos anteriores

El grupo debe elegir un proyecto de los desarrollados en cursos anteriores para su estudio, análisis, documentación y toma de decisiones.

¹<https://www.concursosoftwarelibre.org/>

3.3. Evolución y cambios en el proyecto

Si se ha elegido el proyecto real, los cambios y modificaciones serán llevados a cabo según el enunciado que se proponga en coordinación con los desarrolladores del proyecto y, en su caso, con el coordinador de la asignatura.

Si se ha elegido un proyecto de cursos anteriores. El grupo debe proponer cuáles son los cambios que va a introducir al proyecto que podrían ser, aunque no están limitados a, alguno o varios de los siguientes:

- cambio completo en el lenguaje de programación usado justificando dicho cambio
- corrección de errores
- añadir nueva funcionalidad
- refactorizar el código, es decir, cambiar la arquitectura de alguna parte de la aplicación sin modificar su funcionalidad.
- sustituir algún elemento con el que interactúe la aplicación como, por ejemplo, la base de datos o el servidor en el que se despliegue.

Debe haber al menos una iteración funcional por miembro del equipo, es decir, cada miembro del equipo deberá hacerse cargo de al menos un incremento funcional. Estos incrementos no tienen por qué ser secuenciales en el tiempo y podrían desarrollarse en paralelo.

3.4. Actividad del proyecto

Se debe estar en torno a los siguientes indicadores de la actividad del proyecto para efectuar los cambios de evolución descritos anteriormente. Tenga en cuenta que esto son indicadores y por lo tanto no son datos totalmente rígidos pero sirven para dar una indicación de la dimensión del proyecto que se espera. Los cambios, al final, supondrán al menos:

- 200 líneas de código aproximadamente por miembro del equipo.
- 10 *commits* por miembro del equipo.
- 5 incidencias por miembro del equipo.

Aunque el principal objetivo del trabajo no es la parte de implementación del sistema, sí es necesario tener un número de cambios suficiente como para abordar el proyecto lo cuál se dimensionará por estos indicadores y por el proyecto global en sí.

4 Entregables

El trabajo elaborado por el equipo tendrá una serie de entregables. En los siguientes apartados se describen cuáles serán esos entregables y los apartados que tienen que contener.

Tenga muy en cuenta que a la hora de entregar el documento no podrá “copiar y pegar” de fuentes sin reconocer las mismas, es decir, **no se permite el plagio** y el plagiar será motivo de suspenso inmediato.

En caso de copiar un texto de manera literal se deberá reconocer la fuente desde la que se ha copiado y para ello, la frase copiada se pondrá entre comillas y cursiva y se pondrá una referencia o cita a su origen, por ejemplo así:

*“Continuous integration (CI) is the practice, in software engineering, of merging all developer working copies with a shared mainline several times a day. It was first named and proposed as part of extreme programming (XP)”*¹

4.1. Documento del proyecto

El documento del proyecto debe ser un documento que sintetice los aspectos del proyecto elegido para su desarrollo con respecto a los temas vistos en clases.

Debe tener claramente identificados los nombres y apellidos de cada componente, grupo al que pertenecen (1 o 2, mañana o tarde), curso académico, nombre del proyecto (deliberaciones, almacenamiento,...).

Poner de manera accesible los enlaces importantes que correspondan, por ejemplo, repositorio/s de código del equipo, repositorio compartido, repositorio de incidencias, wiki, despliegue, etcétera.

¹tomado de http://en.wikipedia.org/wiki/Continuous_integration. Ver el siguiente enlace para saber más sobre el plagio y cómo evitarlo: <http://bib.us.es/ximdex/guias/plagio/index.htm>

Debe ser un documento visual en el formato que elija el grupo. Puede ser una presentación, puede ser un vídeo, pueden ser una mezcla de diferentes estilos.

A pesar de ser un documento visual, debe ser un documento presentado de manera profesional guardando la forma en los estilos y contenidos y con el máximo nivel de rigor académico y profesional.

