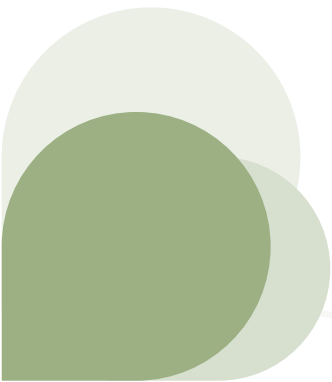


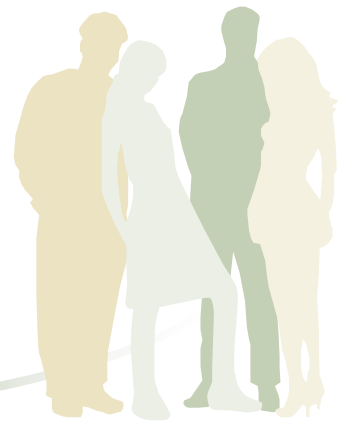
Tema 05 – Gestión de equipos

Ingeniería del Software



Rubén Fuentes Fernández
Dep. Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense Madrid

Trabajando con Antonio Navarro, Juan Pavón y Pablo Gervás



Contenidos

- Introducción
 - Problemática
- Aspectos de la solución
 - Selección de personal
 - Configuración de equipos
 - Organización



Introducción

- La gestión de proyectos software es vital.
- Una mala gestión produce:
 - Proyectos mal organizados
 - Fechas límite imposibles de cumplir
 - Sistemas que no cumplen lo que esperaban los usuarios
 - Sistemas imposibles de mantener
- Solución → buena gestión de proyectos software



Gestión de proyectos

- Cuando mucha gente trabaja en equipo, hace falta un trabajo explícito de administrar el trabajo que realizan:
 - Garantizar que no se repite trabajo
 - Procurar que en cada momento no haya nadie ocioso
 - Controlar que las cosas que se han encargado a alguien se están haciendo y van a estar a tiempo
 - Resolver las dudas en cuanto surgen, para que no se pierda tiempo esperando a la siguiente reunión de todo el grupo
- Sería ideal que eso ocurriera por sí solo...
 - ... pero eso solamente pasa cuando es muy poca gente,...
 - ... o que se pudiera hacer entre todo el equipo...
 - ... pero la verdad es que para que los equipos grandes funcionen hacen falta **responsables** de las tareas importantes.





Actividades de gestión

- Redacción de la propuesta
- Estimación del coste del proyecto
- Planificación del proyecto y elaboración del calendario
- Seguimiento del proyecto y revisiones
- Redacción de informes y presentaciones
- Resolución de problemas
- Selección y evaluación del personal
- Motivación del personal
- Control
- Organización del personal



Personal

- Se necesita personal preparado y motivado.
- Es el factor más valorado por los gestores de proyectos...
 - ... ya que en última instancia los proyectos los desarrollan personas.
- El SEI (*Software Engineering Institute*) ha desarrollado un “Modelo de madurez de la capacidad de gestión de personal” (PCMM) para estudiar la gestión del personal en organizaciones que desarrollan software.
 - <http://www.sei.cmu.edu/cmml/tools/peoplecmm/>





Participantes

- Cinco tipos de participantes en un proyecto software:
 - Gestores superiores
 - Definen los aspectos del negocio, que a menudo tienen una influencia significativa en el proyecto.
 - Gestores (técnicos) del proyecto
 - Planifican, motivan, organizan y controlan a los profesionales que desarrollan software.
 - Profesionales
 - Proporcionan las capacidades técnicas necesarias para la ingeniería de un producto o aplicación.
 - Clientes
 - Especifican los requisitos del proyecto.
 - Usuarios finales
 - Interactúan con el software una vez que se ha entregado.



Jefe de equipo

- Es la persona que lidera a un equipo.
- En función de la organización y el proyecto puede ser un gestor técnico o un profesional.
- Un buen profesional informático no tiene porque ser un buen jefe de equipo.





