

5

Gestión de proyectos

Objetivos

El objetivo de este capítulo es dar un panorama de la gestión de proyectos de software. Cuando termine de leer este capítulo:

- conocerá las tareas principales de los gestores de proyectos de software.
- comprenderá por qué la naturaleza del software hace más difícil la gestión de proyectos software que la gestión de los proyectos de otras ingenierías.
- comprenderá por qué planificar proyectos es esencial en todos los proyectos de software.
- conocerá la forma en que las representaciones gráficas (gráficos de barras y redes de actividades) son utilizadas por los gestores de proyectos para representar las agendas del proyecto.
- conocerá el proceso de gestión de riesgos y algunos de los riesgos que surgen en los proyectos de software.

Contenidos

- 5.1 Actividades de gestión
- 5.2 Planificación del proyecto
- 5.3 Calendarización del proyecto
- 5.4 Gestión de riesgos

La gestión de proyectos de software es una parte esencial de la ingeniería del software. La buena gestión no puede garantizar el éxito del proyecto. Sin embargo, la mala gestión usualmente lleva al fracaso del proyecto. El software es entregado tarde, los costes son mayores que los estimados y los requerimientos no se cumplen.

Los gestores de software son responsables de la planificación y temporalización del desarrollo de los proyectos. Supervisan el trabajo para asegurar que se lleva a cabo conforme a los estándares requeridos y supervisan el progreso para comprobar que el desarrollo se ajusta al tiempo previsto y al presupuesto. La administración de proyectos de software es necesaria debido a que la ingeniería de software profesional siempre está sujeta a restricciones organizacionales de tiempo y presupuesto. El trabajo del gestor de proyectos de software es asegurar que éstos cumplan dichas restricciones y entregar software que contribuya a las metas de la compañía de desarrollo software.

Los gestores de software hacen el mismo tipo de trabajo que otros gestores. Sin embargo, la ingeniería del software es diferente en varios aspectos a otros tipos, lo que hace a la gestión de software particularmente difícil. Algunas de estas diferencias son las siguientes:

1. ***El producto es intangible.*** El gestor de un proyecto de construcción de un embarcadero o de uno de ingeniería civil puede ver el producto mientras se está desarrollando. Si hay un desfase en calendario, el efecto en el producto es visible de forma obvia: partes de la estructura no están completas. El software es intangible. No se puede ver ni tocar. Los gestores de proyectos de software no pueden ver el progreso. Confían en otros para elaborar la documentación necesaria para revisar el progreso.
2. ***No existen procesos del software estándar.*** En las disciplinas de ingeniería con larga historia, el proceso se prueba y verifica. Para tipos particulares de sistemas, como puentes o edificios, el proceso de ingeniería se comprende bien. Sin embargo, los procesos de software varían notablemente de una organización a otra. A pesar de que la comprensión del proceso del software se ha desarrollado de forma significativa en los últimos años, aún no se puede predecir con certeza cuándo un proceso particular tiende a desarrollar problemas. Esto es especialmente cierto cuando el proyecto software es parte un proyecto de ingeniería de un sistema grande.
3. ***A menudo los proyectos grandes son únicos.*** Por lo general, los proyectos grandes de software son diferentes de proyectos previos. En consecuencia, los gestores, aun cuando cuenten con una amplia experiencia, ésta no es suficiente para anticipar los problemas. Más aún, los rápidos cambios tecnológicos en las computadoras y las comunicaciones hacen parecer obsoleta la experiencia previa. Las lecciones aprendidas en esas experiencias pueden no ser transferibles a los nuevos proyectos.

Debido a estos problemas, no es sorprendente que algunos proyectos de software se retrasen, sobrepasen el presupuesto y se entreguen fuera de tiempo. A menudo, los sistemas de software son nuevos y tecnológicamente innovadores. Frecuentemente los proyectos de ingeniería innovadores (como los nuevos sistemas de transporte) también tienen problemas de temporalización. Dadas las mezclas de dificultades, es notable que muchos proyectos de software sean entregados a tiempo y según lo presupuestado.

La gestión de proyectos de software es un tema amplio y no puede tratarse en un solo capítulo. Por lo tanto, en este capítulo se introduce el tema y se describen tres actividades importantes de gestión: planificación, calendarización de proyectos y gestión de riesgos. Los últimos capítulos (en la Parte 6) tratan otros aspectos de la gestión de software, entre los que se incluyen la gestión de personal, la estimación de los costes de software y la gestión de la calidad.

5.1 Actividades de gestión

Es imposible redactar una descripción estándar del trabajo de un gestor de software. El trabajo difiere enormemente dependiendo de la organización y del producto de software a desarrollar. Sin embargo, en algún momento, muchos gestores son responsables de algunas o de la totalidad de las siguientes actividades:

- Redacción de la propuesta
- Planificación y calendarización del proyecto
- Estimación de costes del proyecto
- Supervisión y revisión del proyecto
- Selección y evaluación del personal
- Redacción y presentación de informes

La primera etapa de un proyecto de software implica redactar una propuesta para realizar ese proyecto. La propuesta describe los objetivos del proyecto y cómo se llevaría a cabo. Por lo general, incluye estimaciones de coste y tiempo y justifica por qué el contrato del proyecto se le debe dar a una organización o a un equipo en particular. La redacción de la propuesta es una tarea crítica, ya que la existencia de muchas organizaciones de software depende de las propuestas aceptadas y los contratos asignados. No existen guías para esta tarea; la redacción de propuestas es una habilidad que se adquiere con la práctica y la experiencia.

La planificación de proyectos se refiere a la identificación de actividades, hitos y entregas de un proyecto. Por lo tanto, se debe bosquejar un plan para guiar el desarrollo hacia las metas del proyecto. La estimación del coste es una actividad relacionada con la estimación de los recursos requeridos para llevar a cabo el plan del proyecto. Este aspecto se tratará con mayor detalle más adelante en este capítulo y en el Capítulo 26.

La supervisión de proyecto es una actividad continua. El gestor debe tener conocimiento del progreso del proyecto y comparar el progreso con los costes actuales y los planificados. Aunque muchas organizaciones tienen mecanismos formales para supervisar, un gestor hábil podría formarse una imagen clara de lo que pasa llevando a cabo una entrevista informal con el personal del proyecto.

La supervisión informal predice problemas importantes del proyecto, y revela dificultades que pueden aparecer. Por ejemplo, las entrevistas diarias con el personal del proyecto pueden exteriorizar un problema en un fallo del software. Más que esperar un informe de atraso del proyecto, el gestor de software podría asignar algún experto para resolver el problema o podría decidir si se vuelve a programar.

Durante un proyecto, es normal tener varias revisiones formales de su gestión. Se hace la revisión completa del progreso y de los desarrollos técnicos del proyecto, y se tiene en cuenta el estado del proyecto junto con los propósitos de la organización que ha encargado el software.

El resultado de una revisión puede dar lugar a la cancelación del proyecto. El tiempo de desarrollo para un proyecto grande de software puede ser de varios años. Durante ese tiempo los objetivos organizacionales tienden obviamente a cambiar. Estos cambios pueden significar que el software ya no se necesita o que los requerimientos originales del proyecto son inadecuados. La gestión puede decidir parar el desarrollo del software o cambiar el proyecto para adecuarlo a los cambios de los objetivos de la organización.