Tendrá (al menos) que sintetizar los siguientes aspectos del proyecto, no necesariamente estructurados en este orden:

1. **Resumen:** Se sintetizará de un vistazo lo hecho en el trabajo y los datos fundamentales. Se usarán palabras para resumir el proyecto presentado. Un buen resumen contendrá el problema que se ha tratado, por qué es un problema importante, la solución que se ha planteado, por qué es una buena solución y algunas conclusiones del trabajo.
2. **Introducción y contexto:** se pondrá en contexto el proyecto elegido y los aspectos fundamentales para entender el resto del documento. La introducción es como un resumen ampliado.
3. **Descripción del sistema:** se explicará el sistema desarrollado desde un punto de vista funcional. Se hará una descripción tanto descriptiva como técnica de sus componentes y su relación con el resto de subsistemas. Habrá una sección que enumere explícitamente cuáles son los cambios que se han desarrollado para el proyecto. En otra sección se presentará la planificación del trabajo y qué tareas se han hecho. Haciendo hincapié en el reparto de tareas. En este sentido que esté sentido es importante que estén explícitas las iteraciones funcionales que se van a realizar por cada parte del miembro del equipo. Tenga muy en cuenta lo expresado en la sección 2.2.2 en cuánto a división de tareas.
4. **Elementos de control:** se enumerarán cuáles son los elementos que estarán bajo gestión de la configuración y cómo se va a gestionar la configuración de cada uno de ellos. Es decir, se explicitará la primera tarea de indentificación de los *Configuration Items* (CI) que definan la *base line*.
5. **Entorno de desarrollo:** debe explicar cuál es el entorno de desarrollo que ha usado, cuáles son las versiones usadas y qué pasos hay que seguir para instalar tanto su sistema como los subsistemas relacionados para hacer funcionar el sistema al completo.
6. **Gestión del código fuente:** se explicarán los procesos, técnicas y herramientas para la gestión del código del proyecto. Evite poner información de las herramientas en sí que se pueda encontrar en fuentes bibliográficas o internet. Si es del caso haga referencia a ellas. Céntrese en los aspectos particulares de su proyecto, por ejemplo, ¿cómo se gestionan las ramas en el código? ¿cómo se aplica un parche (*patch*)? ¿cómo se aprueban los cambios? ¿qué roles existen en la gestión del có-

digo?, ¿qué políticas de nombre y estilo se utilizan en el código fuente? ¿cuál es el *usage model* del repositorio de código? Cada una de estos apartados deben estar divididos en secciones.

7. Gestión de la construcción e integración continua: similar al apartado anterior se definirán los procesos que se usan a la hora de construir el proyecto, ¿qué herramientas se usan? ¿cómo se usan? ¿cada cuánto tiempo se realiza una construcción del proyecto? ¿Qué mecanismos IC se usan? ¿cómo?, etcétera.
8. Gestión del cambio, incidencias y depuración: ¿qué mecanismos de depuración se usan? ¿cómo se gestionan los cambios? ¿qué procesos? ¿qué roles? ¿qué estados se manejan? ¿qué políticas para descartar, fomentar o retardar un cambio?. La gestión de incidencias debería contener explícitamente dos partes. Una de cómo se han gestionado la incidencias internas y otro el cómo se han gestionado y se ofrece protocolo para gestionar las incidencias externas tanto las recibidas como las que se reporten a otros subsistemas. Cuando una incidencia esté relacionada con un commit, señalar el commit dentro de la propia incidencia
9. Gestión de liberaciones, despliegue y entregas: ¿qué elementos del proyecto son “entregables”? ¿cómo se generan? ¿cómo se identifican? ¿cómo se gestiona la publicación, la liberación y entrega? ¿dónde se entrega? ¿qué roles existen en la entrega? ¿qué mecanismos de despliegue se definen? ¿qué procesos? ¿qué plataformas? ¿qué herramientas?
10. Mapa de herramientas: Debe dar un esquema de cómo se conectan las herramientas que se usan en el proyecto, qué relaciones tienen o qué relaciones propondría añadir. No olvide explicar bien el mapa de herramientas. Se trata del mapa de herramientas de gestión de la configuración. El mapa de herramientas que se usen para el desarrollo (como bases de datos u otros) no es necesario que aparezca aquí.
11. Conclusiones y trabajo futuro: Enunciar algunas conclusiones. Añadir una parte sobre las mejoras que se proponen para el futuro (curso siguiente) y que no han sido desarrolladas en el sistema actual.

