

Grado en Ingeniería Informática
2020-2021

Apuntes

Dirección de Proyectos de Desarrollo de Software

Jorge Rodríguez Fraile¹



Esta obra se encuentra sujeta a la licencia Creative Commons
Reconocimiento - No Comercial - Sin Obra Derivada

¹Universidad: 100405951@alumnos.uc3m.es | Personal: jrf1616@gmail.com

ÍNDICE GENERAL

I	Introducción	1
1.	INFORMACIÓN	2
1.1.	Presentación	2
2.	MÉTRICA 3	3
2.1.	Proyectos de Desarrollo Software	3
2.2.	Metodología de Desarrollo Software	3
2.3.	Metodología basadas en planificación	4
2.4.	Métrica V3	4
2.5.	Oferta	5

Parte I

Introducción

1. INFORMACIÓN

Teorías: Ángel García Crespo acrespo@ia.uc3m.es

Prácticas: Israel González Carrasco igcarras@ia.uc3m.es

1.1. Presentación

Entregas prácticamente semanales.

Grupos de 7 personas, entre 83 y 84

Consultar continuamente a Israel.

A las revisiones deben asistir todos los participantes, se debe redactar un acta que se le enviara a él para que lo valide(él actúa como cliente).

Trabajo continuo, siempre hay un hito más.

Si se sigue evaluación continua no hay examen final.

50 % Trabajo en equipo, según las entregas, en las que se permiten 2 revisiones de cada entrega.

50 % Trabajo individual, 25 lo evalúan los compañeros y el otro 25 lo evalúa el profesor.

Todos los trabajos puntuables hay que aprobarlos (5).

Crear un correo de equipo para contactar con el profesor.

USAR el formato de Asunto indicado en las Normas.

Todas las entregas con un formato de documento fijo, con todas las partes básicas de un documento.

2. MÉTRICA 3

El cliente solo busca el producto final, no se hace a la idea de todo lo que hay detrás y que se debe tener en cuenta.

Resolver problemas y conflictos, colaborar con el cliente en la definición y consecución de objetivos, planificar el proyecto en todos sus aspectos, dirigir y coordinar recursos, tomar decisiones, etc.

2.1. Proyectos de Desarrollo Software

Proyecto: Tareas interrelacionadas, cosas que debe hacerse y completarse con tiempo determinado, con cierto coste y otras restricciones que pueden aplicarse para llegar a un resultado específico.

Empiezan con una idea u objetivo, que según se progresa se va detallando.

Son temporales, tienen un inicio y final.

Cada proyecto es único, pero siguen un conjunto de procesos.

Crea algo nuevo, una entrega tangible o intangible que es única.

2.2. Metodología de Desarrollo Software

Hay muchos participantes y stakeholders.

Se necesita:

- **Orden:** Definir un orden y establecer distintas prioridades para que el proyecto sea un éxito.
- **Azar:** Evitar que este condicionado por factores aleatorios, reducir el azar o suerte.
- **Eficiencia:** Realizar las tareas de modo racional, para lograrlas con en un menor tiempo y esfuerzo.

Para conseguirlo:

Metodología: Marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo. Herramientas, Métricas y Modelos.

Marco regulador: Establecer, ante problemas reales o potenciales, disposiciones destinadas a obtener un nivel de orientación y operación óptima. Normas y Estándares.

Hace énfasis al entorno en el cual se plantea y estructura el desarrollo de un sistema.

2.3. Metodología basadas en planificación

Énfasis en la **planificación de tareas y control del proyecto**, en especificación precisa de requisitos y modelado. Se centra en el control del proceso.

Principios:

- **Modelo de ciclo de vida:** Secuencia estructurada y bien definida de las etapas necesarias.
- **Ejecutar etapas solo una vez:** Lo que se define en cada etapa debería ser inamovible y hasta que no se completa con éxito no se pasa a la siguiente.
- **Diferentes profesionales:** Profesionales especializados pueden participar en diferentes etapas bien diferenciadas.
- **Diferentes roles:** Se definen más adelante.