Por lo general, los gestores de proyectos tienen que seleccionar a las personas para trabajar en su proyecto. De forma ideal, habrá personal disponible con habilidades y experiencia

apropiada para trabajar en el proyecto. Sin embargo, en muchos casos, los gestores tienen que establecer un equipo ideal mínimo para el proyecto. Las razones que explican esto son:

1. El presupuesto del proyecto no cubre la contratación de personal con sueldos altos. Se tiene que contratar personal con menos experiencia y menor sueldo.
2. El personal con experiencia apropiada no está disponible dentro o fuera de la organización. Es imposible reclutar nuevo personal para el proyecto. Dentro de la organización, los mejores trabajadores ya se han asignado a otros proyectos.
3. La organización desea desarrollar las habilidades de sus empleados. El personal inexperto puede ser asignado al proyecto para aprender y adquirir experiencia.

El gestor de software tiene que trabajar con estas restricciones al seleccionar al personal del proyecto. Sin embargo, todos estos problemas son probables a menos que exista un miembro del proyecto que cuente con algo de experiencia en el tipo de sistema a desarrollar. Sin esta experiencia, probablemente se cometerán muchos errores pequeños. En el Capítulo 25 se abordará la formación del equipo de desarrollo y la selección del personal.

Los gestores del proyecto son responsables de informar a los clientes y contratistas sobre el proyecto. Tienen que redactar documentos concisos y coherentes que resuman la información crítica de los informes detallados del proyecto. Les debe ser posible presentar esta información durante las revisiones de progreso. En consecuencia, comunicarse efectivamente de forma oral y escrita es una habilidad esencial que un gestor de proyectos debe tener.

5.2 Planificación del proyecto

La gestión efectiva de un proyecto de software depende de planificar completamente el progreso del proyecto. El gestor del proyecto debe anticiparse a los problemas que puedan surgir, así como preparar soluciones a esos problemas. Un plan, preparado al inicio de un proyecto, debe utilizarse como un conductor para el proyecto. Este plan inicial debe ser el mejor posible de acuerdo con la información disponible. Éste evolucionará conforme el proyecto progrese y la información sea mejor.

En la Sección 5.2.1 se describe una estructura para un plan de desarrollo de software. Además de un plan del proyecto, los gestores tienen que preparar otros tipos de planes, los cuales se describen brevemente en la Figura 5.1 y serán tratados con más detalle en otros capítulos del libro.

Plan de calidad	Describe los procedimientos y los estándares de calidad que se utilizarán en un proyecto. Véase el Capítulo 24.
Plan de validación	Describe el enfoque, los recursos y la programación utilizados para la validación del sistema.
Plan de gestión de configuraciones	Describe los procedimientos para la gestión de configuraciones y las estructuras a utilizar. Véase el Capítulo 29.
Plan de mantenimiento	Predice los requerimientos del mantenimiento del sistema, los costes del mantenimiento y el esfuerzo requerido. Véase el Capítulo 27.
Plan de desarrollo del personal	Describe cómo se desarrollan las habilidades y experiencia de los miembros del equipo del proyecto.

Figura 5.1 Tipos de plan.

```

Establecer las restricciones del proyecto
Hacer la valoración inicial de los parámetros del proyecto
Definir los hitos del proyecto y productos a entregar
Mientras el proyecto no se haya completado o cancelado repetir
    Bosquejar la programación en el tiempo del proyecto
    Iniciar actividades acordes con la programación
    Esperar (por un momento)
    Revisar el progreso del proyecto
    Revisar las estimaciones de los parámetros del proyecto
    Actualizar la programación del proyecto
    Renegociar las restricciones del proyecto y los productos a entregar
    Si (surgen problemas) entonces
        Iniciar la revisión técnica y la posible revisión
    fin de si
fin de repetir

```

Figura 5.2
Planificación del
proyecto.

El pseudocódigo que se muestra en la Figura 5.2 describe el proceso de planificación del proyecto para el desarrollo de software. Muestra que la planificación es un proceso iterativo que solamente se completa cuando el proyecto mismo se termina. Conforme la información se hace disponible, el plan debe revisarse regularmente. Las metas globales del negocio son un factor importante que debe considerarse cuando se formula el plan del proyecto. Conforme éstas cambien, serán necesarios cambios en el proyecto.

El proceso de planificación se inicia con una valoración de las restricciones que afectan al proyecto (fecha de entrega requerida, personal disponible, presupuesto global, etcétera). Ésta se lleva a cabo en conjunto con una estimación de los parámetros del proyecto, como su estructura, tamaño y distribución de funciones. Entonces se definen los hitos de progreso y productos a entregar. En ese momento, el proceso entra en un ciclo. Se prepara un calendario para el proyecto y las actividades definidas en el calendario se inician o se continúan. Después de algún tiempo (por lo general 2 o 3 semanas), se revisa el proyecto y se señalan las discrepancias. Debido a que las estimaciones iniciales de los parámetros del proyecto son aproximaciones, el plan siempre deberá actualizarse.

Cuando se dispone de más información, los gestores del proyecto revisan las suposiciones del proyecto y la agenda. Si el proyecto se retrasa, tienen que renegociar con el cliente las restricciones del mismo y las entregas. Si esta renegociación no tiene éxito y no se puede cumplir el calendario, se debe llevar a cabo una revisión técnica. El objetivo de esta revisión es encontrar un enfoque alternativo que se ajuste a las restricciones del proyecto y cumpla con las metas del calendario.

Por supuesto, los gestores de proyectos inteligentes no suponen que todo irá bien. Durante el proyecto siempre surgen problemas en algunas descripciones. Las suposiciones iniciales y el calendario deben ser más bien pesimistas que optimistas. Debe haber suficiente holgura para que las contingencias en el plan, las restricciones del proyecto y los hitos no se tengan que negociar cada vez que se efectúa un ciclo en el plan.

5.2.1 El plan del proyecto

El plan del proyecto fija los recursos disponibles, divide el trabajo y crea un calendario de trabajo. En algunas organizaciones, el plan del proyecto es un único documento que incluye todos los diferentes tipos de planes (Figura 5.1). En otros casos, este plan sólo se refiere al proceso de desarrollo. Otros pueden estar referenciados, pero son proporcionados por separado.

El plan que se describe aquí tiene que ver con el último tipo de plan mencionado. Los detalles de este plan varían dependiendo del tipo de proyecto y de la organización. Sin embargo, muchos planes incluyen las siguientes secciones:

1. **Introducción.** Describe brevemente los objetivos del proyecto y expone las restricciones (por ejemplo, presupuesto, tiempo, etcétera) que afectan a la gestión del proyecto.
2. **Organización del proyecto.** Describe la forma en que el equipo de desarrollo está organizado, la gente involucrada y sus roles en el equipo.
3. **Análisis de riesgo.** Describe los posibles riesgos del proyecto, la probabilidad de que surjan estos riesgos y las estrategias de reducción de riesgos propuestas. La gestión de riesgos se estudia en la Sección 5.4.
4. **Requerimientos de recursos de hardware y software.** Describe el hardware y el software de ayuda requeridos para llevar a cabo el desarrollo. Si es necesario comprar hardware, se deben incluir las estimaciones de los precios y las fechas de entrega.
5. **División del trabajo.** Describe la división del proyecto en actividades e identifica los hitos y productos a entregar asociados con cada actividad. En la Sección 5.2.2 se indican los hitos y productos a entregar.
6. **Programa del proyecto.** Describe las dependencias entre actividades, el tiempo estimado requerido para alcanzar cada hito y la asignación de la gente a las actividades.
7. **Mecanismos de supervisión e informe.** Describe la gestión de informes y cuándo deben producirse, así como los mecanismos de supervisión del proyecto a utilizar.

