

# Actividad 6

## Planificación y temporalización

Autor: José Manuel Beas ([jbeasa@uoc.edu](mailto:jbeasa@uoc.edu))

### Planificación de recursos

#### Personas

Los recursos humanos asignados al proyecto (en esta "Fase 1") son los siguientes:

- **Jefe de proyecto:** Con una asignación máxima a este proyecto del 75% debido a las características del proyecto (es un proyecto interno con grandes limitaciones presupuestarias) y debido a que la persona que implementa dicho rol aún debe hacer el seguimiento de un proyecto anterior que se está cerrando (está en fase de "Garantía").
- **Analista funcional:** Una persona con una asignación máxima de un 50% (porque está involucrado en otro proyecto).
- **Arquitecto de Sistemas:** Los arquitectos en general en esta compañía no se pueden asignar a un proyecto en más de un 75% (por norma, puesto que se considera que es conveniente otorgarles cierto margen para tareas de soporte, formación, evaluación, etc.).
- **Técnico de Sistemas:** El Departamento de Sistemas tiene siempre muchos problemas para asignar personal a proyectos internos porque la mayoría de sus técnicos están fuera de las oficinas de la compañía y porque el personal asignado al mantenimiento de dichas instalaciones está sobreutilizado. Por lo tanto, existe una limitación en número de jornadas y horas diarias disponibles del técnico de sistemas asignado: sólo las tardes de la primera semana del proyecto y un día completo de la siguiente semana.
- **Arquitecto Software:** De manera extraordinaria, se ha conseguido que en las dos primeras entregas el arquitecto software pueda tener una asignación de un 100%, con el doble objetivo de avanzar rápidamente en la puesta en marcha del proyecto y de dar una formación práctica a los dos programadores asignados.

- **Programadores J2EE:** Hemos conseguido la asignación de dos programadores J2EE. Aun siendo *juniors*, ambos tienen suficiente experiencia y actitud como para hacerse cargo sin demasiada supervisión de las tareas que se les asigne. Sin embargo, existen tareas de alta complejidad para las que será necesario que los arquitectos de sistemas y *software* se encarguen al menos de resolver los primeros pasos. Así, además, se consigue que estos *juniors* vayan adquiriendo conocimientos y experiencia para el futuro. La motivación en el desarrollo de su carrera profesional es un valor importante dentro de la compañía.
- **Diseñador gráfico:** Este recurso no es crítico porque hay disponibilidad y porque no son costosos. Además, las tareas que se tienen previstas para ellos no son críticas.

## Software

Todo el *software* a emplear será de dominio público (*open source*), con lo que no habrá mayor coste asociado a este capítulo que el de localizarlo, evaluarlo e instalarlo en los diferentes equipos. Las dos primeras tareas están fuera del ámbito del proyecto, no así la última. En cualquier caso, el coste de la misma será únicamente el asociado al Técnico de Sistemas encargado de la misma.

## Hardware

No hemos incluido ninguna planificación temporal sobre material *hardware* porque no se podrá comenzar el proyecto hasta que no esté disponible. Dicho material será el siguiente:

- un servidor para las pruebas integración
- acceso a éste desde internet
- equipos de desarrollo

Cada programador de la compañía tiene un equipo de desarrollo asignado, con lo que simplemente se debe conseguir que éste tenga el *software* necesario (y cumpla con los requisitos mínimos necesarios para realizar su trabajo, e.d., que tenga memoria, CPU, conectividad, etc. adecuados).

## Presupuesto

El coste calculado de este proyecto consiste fundamentalmente en dos partidas:

- **Salarios:** Calculado en función de unas tablas de uso interno de la compañía que establece el coste interno de un empleado según su rol dentro de un proyecto. El montante total es aproximadamente de 30.000 €.

- **Hardware:** En el caso de este proyecto consistirá en el coste calculado del servidor de integración y que viene dado por un valor interno por días de uso para el desarrollo (unos 10 €/jornada), lo que representa un total de aproximadamente 700 €.

## Análisis coste-beneficio

Una vez decidida la arquitectura del sistema, "el qué", se debe decidir "el cómo". Y aquí nos encontramos con dos opciones: acometer todo el desarrollo o implantar un producto comercial (en este caso IBM Lotus *QuickPlace*). Entendemos que habría que hacer un análisis de riesgos y de coste-beneficio para ambas opciones. Nosotros nos vamos a decantar por la primera por las siguientes razones:

- Se pretende aprovechar el abordaje de este desarrollo para formar a parte de la plantilla en las tecnologías involucradas en el mismo.
- También se quieren obtener experiencias que permitan mejorar la metodología de desarrollo empleada hasta ahora.
- Otra razón a añadir es la libertad que ofrece el ser propietario de la solución frente a un producto cerrado.
- Además, con este producto no se tiene experiencia en la compañía, lo que representa un riesgo añadido para el proyecto.
- Y por si fuera poco, hay que añadir el coste de adquisición del producto.

