



Gestión de proyectos

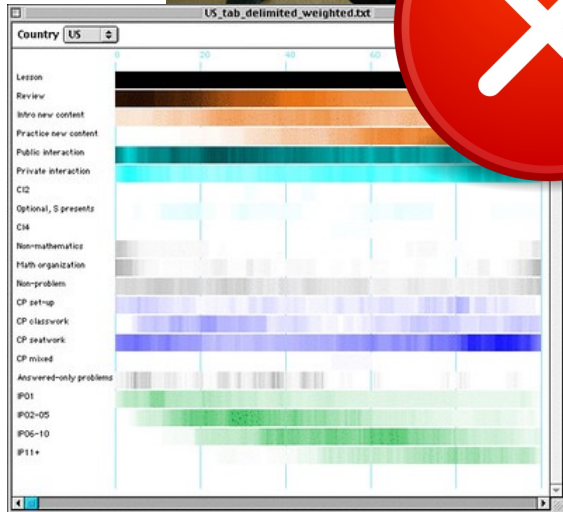
Capítulo 22 – Sommerville



Objetivos

- Completar la introducción que hemos realizado en las sesiones anteriores en relación con la gestión de proyectos para enlazar con el enfoque de Sommerville.
- Conocer con más detalle dos de las áreas de gestión de proyectos
 - Gestión de riesgos
 - Gestión de personal
 - Factores que influyen en la motivación personal
 - Aspectos clave que impactan en el trabajo en equipo (composición, organización y comunicación)

Gestión de proyectos software

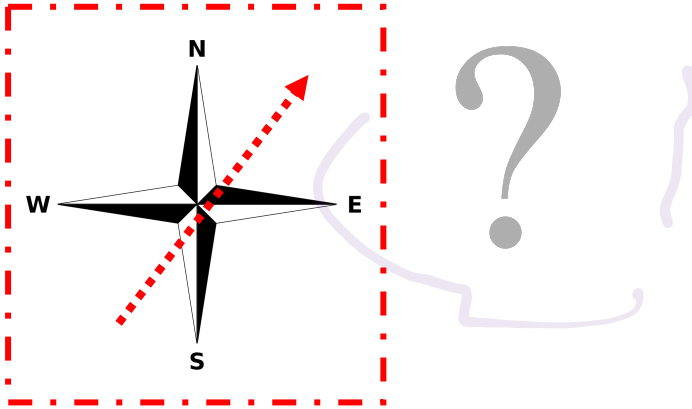


- Es el conjunto de actividades dirigidas a asegurar que el software se entrega a tiempo, de acuerdo con lo planificado y conforme a los requisitos de la organización u organizaciones que financian el proyecto para **conseguir un objetivo de negocio.**



La importancia de la gestión de proyectos

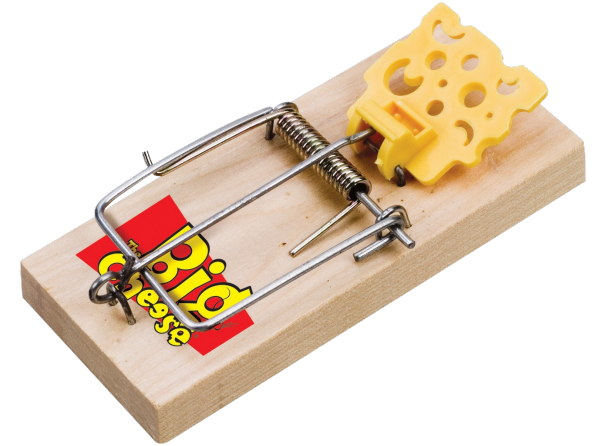
Bu\$ine\$\$



Alcance definido y controlado



Comprensión común



Visibilizar los riesgos

Condición necesaria pero no suficiente

Crterios de éxito de un proyecto





ITIL – Utilidad y Garantía



- **Utilidad (Utility)**

- La funcionalidad ofrecida por un producto o servicio para **satisfacer una necesidad particular**. A veces se resume como “lo que hace el servicio” y puede utilizarse para determinar si un servicio es capaz de **cumplir con los resultados esperados** o si es “se adecúa a su propósito”

- **Garantía (Warranty)**

- El **compromiso de que el producto satisfará los requisitos acordados**.
- El término hace referencia a la disponibilidad del servicio cuando se necesita, a que proporciona la capacidad requerida, y a que es fiable en términos de seguridad y continuidad.