El plan del proyecto debe revisarse regularmente durante el proyecto. Algunas partes, como el calendario del proyecto, cambiarán frecuentemente; otras serán más estables. Para simplificarlas revisiones, se debe organizar el documento en secciones separadas que permitan su reemplazo de forma individual conforme evoluciona el plan.

5.2.2 Hitos y entregas

Los gestores necesitan información para hacer su trabajo. Como el software es intangible, esta información sólo se puede proveer como documentos que describan el estado del software que se está desarrollando. Sin esta información, es imposible juzgar el progreso y no se pueden actualizar los costes y calendarios.

Cuando se planifica un proyecto, se debe establecer una serie de **hitos** —puntos finales de una actividad del proceso del software—. En cada uno, debe existir una salida formal, como un informe, que se debe presentar al gestor. Los informes de hitos no deben ser documentos amplios. Deben ser informes cortos de los logros en una actividad del proyecto. Los hitos deben representar el fin de una etapa lógica en el proyecto. Los hitos indefinidos como «80 % del código completo» son imposibles de validar y carecen de utilidad para la gestión del proyecto. No podemos validar si se ha llegado a esta etapa debido a que la cantidad de código que se tiene que desarrollar no es precisa.

Una entrega es el resultado del proyecto que se entrega al cliente. De forma general, se entrega al final de una fase principal del proyecto como la especificación, el diseño, etcétera. Como regla general, las entregas son hitos, pero éstos no son necesariamente entregas. Dichos hitos pueden ser resultados internos del proyecto que son utilizados por el gestor del proyecto para verificar el progreso del mismo pero que no se entregan al cliente.

Para establecer los hitos, el proceso del software debe dividirse en actividades básicas con sus salidas asociadas. Por ejemplo, la Figura 5.3 muestra las actividades involucradas en la especificación de requerimientos cuando se utiliza la construcción de prototipos para ayudar

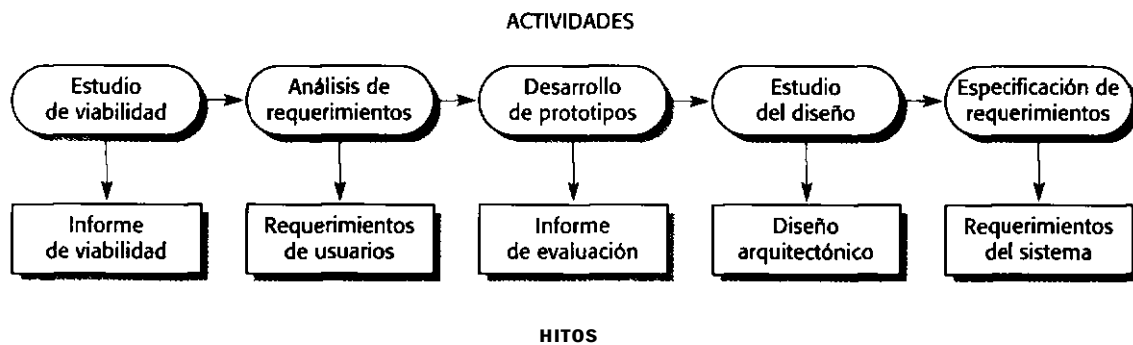


Figura 5.3 Hitos del proceso de especificación de requerimientos.

a validar dichos requerimientos. También se muestran las salidas principales de cada actividad (los hitos del proyecto). Las entregas del proyecto, las cuales son entregadas al cliente, son la definición y especificación de requerimientos.

5.3 Calendarización del proyecto

Ésta es una de las tareas más difíciles para los gestores de proyectos. Los gestores estiman el tiempo y los recursos requeridos para completar las actividades y organizarías en una sucesión coherente. A menos que el proyecto a calendarizar sea similar a otro anterior, las estimaciones previas son una base incierta para la calendarización del nuevo proyecto. La estimación del calendario se complica más por el hecho de que proyectos diferentes pueden utilizar métodos de diseño y lenguajes de implementación diferentes.

Si el proyecto es técnicamente complejo, las estimaciones iniciales casi siempre son optimistas aun cuando los gestores traten de considerar las eventualidades. A este respecto, la calendarización del tiempo para la creación del software no es diferente a la de cualquier otro tipo de proyecto grande y complejo. Los nuevos aeroplanos, los puentes e incluso los nuevos modelos de automóviles se retrasan debido a problemas no anticipados. Por lo tanto, los calendarios se deben actualizar continuamente en la medida que se disponga de mejor información acerca del progreso.

La calendarización del proyecto (véase la Figura 5.4) implica separar todo el trabajo de un proyecto en actividades complementarias y considerar el tiempo requerido para completar dichas actividades. Por lo general, algunas de éstas se llevan a cabo en paralelo. Debemos coordinar estas actividades paralelas y organizar el trabajo para que la mano de obra se utilice

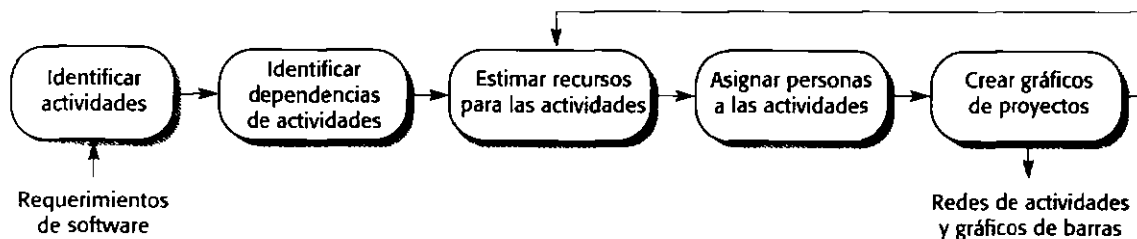


Figura 5.4 El proceso de calendarización del proyecto.

de forma óptima. Deben evitarse situaciones en que el proyecto entero se retrase debido a que no se ha terminado una actividad crítica.

Normalmente, las actividades del proyecto deben durar por lo menos una semana. Hacer subdivisiones más finas significa invertir una cantidad desproporcionada de tiempo en la estimación y revisión de tablas. También es útil asignar una cantidad de tiempo máxima de 8 a 10 semanas para realizar cualquier actividad. Si lleva más tiempo, se deben hacer subdivisiones.

Al estimar la calendarización, los gestores no deben suponer que cada etapa del proyecto estará libre de problemas. Las personas que trabajan en él pueden enfermarse o renunciar, el hardware puede fallar y el software o hardware de soporte puede llegar tarde. Si el proyecto es nuevo y técnicamente complejo, ciertas partes podrían ser más complicadas y llevarían más tiempo del que se estimó originalmente.

Como en los calendarios, los gestores deben estimar los recursos necesarios para completar cada tarea. El recurso principal es el esfuerzo humano que se requiere. Otros recursos pueden ser el espacio en disco requerido en un servidor, el tiempo requerido de hardware especializado, un simulador o el presupuesto para viajes del personal del proyecto. En el Capítulo 26 se trata con mayor detalle este tema.

Una buena regla práctica es estimar como si nada fuera a salir mal, y entonces incrementar la estimación para abarcar los problemas previstos. Con este mismo fin, a la estimación se le debe agregar un factor de contingencia adicional. Este factor extra de contingencia depende del tipo de proyecto, de los parámetros del proceso (fecha de entrega, estándares, etcétera) y de la calidad y experiencia de los ingenieros de software que trabajen en el proyecto. Como regla, para los problemas previstos siempre debe agregarse un 30 % a la estimación original y otro 20 % para cubrir algunas cosas no previstas.