Jefe de equipo: rasgos

- Motivación → Capacidad para motivar al personal
- Organización → Capacidad para aplicar procesos
- Ideas o innovación → Capacidad para fomentar la creatividad del personal
- Resolución de problemas → Capacidad para entender el problema, gestionar las ideas y proporcionar una solución
- Competencia tecnológica → Grado de conocimiento de las últimas tecnologías
- Dotes de gestión → Capacidad para manejar el problema y el personal
- Incentivo de logros → Capacidad para optimizar la productividad de un equipo premiando la iniciativa y los logros
- Influencia y construcción del espíritu de equipo → Capacidad para cohesionar al grupo y entender los problemas personales



El equipo de desarrollo

- El jefe de personal elige al personal para cada proyecto.
- Valora:
 - Experiencia en el dominio de la aplicación → Para desarrollar un proyecto con éxito debe entender el dominio
 - Experiencia con la plataforma → Sólo es importante si hay programación de bajo nivel
 - Experiencia con el lenguaje de programación → Sobre todo en proyectos cortos
 - Fondo educativo → Nuevos miembros del equipo
 - Capacidad de comunicación → Comunicarse con los otros miembros del equipo, con los gestores y con el cliente
 - Adaptabilidad → Capacidad de aprender
 - Actitud → Ante el trabajo y las dificultades
 - Personalidad → Capacidad de trabajo en grupo