Particularidades de la gestión de los proyectos de desarrollo software



- **El producto es intangible**
 - No se puede ver ni tocar, por lo que el jefe de proyecto no puede hacerse una idea del progreso echando un vistazo a los “artefactos” que se van generando.
- **Con frecuencia los proyectos software son únicos (“one-off”)**
 - En estos casos no podemos beneficiarnos de que en su totalidad, o en gran medida, los esfuerzos se puedan replicar, lo que implica que incluso gestores con gran experiencia tengan dificultad para anticipar los problemas.
- **Los procesos software suelen ser variables y específicos de cada organización**
 - Nos cuesta predecir de forma fiable cuándo cierto proceso nos va a dar problemas durante el desarrollo

Factores que influyen en la gestión de proyectos

- Tamaño de la compañía
- Los clientes (internos, externos, organismos gubernamentales)
- Tamaño del software
- Tipo de software (Sw comercial, sistema crítico)
- Cultura corporativa (riesgo, centrada en individuos o en grupos)
- Procesos de desarrollo de software (Agile vs tradicionales)

En función de estos factores la forma concreta en que se desarrolla el trabajo de un jefe de proyecto varía significativamente

Actividades comunes a cualquier proyecto

- **Planificación**

- Los jefes de proyecto son responsables de la planificación, estimación y “calendarización” de las actividades y recursos del proyecto y de asignar tareas a los miembros del equipo.

- **Gestión de riesgos**

- Junto al equipo de proyecto deben asegurarse de que se identifican los riesgos que puedan afectar al normal desarrollo, definiendo un plan de acción, realizando un seguimiento e implementando las acciones definidas en el plan.

- **Gestión de personal**

- Los jefes de proyecto son responsables de formar el equipo que debe llevarlo a cabo, desarrollarlo y gestionarlo durante la vida del proyecto.



Actividades comunes a cualquier proyecto

- **Informar (*reporting*)**

- Los jefes de proyecto son habitualmente los responsables de informar sobre el progreso del proyecto tanto a los clientes como a los responsables de la compañía que está desarrollando el software.

- ***Redacción de propuestas***

- La primera etapa en un proyecto software puede involucrar la redacción de una propuesta con objeto de ganar un contrato para desarrollar un producto o un servicio. La propuesta describe los objetivos del proyecto y cómo se van a desarrollar.





PMP: Actividades de planificación que habría que realizar antes de empezar a planificar riesgos

1. Determinar cómo vamos a planificar cada área de conocimiento
2. Determinar los requisitos detallados del proyecto
3. Crear el *project scope statement*
4. Valorar qué partes del proyecto se van a subcontratar y preparar los documentos asociados
5. Determinar cuál va a ser el equipo de planificación
6. Crear las WBSs y el diccionario WBS
7. Crear la lista de actividades
8. Crear un diagrama de actividades
9. *Estimar*(Identificar) los recursos necesarios
10. Estimar el tiempo y el coste
- 11. Determinar el camino crítico**
12. Desarrollar el plan de tiempos
13. Desarrollar el presupuesto
14. Determinar los estándares de calidad que se van a aplicar, los procesos y las métricas
15. Crear un plan de mejora de procesos
16. Determinar todos los roles y responsabilidades
17. Planificar la gestión de los *stakeholders*
18. Identificar riesgos (cualitativos, cuantitativos y preparar un plan de respuesta)



PMP: Definiciones

- **Project Scope Statement:** Una descripción del alcance del proyecto que incluye los **entregables principales**, las **hipótesis de trabajo** y las **limitaciones identificadas** del proyecto.
- **Work Breakdown Structure (WBS):** Una descomposición jerárquica del alcance total del proyecto que debe ser ejecutado por el equipo para lograr los objetivos del proyecto y desarrollar los entregables requeridos.
 - El nivel más bajo de una WBS es el **workpackage**
 - Un **entregable** es un resultado que aporta algún valor a los *stakeholders* del proyecto
- **Diccionario WBS:** Es un documento en el que, para cada componente de la WBS, se proporciona información detallada de cada entregable, de las actividades asociadas y de su planificación temporal
 - Una **actividad** es una porción diferenciada y planificada del trabajo que se ha de realizar durante el desarrollo de un proyecto.

Scope Baseline



Definición de *Workpackage* (a.k.a. paquete)

- El nivel de granularidad más bajo de la WBS para el que se puede estimar y gestionar una duración y un coste.
- Características:
 - Es el nivel más bajo de entregable
 - No puede descomponerse en entregables más pequeños
 - Proporciona un **valor tangible** a los *stakeholders* del proyecto
 - **Es requerido, reconocido y valorado**
 - Una vez creado se puede hacer entrega a los *stakeholders* del proyecto
 - Cuando se completa permite
 - Satisfacer las expectativas de los *stakeholders* del proyecto
 - Que el equipo de proyecto tenga una sensación objetiva de logro.