Por lo general, el calendario del proyecto se representa como un conjunto de gráficos que muestran la división del trabajo, las dependencias de las actividades y la asignación del personal. Esto se aborda en la siguiente sección. Por lo general, las herramientas de gestión de software, como Microsoft Project, se utilizan para automatizar la producción de diagramas.

5.3.1 Gráficos de barras y redes de actividades

Los gráficos de barras y las redes de actividades son notaciones gráficas que se utilizan para ilustrar la calendarización del proyecto. Los gráficos de barras muestran quién es responsable de cada actividad y cuándo debe comenzar y finalizar ésta. Las redes de actividades muestran las dependencias entre las diferentes actividades que conforman un proyecto. Los gráficos de barras y las redes de actividades se generan automáticamente a partir de una base de datos de la información del proyecto utilizando una herramienta de gestión de proyectos.

Considere el conjunto de actividades que se muestra en la Figura 5.5. Esta tabla recoge las tareas, la duración e interdependencias de éstas. En ella se observa que la Tarea T3 depende de la Tarea T1. Esto significa que T1 debe completarse antes de que comience T3. Por ejemplo, T1 podría ser la preparación de un diseño de un componente y T3 la implementación de ese diseño. Antes de que se inicie la implementación, el diseño debe estar terminado.

Dadas la dependencia y la duración estimada de las actividades, una red de actividades muestra la sucesión de actividades que deben generarse (véase la Figura 5.6). Esta muestra qué actividades se llevan a cabo en paralelo y cuáles deben ejecutarse en secuencia debido a la dependencia con una actividad previa. Las actividades se representan como rectángulos; los

T1	8	
T2	15	
T3	15	T1 (M1)
T4	10	
T5	10	T2, T4 (M2)
T6	5	T1, T2 (M3)
T7	20	T1 (M1)
T8	25	T4 (M5)
T9	15	T3, T6 (M4)
T10	15	T5, T7 (M7)
T11	7	T9 (M6)
T12	10	T11 (M8)

Figura 5.5 Duración y dependencias de las tareas.

hitos y las entregas se muestran con esquinas redondeadas. Las fechas en este gráfico muestran la fecha de inicio de la actividad (están escritas en estilo británico, en el cual el día precede al mes). La red se debe leer de izquierda a derecha y de arriba abajo.

En la herramienta de gestión de proyecto utilizada para elaborar este gráfico, todas las actividades deben terminar en hitos. Una comienza cuando su hito precedente (puede depender de varias actividades) se ha alcanzado. Por lo tanto, en la tercera columna la Figura 5.5 se muestra el hito correspondiente (por ejemplo, M5) que se alcanza cuando las tareas en esa columna terminan (véase la Figura 5.6).

Antes de que se pueda pasar de un hito a otro, todas las trayectorias deben completarse. Por ejemplo, la tarea T9, que se muestra en la Figura 5.6, no se puede iniciar hasta que las tareas T3 y T6 se terminen.

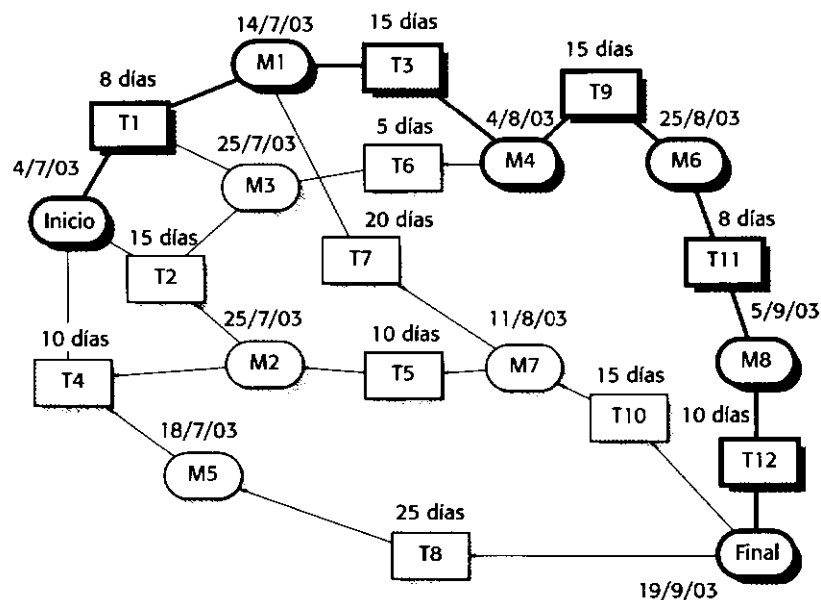


Figura 5.6 Una red de actividades.

El tiempo mínimo requerido para finalizar el proyecto se estima teniendo en cuenta la trayectoria más larga en la red de actividades (el camino crítico). En este caso, es 11 semanas de tiempo transcurrido o 55 días laborales. En la Figura 5.6 el camino crítico se muestra como una sucesión de rectángulos más resaltados. El calendario completo del proyecto depende de dicho camino. Cualquier aplazamiento al completar una actividad crítica provoca retraso en el proyecto, ya que las actividades siguientes no pueden comenzar hasta que se finalice la actividad retrasada.

Sin embargo, los retrasos de las actividades que no están ligadas al camino crítico no provocan necesariamente un aplazamiento en todo el calendario. Mientras los retrasos no se propaguen a estas actividades como para que se exceda el tiempo total del camino crítico, el proyecto no se verá afectado. Por ejemplo, si T8 se retrasa, esto no afectará a la fecha final de terminación del proyecto puesto que no se encuentra sobre el camino crítico. El gráfico de barras (véase la Figura 5.7) del proyecto muestra la holgura de los posibles retrasos como barras sombreadas.

Los gestores también utilizan las redes de actividades para asignar los trabajos en el proyecto. Dichas redes pueden dar una percepción de la dependencia de las actividades que de forma intuitiva no son obvias. Es posible modificar el diseño del sistema de tal forma que se acorte al camino crítico. El calendario del proyecto se puede acortar debido a que se reduce la cantidad de tiempo de espera para finalizar las actividades.

Inevitablemente, las agendas del proyecto iniciales serán incorrectas. A medida que el proyecto se desarrolla, las estimaciones deben ser comparadas con los tiempos reales. Esta comparación puede ser utilizada como base para una revisión de la agenda de las siguientes partes del proyecto. Cuando se muestran los tiempos reales, debemos revisar el gráfico de