Así pues, vamos a analizar la relación coste-beneficio del desarrollo propio de este proyecto:

### Beneficios tangibles e intangibles

Es difícil cuantificar los beneficios tangibles que se producirán por la introducción del nuevo portal en la intranet, pero podríamos ("a ojo de buen cubero") decir que el trabajo de los consultores y gerentes se verá beneficiado en al menos un 2% (unos 2.500 €/mes<sup>1</sup>) gracias a que el acceso al correo-e, a la documentación, a un directorio de empleados y clientes, a los proyectos y componentes desarrollados en la compañía estará centralizado y de acceso incluso desde fuera de las oficinas.

Este último detalle de permitir compartir mucha información corporativa a través de internet puede permitir que muchos de estos consultores y gerentes realicen parte de sus tareas semanales desde fuera de las oficinas, con el consiguiente ahorro y comodidad para ambos (empleados y compañía).

---

<sup>1</sup> Calculado como el 2% de un coste medio por empleado de 200 € / jornada y unos 30 empleados favorecidos aproximadamente en una primera fase de implantación.

Además de una homogeneización en el acceso a la información, también se conseguirá la puesta en marcha de una metodología de desarrollo avalada por el uso de herramientas concretas. Esto también representará una mejora en la calidad de la producción y unos menores costes (tanto en la producción como en el mantenimiento). Esto también es difícil de calcular, pero nos atrevemos a estimar que un proyecto cualquiera se podría ver beneficiado en un 5% por la introducción de estas herramientas y métodos (lo que puede significar un ahorro del orden de 8.000 €/mes<sup>2</sup>).

Por si fuera poco, habría que añadir un beneficio intangible muy importante desde el punto de vista de la Compañía: la promoción profesional de dos *juniors*, lo que beneficia a la organización, a ellos y a la motivación del resto de compañeros.

### Costes tangibles e intangibles

Los costes tangibles son muy evidentes: los salarios (que ya hemos visto que se podrían valorar en unos 30.000 € durante los tres meses de duración del proyecto) y el *hardware* necesario (que serían unos 700 € también durante los tres meses). Una vez implantado el sistema, requerirá un mantenimiento, pero éste se estima que no superará los 2.000 €/mes<sup>3</sup>.

### Retorno de la inversión (ROI)

ROI = (Beneficio neto anual - Coste desarrollo anualizado) / Inversión promedio

$$Bna = (8000 + 2500) \times 12 = 126.000 \text{ € / año}$$

$$Cda = 30000 + 2000 \times 12 = 54.000 \text{ € / año}$$

$$Ip = Cda + 700 = 54.700 \text{ €}$$

Por tanto, ROI =  $(126.000 - 54.000) / 54.700 = 132 \%$

Esto representa un retorno de la inversión realmente espectacular; considérese que el valor normal de esta ratio en un proyecto cualquiera está en torno al 20 ó 30 %.

## Temporalización

### Definición y Secuencia de las actividades

En el documento adjunto ("Plan de Proyecto") se describe la secuencia de actividades en detalle. Sin embargo, aquí nos gustaría explicar las restricciones más importantes y las guías seguidas para obtener dicha planificación.

---

<sup>2</sup> Calculado sobre una media de 4 proyectos de 40.000 €/mes de presupuesto.

<sup>3</sup> Calculado como el coste medio de la dedicación de las personas encargadas de dicho mantenimiento.

Se ha planteado el plan de proyecto dividido en entregas para intentar aplicar metodologías ágiles como XP (eXtreme Programming). La idea es tener el sistema funcionando lo más pronto posible y luego ir añadiendo funcionalidades. Así se consigue encontrar fallos de planteamiento muy pronto y adaptarse a los cambios con mayor facilidad. Tiene muchos pros y muchos contras, pero eso es otra discusión.

En el caso concreto de nuestro proyecto, las entregas se plantean como una cadena secuencial de tareas. Así, cuando se concluye la primera entrega, se evalúa la situación (posibles desviaciones, modificaciones sobre los requerimientos, etc.) y se aborda la siguiente entrega con las modificaciones que sean necesarias pero, sobre todo, con un sistema ya integrado en el entorno *de producción*, lo que valdrá al jefe de proyecto para reducir las típicas tensiones provocadas por no poder demostrar a los clientes (en este caso la Dirección), y a los propios miembros del equipo, los resultados obtenidos a medida que se van realizando las diferentes entregas.

Cada entrega se considera concluida cuando ha finalizado con éxito la tarea de "pruebas de integración". Además de estas pruebas de integración, cada desarrollador puede realizar las pruebas de los componentes *software* que haya necesitado desarrollar gracias a un sistema automático que comprueba los casos de prueba definidos para ellos al principio del desarrollo de cada componente. Éste es uno de los valores añadidos que ofrece el desarrollo de este proyecto a la compañía: una mejora en la metodología de desarrollo.

A *grosso modo* describimos a continuación las principales tareas implicadas en este desarrollo por entregas. Entre paréntesis se indican las fechas previstas para las entregas. En el caso particular de nuestro proyecto, estas fechas son realmente de uso interno del propio equipo de desarrollo puesto que no tienen repercusión sobre terceros, sin embargo, es responsabilidad del jefe de proyecto hacer cumplir escrupulosamente estas fechas puesto que se corre el riesgo de incurrir en retrasos inaceptables.