Definición de actividad

- Una actividad es **parte del trabajo necesario para conseguir un entregable**
 - Por tanto **se puede medir su duración y su coste**
 - Contribuye al valor del *workpackage*, pero **en sí misma no es portadora de valor**.
 - No puede ser objeto de una entrega, no es transferible
 - Ejemplos: verificar el código, localizar y adquirir un componente necesario para el desarrollo, investigar los términos de la licencia de un componente suministrado por una tercera parte, etc.
- Un conjunto de actividades relacionadas con un propósito dan como resultado un *workpackage*.



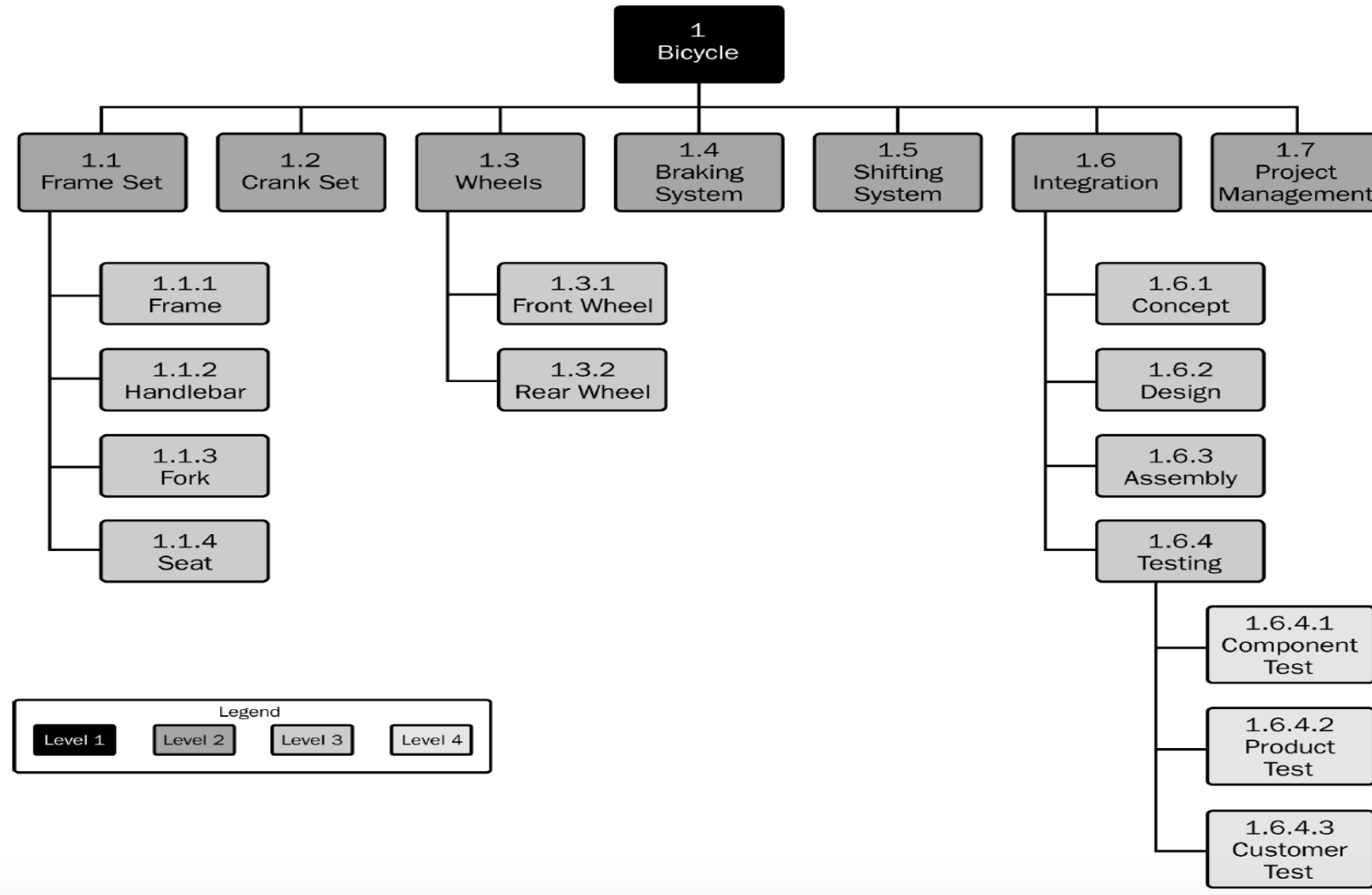


Diferencia entre *workpackage* y actividad

| Workpackage | Actividad |
|--|---|
| Es el nivel más bajo de entregable en un proyecto | Es una forma de crear un entregable. Un conjunto de actividades coordinadas con un propósito dan como resultado un <i>workpackage</i> |
| Es el resultado o <i>producto</i> de un esfuerzo | Es un elemento del esfuerzo, un paso de cara a lograr su resultado |
| Satisface parcialmente los objetivos de un proyecto. Aporta valor y satisfacción a los <i>stakeholders</i> . | En sí misma una actividad no proporciona valor a los <i>stakeholders</i> |
| Tiene un valor tangible y es objeto de transferencia a los <i>stakeholders</i> del proyecto | Una actividad no puede ser objeto de una entrega a los <i>stakeholders</i> |
| Normalmente se define usando un nombre, e.g. Modulo X, Subsistema Z | Normalmente se define usando un verbo: Diseñar, verificar, codificar, etc. |