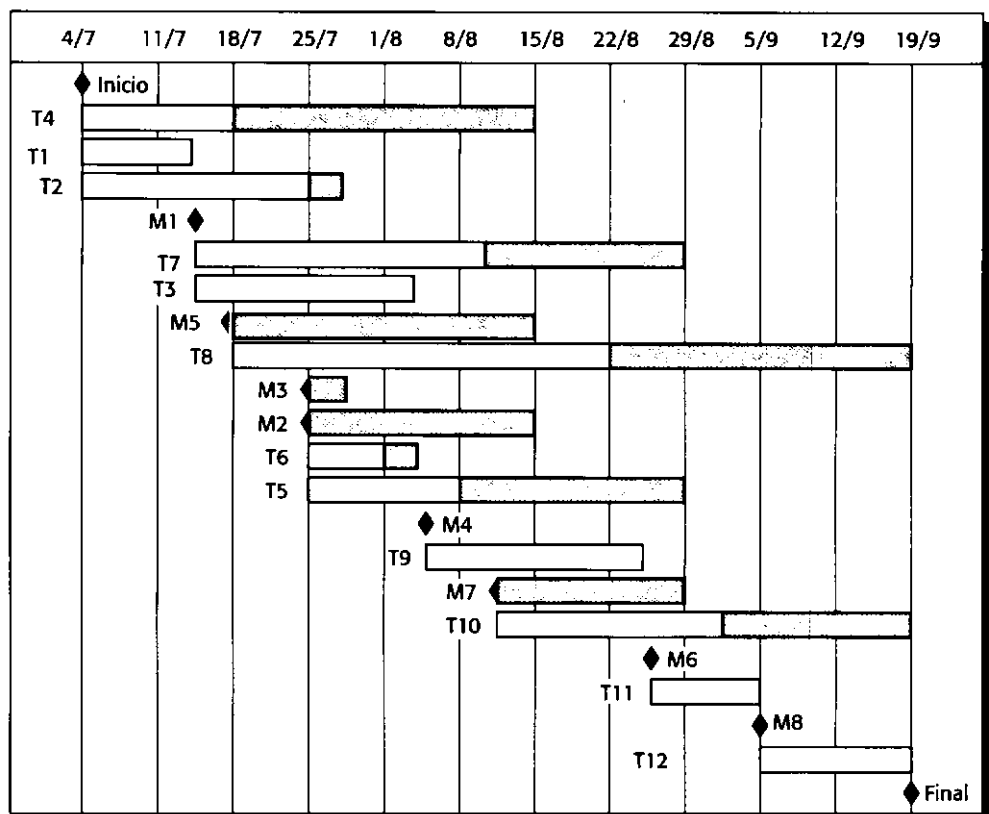


Figura 5.7 Gráfico de barras de actividades.

actividades. Las actividades siguientes del proyecto pueden ser reorganizadas para reducir la duración del camino crítico.

La Figura 5.7 es una forma alternativa de representar la información del calendario del proyecto. Es un gráfico de barras (algunas veces llamado gráfico de Gantt, su inventor) que muestra el calendario de un proyecto y las fechas iniciales y finales de las actividades. Algunas de las actividades que se muestran en la Figura 5.7 son seguidas por una barra sombreada cuya longitud es calculada por una herramienta de calendarización. Esto muestra que existe alguna flexibilidad en las fechas de terminación de estas actividades. Si alguna de éstas no se completa a tiempo, el camino crítico no se verá afectado hasta el final del periodo marcado por la barra sombreada. Las actividades que caen sobre dicho camino no tienen margen de error y se distinguen por no tener asociada una barra sombreada.

Además de considerar la calendarización, los gestores de proyectos también deben tener en cuenta la asignación de recursos y, en particular, la asignación de personal a las actividades del proyecto. Esta asignación puede ser introducida también desde las herramientas de gestión de proyectos, y los gráficos de barras generados muestran el personal del proyecto (Figura 5.8). Las personas no tienen por qué estar asignadas al proyecto en todo momento. Durante diferentes periodos pueden estar de vacaciones, trabajando en otros proyectos, en cursos de formación o realizando otras actividades.

Por lo general, las grandes organizaciones emplean varios especialistas para que trabajen en el proyecto cuando sea necesario. En la Figura 5.8 se puede ver que Mary y Jim son especialistas que sólo trabajan en una actividad del proyecto. Esto puede provocar problemas en la calendarización. Si un proyecto se retrasa mientras un especialista trabaja en él, puede causar efectos contundentes en los otros proyectos. Estos proyectos serán retrasados ya que el especialista no estará disponible.

5.4 Gestión de riesgos

Una tarea importante del gestor de proyectos es anticipar los riesgos que podrían afectar a la programación del proyecto o a la calidad del software a desarrollar y emprender acciones para

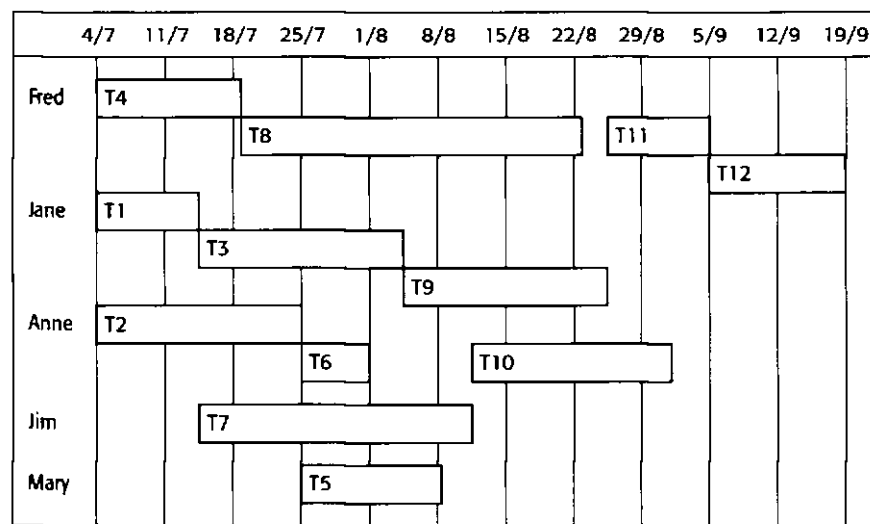


Figura 5.6 Gráfico de asignación de personal/tiempo.

evitar esos riesgos. Los resultados de este análisis de riesgos se deben documentar a lo largo del plan del proyecto junto con el análisis de consecuencias cuando el riesgo ocurra. Identificar éstos y crear planes para minimizar sus efectos en el proyecto se llama gestión de riesgos (Hall, 1998; Ould, 1999).

De forma simple, se puede concebir un riesgo como una probabilidad de que una circunstancia adversa ocurra. Los riesgos son una amenaza para el proyecto, para el software que se está desarrollando y para la organización. Estas categorías de riesgos se definen como se muestra a continuación:

1. **Riesgos del proyecto.** Éstos afectan la calendarización o los recursos del proyecto. Un ejemplo podría ser la pérdida de un diseñador experimentado.
2. **Riesgos del producto.** Éstos afectan a la calidad o al rendimiento del software que se está desarrollando. Un ejemplo podría ser que el rendimiento en un componente que hemos comprado sea menor que el esperado.
3. **Riesgos del negocio.** Estos afectan a la organización que desarrolla o suministra el software. Por ejemplo, que un competidor introduzca un nuevo producto es un riesgo de negocio.

Por supuesto, estos tipos no son exclusivos entre sí. Si un programador experto abandona el proyecto, esto es un riesgo para el proyecto (debido a que la entrega del sistema se puede retrasar), para el producto (debido a que un sustituto puede no ser tan experto y cometer muchos errores) y para el negocio (debido a que esa experiencia puede no contribuir a negocios futuros).

Los riesgos que pueden afectar a un proyecto dependen del propio proyecto y del entorno organizacional donde se desarrolla. Sin embargo, muchos riesgos son universales. La Figura 5.9 muestra algunos de estos riesgos.

