1

**Requerimientos**

Las tareas de la ingeniería de requerimientos llevarán a la comprensión de cuál será el efecto que tendrá el software en el negocio, qué es lo que quiere el cliente y cómo interactuarán los usuarios finales con el software. Es importante porque diseñar y construir un elegante programa de cómputo que resuelva el problema equivocado no satisface las necesidades de nadie. A grandes rasgos, se compone de la *concepción*, tarea que define el alcance y la naturaleza del problema que se va a resolver. Va seguida de la *indagación*, labor que ayuda a los participantes a definir lo que se requiere. Después sigue la *elaboración*, donde se refinan y modifican los requerimientos básicos. Su objetivo final es proporcionar a todas las partes un entendimiento escrito del problema.

La ingeniería de requerimientos debe adaptarse a las necesidades del proceso, del proyecto, del producto y de las personas que hacen el trabajo. Incluye siete tareas diferentes:

1. Concepción: se establece el entendimiento básico del problema, las personas que quieren una solución, la naturaleza de la solución que se desea, así como la eficacia de la comunicación y colaboración preliminares entre los otros participantes y el equipo de software.
2. Indagación: preguntar al cliente, a los usuarios y a otras personas cuáles son los objetivos para el sistema o producto, qué es lo que va a lograrse, cómo se ajusta el sistema o producto a las necesidades del negocio y, finalmente, cómo va a usarse el sistema o producto en las operaciones cotidianas.

* Problemas de alcance: está mal definida o los clientes o usuarios finales especifican detalles técnicos innecesarios.
* Problemas de entendimiento: Los clientes o usuarios no están completamente seguros de lo que se necesita, no comprenden las limitaciones de su ambiente de computo, tienen problemas para comunicar las necesidades al ingeniero de sistemas.
* Problemas de volatilidad. Los requerimientos cambian con el tiempo.

1. Elaboración: desarrollar un modelo refinado de los requerimientos que identifique distintos aspectos de la función del software, su comportamiento e información.
2. Negociación: Se pide a clientes, usuarios y otros participantes que ordenen sus requerimientos según su prioridad y que después analicen los conflictos. Un enfoque iterativo que da prioridad a los requerimientos, se evalúa su costo y riesgo, y se enfrentan los conflictos internos.
3. Especificación. Una especificación puede ser un documento escrito, un conjunto de modelos gráficos, un modelo matemático formal, un conjunto de escenarios de uso, un prototipo o cualquier combinación de éstos.
4. Validación: La calidad de los productos del trabajo de los requerimientos se evalúa. La validación de los requerimientos analiza la especificación a fin de garantizar que todos ellos han sido enunciados sin ambigüedades; participan ingenieros de software, clientes, usuarios y otros participantes.
5. Administración de los requerimientos: conjunto de actividades que ayudan al equipo del proyecto a identificar, controlar y dar seguimiento a los requerimientos y a sus cambios en cualquier momento del desarrollo del proyecto.

**Bases**

* Identificación de los participantes: “Cualquier persona que se beneficie en forma directa o indirecta del sistema en desarrollo”
* Reconocer los múltiples puntos de vista: A medida que se recaba información procedente de múltiples puntos de vista, los requerimientos que surjan tal vez sean inconsistentes o estén en conflicto uno con otro. Debe clasificarse toda la información de los participantes (incluso los requerimientos inconsistentes y conflictivos) en forma que permita a quienes toman las decisiones escoger para el sistema un conjunto de requerimientos que tenga coherencia interna.
* Trabajar hacia la colaboración: identificar las áreas de interés común y las de conflicto o incongruencia.
* Hacer las primeras preguntas: Las preguntas que se hacen en la concepción del proyecto deben estar “libres del contexto”. El primer conjunto de ellas se centra en el cliente y en otros participantes, en las metas y beneficios generales. Las preguntas siguientes permiten entender mejor el problema y hacen que el cliente exprese sus percepciones respecto de la solución. Las preguntas finales se centran en la eficacia de la actividad de comunicación en sí.

**Indagación de los requerimientos**

Combina elementos de la solución de problemas, elaboración, negociación y especificación.

* Recabación de los requerimientos en forma colaborativa: Un “facilitador” (cliente, desarrollador o participante externo) controla la reunión. La meta es identificar el problema, proponer elementos de la solución, negociar distintos

enfoques y especificar un conjunto preliminar de requerimientos de la solución en una atmósfera que favorezca el logro de la meta. Se distribuye la solicitud de producto a todos los asistentes. Mientras se revisa la solicitud del producto antes de la reunión, se pide a cada asistente que elabore una lista de objetos que sean parte del ambiente que rodeará al sistema. Cada asistente desarrollará una lista de restricciones. Una vez que se presentan las listas individuales acerca de un área temática, el grupo crea una lista, eliminando las entradas redundantes o agregando ideas nuevas que surjan durante el análisis, pero no se elimina ninguna. Sigue el análisis, coordinado por el facilitador.

* Despliegue de la función de calidad: técnica de administración de la calidad que traduce las necesidades del cliente en requerimientos técnicos para el software. Se concentra en maximizar la satisfacción del cliente a partir del proceso de ingeniería del software.