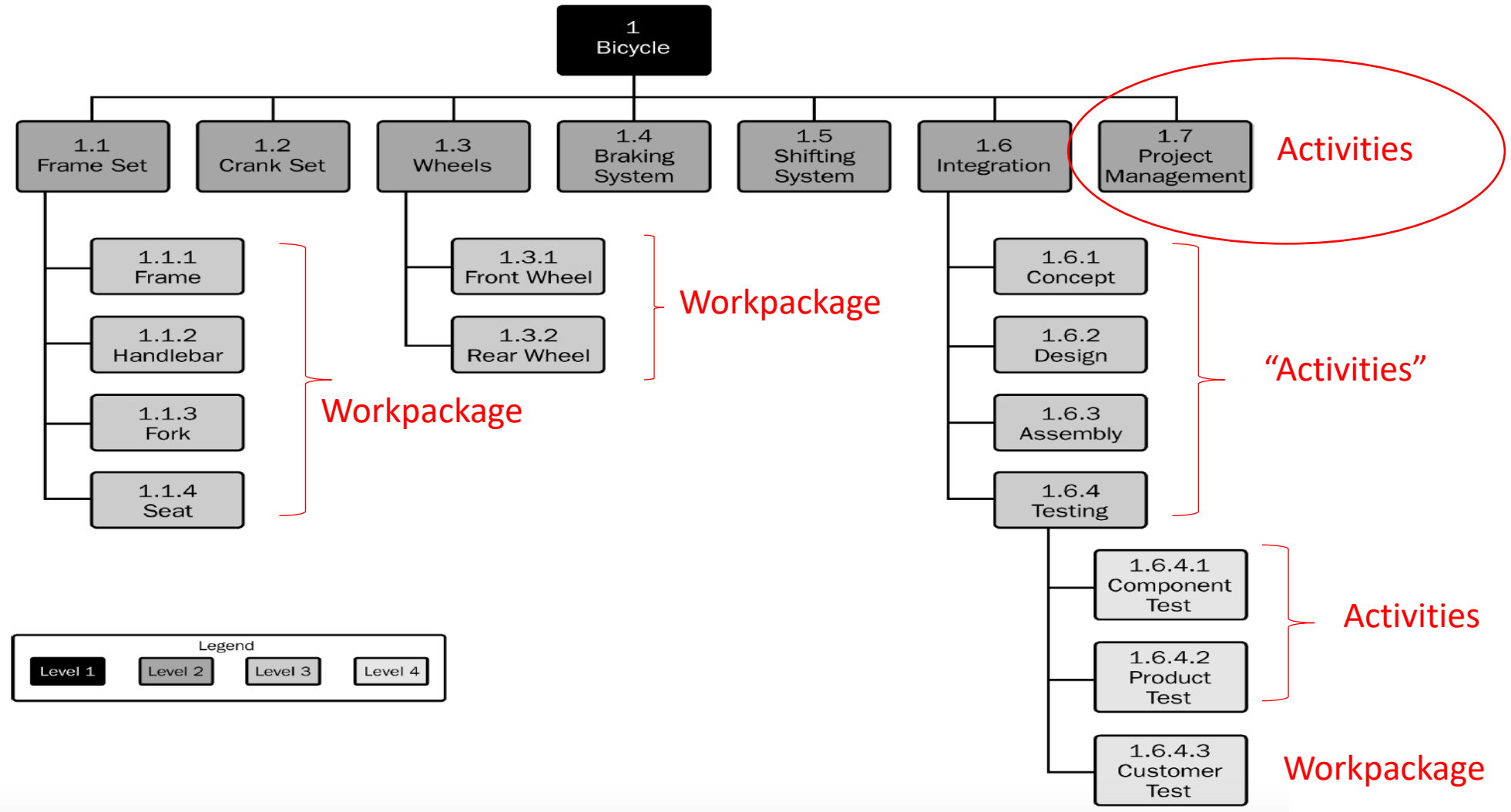


Ejemplo de una WBS para una bicicleta





Ejemplo de una WBS para una bicicleta



Q&A