La gestión de riesgos es importante particularmente para los proyectos de software debido a las incertidumbres inherentes con las que se enfrentan muchos proyectos. Estas incerti-

Rotación de personal	Proyecto	Personal con experiencia abandona el proyecto antes de que finalice.
Cambio de gestión	Proyecto	Habrà un cambio de gestión organizacional con diferentes prioridades.
No disponibilidad del hardware	Proyecto	El hardware esencial para el proyecto no será entregado a tiempo.
Cambio de requerimientos	Proyecto y producto	Habrà más cambios en los requerimientos de lo esperado.
Retrasos en la especificación	Proyecto y producto	Las especificaciones de las interfaces esenciales no estarán a tiempo.
Subestimación del tamaño	Proyecto y producto	El tamaño del sistema se ha subestimado.
Bajo rendimiento de la herramienta CASE	Producto	Las herramientas CASE que ayudan al proyecto no tienen el rendimiento esperado.
Cambio de tecnología	Negocio	Un producto competitivo se pone en venta antes de que el sistema se complete.
Competencia del producto	Negocio	La tecnología fundamental sobre la que se construirà el sistema se sustituye por nueva tecnología.

Figura 5.9 Posibles riesgos del software.

dumbres son el resultado de los requerimientos ambiguamente definidos, las dificultades en la estimación de tiempos y los recursos para el desarrollo del software, la dependencia en las habilidades individuales, y los cambios en los requerimientos debidos a los cambios en las necesidades del cliente. Es preciso anticiparse a los riesgos: comprender el impacto de éstos en el proyecto, en el producto y en el negocio, y considerar los pasos para evitarlos. En el caso de que ocurran, se deben crear planes de contingencia para que sea posible aplicar acciones de recuperación.

La Figura 5.10 muestra el proceso de gestión de riesgos. Éste comprende varias etapas:

1. **identificación de riesgos.** Identificar los posibles riesgos para el proyecto, el producto y los negocios.
2. **Análisis de riesgos.** Valorar las probabilidades y consecuencias de estos riesgos.
3. **Planificación de riesgos.** Crear planes para abordar los riesgos, ya sea para evitarlos o minimizar sus efectos en el proyecto.
4. **Supervisión de riesgos.** Valorar los riesgos de forma constante y revisar los planes para la mitigación de riesgos tan pronto como la información de los riesgos esté disponible.

El proceso de gestión de riesgos, como otros de planificación de proyectos, es un proceso iterativo que se aplica a lo largo de todo el proyecto. Una vez que se genera un conjunto de planes iniciales, se supervisa la situación. En cuanto surja más información acerca de los riesgos, éstos deben analizarse nuevamente y se deben establecer nuevas prioridades. La prevención de riesgos y los planes de contingencia se deben modificar tan pronto como surja nueva información de los riesgos.

Los resultados del proceso de gestión de riesgos se deben documentar en un plan de gestión de riesgos. Éste debe incluir un estudio de los riesgos a los que se enfrenta el proyecto, un análisis de éstos y los planes requeridos para su gestión. Si es necesario, puede incluir algunos resultados de la gestión de riesgos; por ejemplo, planes específicos de contingencia que se activan si aparecen dichos riesgos.

5.4.1 Identificación de riesgos

Ésta es la primera etapa de la gestión de riesgos. Comprende el descubrimiento de los posibles riesgos del proyecto. En principio, no hay que valorarlos o darles prioridad en esta etapa aunque, en la práctica, por lo general no se consideran los riesgos con consecuencias menores o con baja probabilidad.

Esta identificación se puede llevar a cabo a través de un proceso de grupo utilizando un enfoque de tormenta de ideas o simplemente puede basarse en la experiencia. Para ayudar al pro-

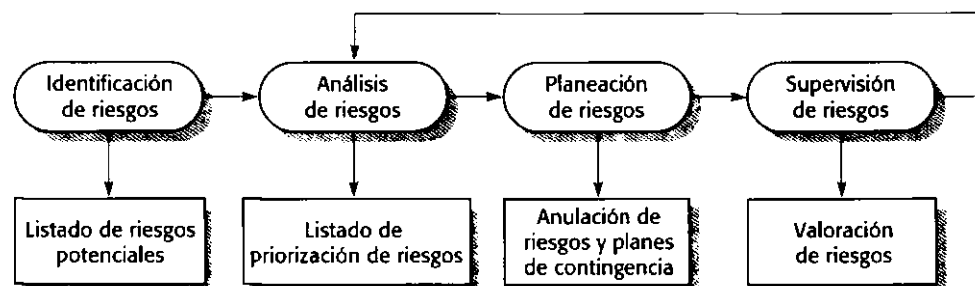


Figura 5.10
El proceso de
gestión de riesgos.

ceso, se utiliza una lista de posibles tipos de riesgos. Hay al menos seis tipos de riesgos que pueden aparecer:

1. **Riesgos de tecnología.** Se derivan de las tecnologías de software o de hardware utilizadas en el sistema que se está desarrollando.
2. **Riesgos de personal.** Riesgos asociados con las personas del equipo de desarrollo.
3. **Riesgos organizacionales.** Se derivan del entorno organizacional donde el software se está desarrollando.
4. **Riesgos de herramientas.** Se derivan de herramientas CASE y de otro software de apoyo utilizado para desarrollar el sistema.
5. **Riesgos de requerimientos.** Se derivan de los cambios de los requerimientos del cliente y el proceso de gestionar dicho cambio.
6. **Riesgos de estimación.** Se derivan de los estimados administrativos de las características del sistema y los recursos requeridos para construir dicho sistema.

La Figura 5.11 proporciona algunos ejemplos de riesgos posibles en cada una de estas categorías. El resultado de este proceso debe ser una larga lista de riesgos que podrían presentarse y afectar al producto, al proceso o al negocio.

5.4.2 Análisis de riesgos

Durante este proceso, se considera por separado cada riesgo identificado y se decide acerca de la probabilidad y la seriedad del mismo. No existe una forma fácil de hacer esto —recae en la opinión y experiencia del gestor del proyecto—. No se hace una valoración con números precisos sino en intervalos:

- La probabilidad del riesgo se puede valorar como muy bajo (< 10%), bajo (10-25%), moderado (25-50%), alto (50-75%) o muy alto (>75%).
- Los efectos del riesgo pueden ser valorados como catastrófico, serio, tolerable o insignificante.

Tecnología	La base de datos que se utiliza en el sistema no puede procesar muchas transacciones por segundo como se esperaba. Los componentes de software que deben reutilizarse contienen defectos que limitan su funcionalidad.
Personal	Es imposible reclutar personal con las habilidades requeridas para el proyecto. El personal clave está enfermo y no disponible en momentos críticos. La capacitación solicitada para el personal no está disponible.
Organizacional	La organización se reestructura de tal forma que una gestión diferente se responsabiliza del proyecto. Los problemas financieros de la organización fuerzan a reducciones en el presupuesto del proyecto.
Herramientas	Es ineficiente el código generado por las herramientas CASE. Las herramientas CASE no se pueden integrar.
Requerimientos	Se proponen cambios en los requerimientos que requieren rehacer el diseño. Los clientes no comprenden el impacto de los cambios en los requerimientos.
Estimación	El tiempo requerido para desarrollar el software está subestimado. La tasa de reparación de defectos está subestimada. El tamaño del software está subestimado.

Figura 5.11 Riesgos y tipos de riesgos.

El resultado de este proceso de análisis se debe colocar en una tabla, la cual debe estar ordenada según la seriedad del riesgo. La Figura 5.12 ilustra esto para los riesgos identificados en la Figura 5.11. Obviamente, aquí es arbitraria la valoración de la probabilidad y seriedad. En la práctica, para hacer esta valoración se necesita información detallada del proyecto, el proceso, el equipo de desarrollo y la organización.

Por supuesto, tanto la probabilidad como la valoración de los efectos de un riesgo cambian conforme se disponga de mayor información acerca del riesgo y los planes de gestión del mismo se implementen. Por lo tanto, esta tabla se debe actualizar durante cada iteración del proceso de riesgos.