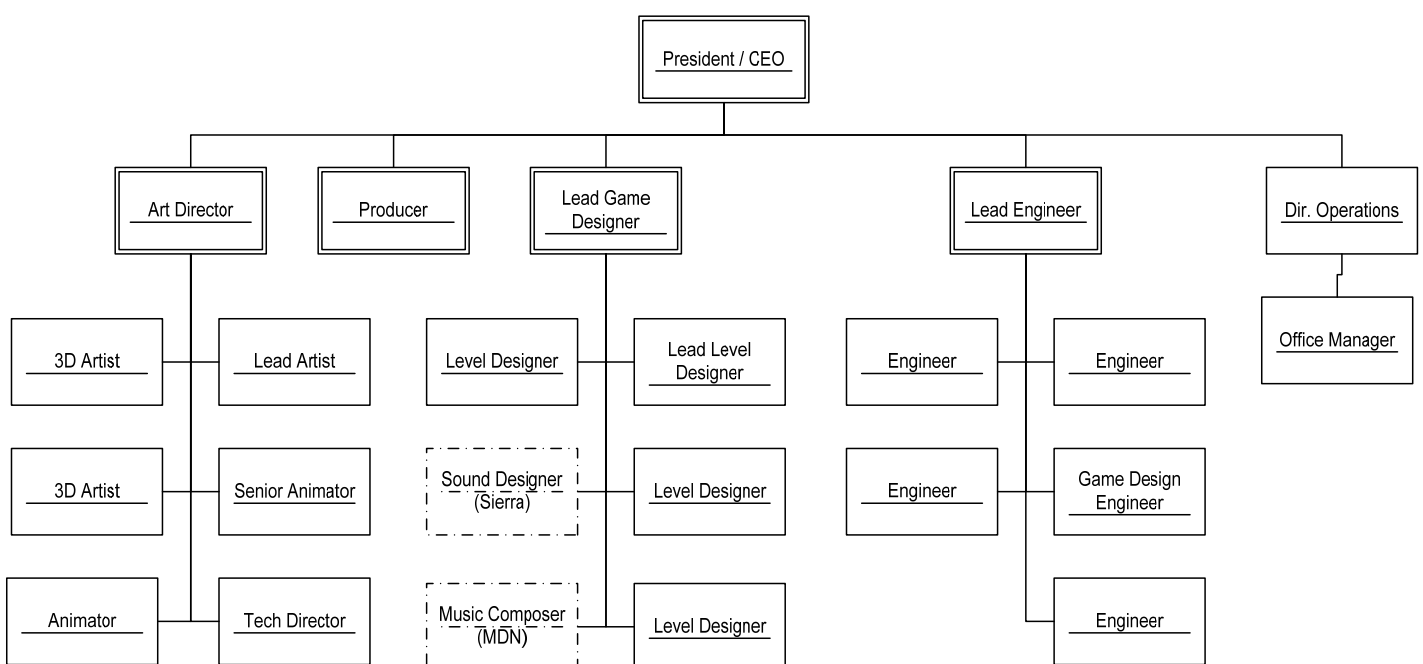


Organización del equipo de desarrollo

- El gestor del proyecto es el responsable de organizar el personal.
- Opciones:
 - Individuos trabajando de forma independiente en distintas tareas, con poco trabajo conjunto.
 - Coordinación por el gestor del proyecto.
 - Formación de equipos informales que acometen distintas tareas, donde se puede elegir un jefe.
 - Coordinación por el gestor del proyecto.
 - Organización de equipos bien determinados, donde a cada equipo se le asigna un conjunto de tareas bien definido.
 - Cada equipo tiene una estructura específica y bien definida.
 - La coordinación se divide entre el equipo y el gestor del proyecto.
- De estas opciones, la más productiva es la última.



Estructura de equipo



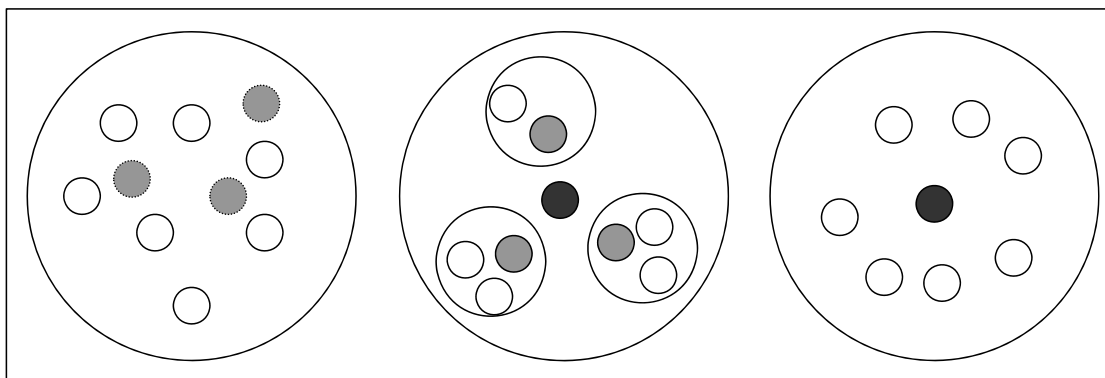


Factores a considerar

- Las diferentes estructuras de equipo potencian distintos factores:
 - Jerárquica o democrática
 - Jefe o consenso
 - Flexibilidad o control
 - Imaginación o rutina



Estructuras de equipos de desarrollo



Descentralizado
Democrático (DD)

Descentralizado
Controlado (DC)

Centralizado
Controlado (CC)

Estructuras de equipo [Mantei, 1981]





Descentralizado democrático

- No tiene un jefe permanente.
- Se nombra un jefe en función de cada tarea.
- Las decisiones, problemas y enfoques se llevan a consenso del grupo.
- La comunicación entre los miembros del equipo es horizontal.



Descentralizado controlado

- Tiene un jefe de equipo para las tareas.
- Tiene jefes secundarios para subtareas.
- La resolución de problemas se hace en grupo.
- El jefe de grupo distribuye la implementación de tareas entre los subgrupos.
- La comunicación entre subgrupos e individuos es horizontal.
- Hay comunicación vertical entre los jefes secundarios y el jefe de equipo.





Centralizado controlado

- Hay un único jefe de equipo.
- Este jefe resuelve los problemas a alto nivel, y la coordinación del equipo.
- Comunicación vertical entre el jefe y los miembros del equipo.
- Fue el primer organigrama que se empezó a aplicar.