En general, se han tratado de planificar las tareas más complicadas tecnológicamente al principio debido a la cercanía de las vacaciones de verano, que disminuye drásticamente la disponibilidad y continuidad de los recursos. También se han dejado las búsquedas para la última entrega porque se ha considerado como una funcionalidad de la que bien se podría prescindir en caso de desviaciones graves en la planificación.

- Entrega #1 (27/06/03, 15 jornadas)
  - Entorno de desarrollo
  - Entorno de pruebas y demo
  - Autenticación
  - Autorización
  - Pruebas de integración #1
  - Formación del equipo de desarrollo
- Entrega #2 (15/07/03, 12 jornadas)
  - Administración autorización
  - Documentos (sólo lectura)
  - Componentes (sólo lectura)
  - Pruebas de integración #2
- Entrega #3 (24/07/03, 7 jornadas)
  - Directorio (presentación en árbol)

- Directorio (búsquedas en LDAP)
  - Configuración (página personal)
  - Pruebas de integración #3
- Entrega #4 (5/08/03, 8 jornadas)
  - Directorio (clientes)
  - Publicación de documentos
  - Publicación de componentes
  - Pruebas de integración #4
- Entrega #5 (18/08/03, 8 jornadas)
  - Mi Correo (leer)
  - Mi Correo (enviar)
  - Ayuda
  - Pruebas de integración #5
- Entrega #6 (28/08/03, 8 jornadas)
  - Proyectos
  - Tareas
  - Mis Tareas (sólo lectura)
  - Ayuda
  - Mi Correo (gestión de carpetas)
  - Mi Correo (configuración)
  - Pruebas de integración #6
- Entrega #7 (10/09/03, 9 jornadas)
  - Mis Tareas
  - Búsquedas (en documentos)
  - Búsquedas (en componentes)
  - Búsquedas (en proyectos)
  - Búsquedas (en páginas personales)
  - Pruebas de integración #7

Además, también se consideran dos tareas importantes pero, en cierto modo, fuera del ámbito de las entregas: la documentación y la formación a los usuarios.

## Dependencias

Todas las dependencias están descritas en el documento adjunto "Plan de Proyecto". Se han intentado paralelizar tareas todo lo que la disponibilidad de recursos nos ha permitido. No han sido muchas, de ahí que casi todas las tareas formen parte del camino crítico del proyecto.

## Riesgos

- **Vacaciones:** En el momento de realizar la planificación del proyecto se desconocen los días en los que algunas de las personas involucradas en el mismo estarán de vacaciones. Hay que tener en cuenta que el mes de agosto se encuentra en medio de todo el desarrollo del proyecto, con lo que la incidencia de las vacaciones es un factor de riesgo importante que será necesario controlar. Es muy posible que sea necesario plantear sustituciones temporales de alguno de los miembros del equipo, con lo que habrá que tener en cuenta el efecto sobre las tareas, el sobrecoste que esto representa (tanto en tiempo como en dinero) y el solapamiento necesario para realizar la formación de los sustitutos.

### **Duración de las actividades**

Dado que las tareas se han agrupado en entregas con una duración de alrededor de dos semanas, las actividades no tienen duraciones muy largas, con lo que las desviaciones en las estimaciones tienden a controlarse con cierta facilidad.

### **Agenda y control del proyecto**

- **Fecha de inicio:** 9/6/2003
- **Fecha de finalización:** 15/9/2003 (lo que daría una duración de unos tres meses, que es lo que se había pactado con Dirección).
- **Hitos de control:** Cada entrega representa un hito de control en sí misma. Además, sirven de punto de referencia para los miembros del equipo, que tienen siempre un objetivo temporal cercano, lo que permite mantener una cierta tensión constantemente. Esto es un arma de doble filo porque no hay que caer en la tentación de infravalorar las duraciones de las tareas porque iría en detrimento del rendimiento del equipo.

### **Camino crítico**

Dado lo ajustado del presupuesto, casi todas las actividades son críticas, por lo que es imprescindible que el jefe de proyecto esté muy al tanto de la evolución de todas las tareas y que actúe rápidamente ante cualquier retraso. También es importante advertir de esta característica del proyecto tanto a la Dirección como al Equipo de desarrollo.

### **Acciones correctoras**

Ante un retraso en una tarea se debe intentar replantear el plan de proyecto para que se sigan cumpliendo los objetivos. Si las razones causantes del retraso justificaran ante Dirección la asignación de nuevos recursos y/o el aplazamiento de la conclusión del proyecto (es decir, si aumentaran el presupuesto) entonces se debería replanificar el proyecto desde ese punto.

Si ninguna de estas dos circunstancias se dieran, se debería plantear eliminar funcionalidades en el resultado final. Para ello se priorizarán las principales tareas y se replanificará atendiendo a estas prioridades. Sin embargo, sólo se deben eliminar tareas poco prioritarias si éstas son críticas, es decir, si su eliminación permite alcanzar la fecha de entrega acordada.