Una vez que los riesgos se hayan analizado y clasificado, se debe discernir cuáles son los más importantes que se deben considerar durante el proyecto. Este discernimiento debe depender de una combinación de la probabilidad de aparición del riesgo y de los efectos del mismo. En general, siempre se deben tener en cuenta todos los riesgos catastróficos, así como todos los riesgos serios que tienen más que una moderada probabilidad de ocurrir.

Boehm (Boehm, 1988) recomienda identificar y supervisar los «10 riesgos más altos», pero este número parece demasiado arbitrario. El número exacto de riesgos a supervisar debe depender del proyecto. Pueden ser cinco o 15. No obstante, el número apropiado debe ser manejable. Un número muy grande de riesgos requiere obtener mucha información. De los riesgos identificados en la Figura 5.12, conviene considerar los ocho que tienen consecuencias serias o catastróficas.

Los problemas financieros de la organización fuerzan a reducir el presupuesto del proyecto.	Baja	Catastrófico
Es imposible reclutar personal con las habilidades requeridas para el proyecto.	Alta	Catastrófico
El personal clave está enfermo y no disponible en momentos críticos.	Moderada	Serio
Los componentes de software que deben reutilizarse contienen defectos que limitan su funcionalidad.	Moderada	Serio
Se proponen cambios en los requerimientos que requieren rehacer el diseño.	Moderada	Serio
La organización se reestructura de tal forma que cambia el grupo de gestión.	Alta	Serio
La base de datos que se utiliza en el sistema no puede procesar muchas transacciones por segundo como se esperaba.	Moderada	Serio
El tiempo requerido para desarrollar el software está subestimado.	Alta	Serio
Las herramientas CASE no se pueden integrar.	Alta	Tolerable
Los clientes no comprenden el impacto de los cambios en los requerimientos.	Moderada	Tolerable
La capacitación solicitada para el personal no está disponible.	Moderada	Tolerable
La tasa de reparación de defectos está subestimada.	Moderada	Tolerable
El tamaño del software está subestimado.	Alta	Tolerable
Es ineficiente el código generado por las herramientas CASE.	Moderada	insignificante

Figura 5.12 Análisis de riesgos.

5.4.3 Planificación de riesgos

El proceso de planificación de riesgos considera cada uno de los riesgos clave que han sido identificados, así como las estrategias para gestionarlos. Otra vez, no existe un proceso sencillo que nos permita establecer los planes de gestión de riesgos. Depende del juicio y de la experiencia del gestor del proyecto. La Figura 5.13 muestra posibles estrategias para los riesgos que han sido identificados en la Figura 5.12. Estas estrategias seguidas pueden dividirse en tres categorías.

1. **Estrategias de prevención.** Siguiendo estas estrategias, la probabilidad de que el riesgo aparezca se reduce. Un ejemplo de este tipo de estrategias es la estrategia de evitación de defectos en componentes de la Figura 5.13.
2. **Estrategias de minimización.** Siguiendo estas estrategias se reducirá el impacto del riesgo. Un ejemplo de esto es la estrategia frente a enfermedad del personal de la Figura 5.13.
3. **Planes de contingencia.** Seguir estas estrategias es estar preparado para lo peor y tener una estrategia para cada caso. Un ejemplo de este tipo de estrategia es el mostrado en la Figura 5.13 con la estrategia para problemas financieros.

Puede verse aquí una analogía con las estrategias utilizadas en sistemas críticos para asegurar fiabilidad, protección y seguridad. Básicamente, es mejor usar una estrategia para evitar el riesgo. Si esto no es posible, utilizar una para reducir los efectos serios de los riesgos. Finalmente, tener estrategias para reducir el impacto del riesgo en el proyecto y en el producto.

5.4.4 Supervisión de riesgos

La supervisión de riesgos normalmente valora cada uno de los riesgos identificados para decidir si éste es más o menos probable y si han cambiado sus efectos. Por supuesto, esto no se puede observar de forma directa, por lo que se tienen que buscar otros factores para dar indi-

Problemas financieros de la organización	Preparar un documento breve para el gestor principal que muestre que el proyecto hace contribuciones muy importantes a las metas del negocio.
Problemas de reclutamiento	Alertar al cliente de las dificultades potenciales y los posibles retrasos, investigar la compra de componentes.
Enfermedad de personal	Reorganizar el equipo de tal forma que haya solapamiento en el trabajo y las personas comprendan el de los demás.
Componentes defectuosos	Reemplazar los componentes defectuosos con los comprados de fiabilidad conocida.
Cambios de los requerimientos	Rastrear la información para valorar el impacto de los requerimientos, maximizar la información oculta en ellos.
Reestructuración organizacional	Preparar un documento breve para el gestor principal que muestre que el proyecto hace contribuciones muy importantes a las metas del negocio.
Rendimiento de la base de datos	Investigar la posibilidad de comprar una base de datos de alto rendimiento.
Tiempo de desarrollo subestimado	Investigar los componentes comprados y la utilización de un generador de programas.

Figura 5.13 Estrategia de gestión de riesgos.

cios de la probabilidad del riesgo y sus efectos. Obviamente, estos factores dependen de los tipos de riesgo. La Figura 5.14 da algunos ejemplos de los factores que ayudan en la valoración de estos tipos de riesgos.

La supervisión de riesgos debe ser un proceso continuo y, en cada revisión del progreso de gestión, cada uno de los riesgos clave debe ser considerado y analizado por separado.

Tecnología	Entrega retrasada del hardware o de la ayuda al software, muchos problemas tecnológicos reportados.
Personal	Baja moral del personal, malas relaciones entre los miembros del equipo, disponibilidad de empleo.
Organizacional	Chismorreos organizacionales, falta de acciones por el gestor principal.
Herramientas	Rechazo de los miembros del equipo para utilizar herramientas, quejas acerca de las herramientas CASE, peticiones de estaciones de trabajo más potentes.
Requerimientos	Peticiones de muchos cambios en los requerimientos, quejas del cliente.
Estimación	Fracaso en el cumplimiento de los tiempos acordados, y en la eliminación de defectos reportados.

Figura 5.14 Factores de riesgo.

PUNTOS CLAVE

- Es esencial una buena gestión de proyectos de software para que los proyectos de ingeniería de software se desarrollen a tiempo y según presupuesto.
- La gestión de proyectos de software es diferente a la gestión de otro tipo de ingenierías. El software es intangible. Los proyectos pueden ser nuevos o innovadores, por lo que no existe un conjunto de experiencias para guiar su gestión. El proceso del software no se comprende del todo.
- Los gestores de software tienen diversos papeles. Sus actividades más significativas son la planificación, estimación y calendarización de los proyectos. La planificación y la estimación son procesos iterativos. Tienen continuidad a lo largo del proyecto. En cuanto se tenga más información, se deben revisar los planes y calendarios.
- Un hito de un proyecto es el resultado predecible de una actividad en el que se debe presentar un informe del progreso a la gestión. Los hitos ocurren de forma frecuente en un proyecto de software. Una entrega es un hito que se entrega al cliente del proyecto.
- La calendarización de proyectos implica la creación de varias representaciones gráficas de partes del plan del proyecto. Éstas incluyen redes de actividades que muestran las interrelaciones de las actividades del proyecto y gráficos de barras que muestran la duración de dichas actividades.
- Se deben identificar y valorar los riesgos mayores del proyecto para establecer su probabilidad y consecuencias para éste. En cuanto a los riesgos más probables y potencialmente serios, se deben hacer planes para anularlos, gestionarlos o tratarlos. Estos riesgos se deben analizar de manera explícita en cada reunión del progreso del proyecto.