Recuerde acompañar a la descripción abstracta de cómo hace una determinada tarea (e.g. cuál es el proceso de gestión de incidencias) con un ejemplo concreto en el que se vea cómo se ha gestionado en la práctica en su proyecto dicha tarea (e.g. gestión de una incidencia concreta de su proyecto).

Debe quedar claro, en cada parte, la diferencia entre:

- el proceso abstracto de gestión definido
- la herramienta que da soporte e implementa dicho proceso y cómo se usa dicha herramienta

- la posible descripción de la instalación de dicha herramienta si es que fuera relevante y necesario. En muchos casos podría ser más práctico poner un vídeo y poner un enlace al mismo describiendo estos pasos.
- la descripción de cómo se ha hecho en esa parte con el ejemplo concreto que se presente.
- Lecciones aprendidas si las hubiera. En esta partes pueden poner reflexiones y lecciones aprendidas. Se hizo algo de una forma y se ha visto que, tal vez, hubiera sido mejor hacerlo de otro modo, etcétera.

En el proceso abstracto se define la norma de uso del proceso sin especificar nada concreto con respecto a una herramienta determinada (e.g. GIT). En la parte de cómo dar soporte a ese proceso con una herramienta determinada se especifican detalles de cómo dar soporte al proceso abstracto con la herramienta. Por último, en el ejemplo práctico se debe poner por una parte un enunciado claro del ejercicio y por otra parte la solución a dicho ejercicio. El ejercicio tiene que ser algo concreto y personalizado para el proyecto en cuestión. No valen ejercicios genéricos que valdrían para cualquier proyecto. Hay una confusión generalizada entre usar una herramienta (por ejemplo jenkins o maven) y afirmar que se está siguiendo una aproximación determinada (por ejemplo de construcción automática o integración continua). **Las herramientas son medios que dan soporte a un proceso, no son el proceso.**

Es responsabilidad del equipo ser lo más completo posible en la elaboración del documento. No se trata de “rellenar por rellenar” o cubrir solo lo que se comenta en este documento. Piense en el trabajo como un entregable final profesional de modo que se evaluará el documento en su conjunto.

4.2. Máquina virtual

Se deberá entregar o intentar hacer accesible una o varias máquinas virtuales que contengan todos los elementos necesarios para realizar los ejercicios de los apartados anteriores además del proyecto en cuestión y todas sus herramientas.

Se recomienda usar un servidor de despliegue para poder desplegar la plataforma completa y tenerla accesible para la evaluación.

4.3. Diario del equipo

El equipo debe gestionar un documento como diario del equipo en el que se vayan reflejando todas las decisiones importantes del mismo. Este diario puede llevarse de manera digital o física a decisión del equipo y para ello puede usarse cualquier herramienta que se considere necesaria.

En cualquier caso el diario será uno de los entregables al final del proyecto. Debe contener las actas de las reuniones y las decisiones importantes que se tomaron. Vea ejemplos de años anteriores para elaborar el diario.

Además de decir cuándo se han tomado las decisiones, añadir qué tiempos y tareas se decidieron y qué imputación en horas real se ha hecho. También, en caso de haber diferencias significativas, se debe hacer un agregado por miembros del grupo y el total de horas imputadas. De no haber diferencias explícitas se considerará que todos los miembros del proyecto han aportado por igual. Si se quieren hacer diferencias, sería conveniente hacer algo que desglose de mutuo acuerdo el grado de implicación en el proyecto, por ejemplo, en una escala de 1 a 5. Si esto no está puesto explícitamente, se considerará que todos han trabajado de manera similar.

Si es posible incluir una foto de cada miembro del grupo al lado de su nombre.

4.4. *Milestones* intermedios

Los equipos deberán cumplir una serie de *milestones* intermedios definidos al inicio del curso y publicados en la WIKI de la asignatura. Se explicará lo que tienen que tener preparado para cada sesión.

4.5. Fecha para la entrega

Se anunciará en el WIKI de la asignatura una entrega final. Luego habrá la oportunidad de una entrega extra con revisiones.