El problemas: No está bien definido el volver a una etapa anterior, el usuario puede que no vea el producto hasta el final, puede no adaptarse bien a los cambios y se necesita tener unos buenos requisitos iniciales (que no siempre es fácil).

No existe una metodología universal para hacer frente a con éxito a cualquier proyecto de desarrollo de software.

La metodología debe ser adaptada al contexto del proyecto, los recursos técnicos y humanos.

2.4. Métrica V3

Perfiles: Permite definir que rol deben tomar los participantes.

Proceso:

1. Planificación de Sistema de Información
2. Procesos de desarrollo:
 - Estudio de Viabilidad del Sistema.
 - Análisis del Sistema de Información.
 - Diseño del Sistema de Información.
 - Construcción del Sistema de Información.
 - Implantación y Aceptación del Sistema.
3. Mantenimiento de Sistema de Información.

Descompone cada uno de los procesos en actividades, y estas a su vez en tareas.

Cada tarea se describe su contenido haciendo referencia a sus principales acciones, productos, técnicas, prácticas y participantes.

Interfaces:

- Gestión de Proyectos: Planificación, el seguimiento y control de las actividades, los recursos humanos y materiales que intervienen en el desarrollo.
- Aseguramiento de la calidad: Garantizar que el sistema resultante cumpla con unos requisitos de calidad.
- Seguridad: Gestión de riesgos y planes de contingencia, dota de mecanismos de seguridad adicionales.
- Gestión de configuración: Procesos de procedimientos administrativos y técnicos durante el desarrollo. Su finalidad es identificar, definir, proporcionar información y controlar los cambios en la configuración del sistema.

Participantes:

- Perfil Directivo: En nuestro proyecto no habrá uno.
- Perfil Jefe de Proyecto: Solo uno en el proyecto.
- Perfil de consultor: Consultor, Especialista en Comunicaciones, Técnico de sistemas y Técnico de Comunicaciones.
- Perfil de Analista: Analista, DBA, Equipo de proyecto, Equipo de Formación, etc.
- Perfil de Programador.
- Cliente y entidades externas: Clientes, Proveedores y Stakeholders.

2.5. Oferta

Lo más importante del proyecto, es que nos lo den, para ello hay que hacer una buena oferta. Una vez nos lo asignen ya nos preocupamos de los demás factores.

Ser capaz de **demostrar que se ha entendido lo que el cliente pretende**, el problema y como se va a solucionar.

Debe ser un **documento agradable a la vista**, tiene que demostrar el interés por el proyecto.

Sin faltas de ortografía, tipografía fija, facilitar la lectura. . .

Se debe **incluir el precio del proyecto**, como diferencia entre el coste y el precio.

Coste: Cuanto nos va a costar hacerlo a nosotros. Personal, equipo, viajes, ...

Costes directos: Si hay proyecto, el gasto que supone.

Costes indirectos: Los que tiene la empresa haya o no proyecto. Luz, internet, limpieza, jefe, comercial, gestoría, coche de jefe, comida cliente, RR. HH., etc. Se suman todos estos costes y se saca un porcentaje con respecto a los directos. De esta manera aplicamos sobre los costes directos este extra.

Precio: Lo que el cliente está dispuesto a pagar, no solo cubrirá el coste, sino que debe haber un margen de beneficio y un marco de riesgos.

EL CLIENTE SOLO CONOCE EL PRECIO, NO EL COSTE, NI BENEFICIO.

Beneficio: Lo que se gana, pero debe haber un extra por si aumentan los costes, para no tener perdidas.

El **precio se da SIN IVA**, por si cambia antes de que nos acepten la oferta, de esta manera ya se aplicará más adelante.

Hay que tener en cuenta el concepto de beneficio y el IVA.

Debe darse un **periodo de garantía** al cliente (en software no hay periodo obligatorio establecido)

La oferta tiene un **periodo de validez**, de 1, 2 o 6 meses, para que en un futuro no nos den ese proyecto. Puede haber cambiado la situación o estar desactualizado.

Hay que establecer la **forma de pago**, cuando se van a realizar los pagos. Cuanto es el adelanto, como se pagará durante el proceso y el pago final a la entrega.