LECTURAS COMPLEMENTARIAS

Waltzing with Bears: Managing Risk on Software Projects. Una introducción muy práctica y fácil de leer sobre riesgos y su gestión. (T. DeMarco y T. Lister, 2003, Dorset House.)

Managing Software Quality and Business Risk. El Capítulo 3 de este libro es sencillamente el mejor estudio de riesgos que se haya visto. El libro se orienta a éstos y es probablemente el mejor libro de este tema disponible en el momento de esta redacción. (M. Ould, 1999, John Wiley and Sons.)

The Mythical Man Month. Los problemas de la gestión de software no han cambiado desde los años 60 y éste es uno de los mejores libros sobre el tema. Una recopilación interesante y legible de la gestión de uno de los primeros grandes proyectos de software, el sistema operativo IBM OS/360. La segunda edición incluye otros artículos clásicos de Brooks. (F. P. Brooks, 1975, Addison-Wesley.)

Software Project Survival Guide. Una explicación muy pragmática sobre gestión de software, pero que contiene unos buenos consejos. Es fácil de leer y entender. (S. McConneil, 1998, Microsoft Press.)

Véase la Parte 6 para otras lecturas sobre gestión.

EJERCICIOS

- 5.1** Explique por qué la intangibilidad de los sistemas de software plantea problemas para la gestión de proyectos de software.
- 5.2** Explique por qué los mejores programadores no siempre son los mejores gestores de software. La respuesta puede tener como base la lista de actividades de gestión dadas en la Sección 5.1.
- 5.3** Explique por qué el proceso de planificación de proyectos es iterativo y por qué un plan se debe revisar continuamente durante el proyecto de software.
- 5.4** Explique brevemente el propósito de cada una de las secciones en un plan de proyecto de software.
- 5.5** ¿Cuál es la diferencia fundamental entre un hito y una entrega?
- 5.6** La Figura 5.15 muestra un conjunto de actividades, duraciones y dependencias. Diseñe una red de actividades y un gráfico de barras que muestren la programación del proyecto.
- 5.7** La Figura 5.5 señala la duración de las tareas para las actividades del proyecto de software. Suponga que hay un serio retraso no anticipado y que en lugar de requerir 10 días, la tarea T5 requiere 40 días. Revise la red de actividades resultante, resaltando el nuevo camino crítico. Diseñe un nuevo gráfico de barras que muestre cómo se podría reorganizar el proyecto.
- 5.8** Utilizando las instancias referidas en la literatura para los problemas en los proyectos, haga una lista de las dificultades de gestión en esos proyectos de calendarización fallidos. (Comience con el libro de Brooks, sugerido en la sección de lecturas complementarias.)
- 5.9** Además de los riesgos que se muestran en la Figura 5.11, identifique otros seis posibles riesgos en los proyectos de software.

T1	10	
T2	15	T1
T3	10	T1, T2
T4	20	
T5	10	
T6	15	T3, T4
T7	20	T3
T8	35	T7
T9	15	T6
T10	5	T5, T9
T11	10	T9
T12	20	T10
T13	35	T3, T4
T14	10	T8, T9
T15	20	T12, T14
T16	10	T15

Figura 5.15
Duración y
dependencias
de las tareas.

- 5.10** Los contratos de precio prefijado, donde el contratista ofrece un precio fijo para completar el sistema, pueden ser utilizados para traspasar los riesgos del proyecto del cliente al contratista. Si algo va mal, el contratista asumirá la diferencia. Indique de qué modo el uso de contratos puede incrementar la probabilidad de la aparición de riesgos.
- 5.11** Su jefe le ha solicitado que entregue un software en un tiempo que sólo puede ser posible cumplir preguntando al equipo del proyecto si desea trabajar horas extras sin pago alguno. Todos los miembros del equipo tienen hijos pequeños. Comente si debería aceptar esta petición de su jefe o si debería persuadir al equipo para dar su tiempo a la organización más que a sus familias. ¿Qué factores podrían ser significativos en la decisión?
- 5.12** Como programador, se le ofrece un ascenso como gestor de proyecto, pero su sensación es que puede tener una contribución más efectiva en un papel técnico que en uno administrativo. Comente cuándo debería aceptar ese ascenso.

Capítulo 6	Requerimientos del software
Capítulo 7	Proceso de la ingeniería de requerimientos
Capítulo 8	Modelos de sistemas
Capítulo 9	Especificación de sistemas críticos
Capítulo 10	Especificación formal

Tal vez el principal problema al que nos enfrentamos en el desarrollo de sistemas grandes y complejos es el de la ingeniería de requerimientos. Ésta trata de establecer lo que el sistema debe hacer, sus propiedades emergentes deseadas y esenciales, y las restricciones en el funcionamiento del sistema y los procesos de desarrollo del software. Por lo tanto puede considerar la ingeniería de requerimientos como el proceso de comunicación entre los clientes y usuarios del software y los desarrolladores del mismo.

La ingeniería de requerimientos no es simplemente un proceso técnico. Los requerimientos del sistema están influenciados por las preferencias, aversiones y prejuicios de los usuarios y por cuestiones políticas y organizacionales. Éstas con características fundamentales humanas, y las nuevas tecnologías, como los casos de uso, los escenarios y los métodos formales, no nos ayudan mucho en la resolución de estos problemas espinosos.

Los capítulos de esta sección se dividen en dos clases - en los Capítulos 6 y 7 introduzco los fundamentos de la ingeniería de requerimientos, y en los Capítulos 8 a 10 describo los modelos y técnicas utilizados en el proceso de ingeniería de requerimientos. Más específicamente:

1. El Capítulo 6 trata sobre los requerimientos del software y los documentos de requerimientos. Se considera qué se entiende por requerimiento, los diferentes tipos de requerimientos y cómo estos requerimientos se organizan en un documento de especificación de requerimientos. En este tema se introduce el segundo caso de estudio, un sistema de biblioteca.
2. El Capítulo 7 se centra en las actividades en el proceso de ingeniería de requerimientos; cómo los estudios de viabilidad siempre deben ser parte de la ingeniería de requerimientos, de las técnicas para la obtención y análisis de requerimientos, y de la validación de requerimientos. Debido a que los requerimientos inevitablemente cambian, también se aborda el importante tema de la gestión de requerimientos.
3. El Capítulo 8 describe los tipos de modelos de sistemas que se pueden desarrollar en el proceso de ingeniería de requerimientos. Éstos proporcionan una descripción más detallada a los desarrolladores del sistema. Aquí el énfasis está en el modelado orientado a objetos pero también incluyo una descripción de los diagramas de flujo de datos. Se considera que éstos son intuitivos y útiles, especialmente para darle una imagen de cómo la información es procesada por un sistema desde el principio hasta el final.
4. El énfasis en los Capítulos 9 y 10 está en la especificación de los sistemas críticos. El Capítulo 9 aborda la especificación de las propiedades de confiabilidad emergentes. Describe los enfoques dirigidos por riesgos y asuntos de la especificación de la seguridad, la habilidad y la protección. En el Capítulo 10, se introducen las técnicas de especificación formal. Los métodos formales han tenido menos impacto del que se predijo, pero cada vez se utilizan más en la especificación de la seguridad y de los sistemas de misión críticos. Aborda tanto los enfoques algebraicos como los basados en modelos en este capítulo.