Determinar la estructura

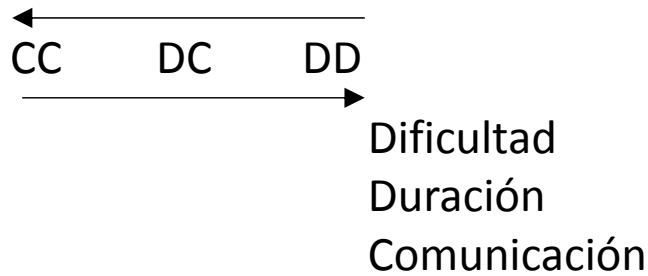
- Mantei [1981] identifica siete factores de un proyecto* para determinar la estructura a elegir:
 - Dificultad del problema
 - Tamaño en Líneas de Código (LDC) o Puntos de Función (PF)
 - Duración del equipo
 - Modularidad del problema
 - Calidad y fiabilidad del sistema a construir
 - Fecha de entrega
 - Comunicación requerida en el proyecto

*El tamaño del equipo también es relevante



Determinar la estructura

- Tamaño
Modularidad
Fiabilidad
Fecha entrega



Factores Mantei y estructura de equipo



Dirección y jerarquía

- Los jefes deciden, organizan y mandan
 - A partir de 4 personas, hace falta jefe.
 - A partir de 6 personas, hacen falta sub-jefes.
 - La disciplina es vital, pero hay que dar margen a la creatividad y ambición individuales.
- Encargados de área
 - Jefes de departamento
 - Jefes de equipo
 - Resto de miembros del equipo
- No se deben tomar decisiones en los niveles equivocados.
 - Definir adecuadamente las interacciones en el equipo.
 - Ej. si se permite la comunicación directa entre programadores pueden tomarse decisiones apresuradas que se desconocen en los niveles superiores y pueden tener repercusiones negativas a medio plazo.





Cohesión

- Con independencia de la estructura, la cohesión del equipo es fundamental.
- Ventajas de un equipo cohesionado:
 - Desarrollo estándar de calidad de grupo.
 - Los miembros del grupo trabajan mejor juntos.
 - Se conoce el trabajo de los otros miembros.
 - Se practica la programación sin ego.
- Inconvenientes:
 - Resistencia al cambio en el liderazgo.
 - Pensamiento de grupo.



Problemas clave

- Hay cinco problemas clave que afectan a la cohesión del equipo:
 - Atmósfera de trabajo frenética
 - Frustración causada por factores tecnológicos, del negocio o personales
 - Falta de un modelo de proceso adecuado
 - Definición confusa de los roles
 - Continua y repetida exposición al fallo





Potenciar el trabajo efectivo en grupo

- Distribución clara de roles en el equipo
 - Roles asociados a responsabilidades y tareas
- Cauces de comunicación bien definidos
 - Mecanismos, qué se comunica y a quién se comunica
- Mecanismos claros de toma de decisión
 - Cómo se resuelven los problemas y quién decide sobre ellos
- Seguimiento del trabajo
 - Métricas, reuniones e informes sobre la marcha del trabajo
 - Criterios de evaluación de los anteriores
 - Gestión de riesgos específicos
- Potenciar la visión de equipo
 - Pero manteniendo la responsabilidad individual



CONCLUSIONES





Conclusiones

- Los proyectos de desarrollo software son complejos desde el punto de vista de la gestión de sus equipos.
- Se requiere una coordinación y comunicación adecuada para atender a factores como:
 - Escala
 - Incertidumbre
 - Interoperatividad
- La gestión del equipo no surge improvisadamente, sino que es otro aspecto del proyecto que se ha de gestionar específicamente.



Glosario

- CC = Centralizado Controlado
- DC = Descentralizado Controlado
- DD = Descentralizado Democrático
- LDC = Línea de Código
- PCMM = *People Capability Maturity Model*
- PF = Punto de Función
- SEI = *Software Engineering Institute*





Referencias

- R. Pressman: Ingeniería del Software. Un enfoque práctico, 7ª edición. McGraw-Hill, 2010.
 - Capítulo 3
- I. Sommerville: Ingeniería del Software, 7ª edición. Addison Wesley, 2007.
 - Capítulo 22
- CMU-SEI: People CMM – Managing and developing the workforce of an organization. CMU-SEI, 2011. Disponible en: <http://www.sei.cmu.edu/cmmi/tools/peoplecmm/>, accedido en 10/11/2011.
- Mantei M.: The Effect of Programming Team Structures on Programming Tasks. Communications of the ACM 24(3), pp. 106-113, ACM, 1981.



EXTRAS