1. Requerimientos normales: Objetivos y metas que se establecen para un producto o sistema durante las reuniones con el cliente.
2. Requerimientos esperados: Están implícitos en el producto o sistema y quizá sean tan importantes que el cliente no los mencione de manera explícita.
3. Requerimientos emocionantes: Estas características van más allá de las expectativas del cliente

* Escenarios de uso: los desarrolladores y usuarios crean un conjunto de escenarios que identifican la naturaleza de los usos para el sistema que se va a construir. Los escenarios, que a menudo se llaman *casos de uso* proporcionan la descripción de la manera en la que se utilizará el sistema.
* Indagación de los productos de trabajo:

1. Un enunciado de la necesidad y su factibilidad.
2. Un enunciado acotado del alcance del sistema o producto.
3. Una lista de clientes, usuarios y otros participantes que intervienen en la indagación de los requerimientos.
4. Una descripción del ambiente técnico del sistema.
5. Una lista de requerimientos (de preferencia organizados por función) y las restricciones del dominio que se aplican a cada uno.
6. Un conjunto de escenarios de uso que dan perspectiva al uso del sistema o producto en diferentes condiciones de operación.
7. Cualesquiera prototipos desarrollados para definir requerimientos.

Cada uno de estos productos del trabajo es revisado por todas las personas que participan en la indagación de los requerimientos.

**Desarrollo de casos de uso**

Un caso de uso narra una historia estilizada sobre cómo interactúa un usuario final con el sistema en circunstancias específicas, es decir, ilustra el software o sistema desde el punto de vista del usuario final.

El primer paso es definir un conjunto de “actores”, los actores son las distintas personas (o dispositivos) que usan el sistema o producto en el contexto de la función y comportamiento que va a describirse, es cualquier cosa que se comunique con el sistema o producto y que sea externo a éste.

* Los *actores principales* interactúan para lograr la función requerida del sistema y obtienen el beneficio previsto de éste. Trabajan con el software en forma directa y con frecuencia.
* Los *actores secundarios* dan apoyo al sistema, de modo que los primarios puedan hacer su trabajo.

**Elaboración del modelo de requerimientos**

El objetivo del modelo del análisis es describir los dominios de información, función y comportamiento que se requieren para un sistema basado en computadora

* Elementos del modelo de requerimientos:

1. Elementos basados en el escenario: El sistema se describe desde el punto de vista del usuario con el empleo de un enfoque basado en el escenario. Los elementos del modelo de requerimientos basados en el escenario sirven como entrada para la creación de otros elementos de modelado.
2. Elementos basados en clases: Cada escenario de uso implica un conjunto de objetos que se manipulan cuando un actor interactúa con el sistema. Estos objetos se clasifican en clases: conjunto de objetos que tienen atributos similares y comportamientos comunes.
3. Elementos de comportamiento: Tiene un efecto profundo en el diseño que se elija y en el enfoque de implementación que se aplique, debe proveer elementos de modelado que ilustren el comportamiento. El *diagrama de estado* es un método de representación del comportamiento de un sistema que ilustra sus estados y los eventos que ocasionan que el sistema cambie de estado. Un estado es cualquier modo de comportamiento observable desde el exterior. Además, el diagrama de estado indica acciones (como la activación de un proceso, por ejemplo) tomadas como consecuencia de un evento en particular.
4. Elementos orientados al flujo: El sistema acepta entradas en varias formas, aplica funciones para transformarla y produce salidas en distintos modos.

* Patrones de análisis: sugieren soluciones dentro del dominio de la aplicación que pueden volverse a utilizar cuando se modelen muchas aplicaciones. Aceleran el desarrollo de los modelos de análisis abstracto que capturan los principales requerimientos del problema concreto y facilitan la transformación del modelo de análisis en un modelo del diseño.

**Requerimiento de las negociaciones**

Se tiene que entrar en negociaciones con uno o varios participantes. En la mayoría de los casos, se pide a éstos que evalúen la funcionalidad, desempeño y otras características del producto o sistema, en contraste con el costo y el tiempo para entrar al mercado. El objetivo es desarrollar un plan del proyecto que satisfaga las necesidades del participante y que al mismo tiempo refleje las restricciones del mundo real que se hayan establecido al equipo del software.

Conjunto de actividades de negociación al principio de cada iteración del proceso de software. En lugar de una sola actividad de comunicación con el cliente, se definen las actividades siguientes:

1. Identificación de los participantes clave del sistema o subsistema.
2. Determinación de las “condiciones para ganar” de los participantes
3. Negociación de las condiciones para ganar de los participantes a fin de reconciliarlas en un conjunto de condiciones ganar-ganar para todos los que intervienen (incluso el equipo de software).

**Validación de los requerimientos**

A medida que se crea cada elemento del modelo de requerimientos, se estudia para detectar inconsistencias, omisiones y ambigüedades. Los participantes asignan prioridades a los requerimientos representados por el modelo y se agrupan en paquetes de requerimientos que se implementarán como incrementos del software.

Preguntas

1. Su finalidad es la comprensión de cuál será el efecto que tendrá el software en el negocio, qué es lo que quiere el cliente y cómo interactuarán los usuarios finales con el software.
2. **Ingeniería de requerimientos**
3. Indagación de los requerimientos
4. Validación de requerimientos
5. Modelo de requerimientos
6. En esta tarea de la ingeniería de requerimientos se establece el entendimiento básico del problema, las personas que quieren una solución, la naturaleza de la solución que se desea, así como la eficacia de la comunicación y colaboración preliminares entre los otros participantes y el equipo de software
7. Indagación
8. **Concepción**
9. Negociación
10. Elaboración
11. Esta tarea de la ingeniería de requerimientos consiste en preguntar al cliente, a los usuarios y a otras personas cuáles son los objetivos para el sistema o producto, qué es lo que va a lograrse, cómo se ajusta el sistema o producto a las necesidades del negocio y, finalmente, cómo va a usarse el sistema o producto en las operaciones cotidianas.

a) Validación

b) Concepción

c) Negociación

**d) Indagación**

1. Este problema surge cuando los requerimientos están mal definidos o los clientes o usuarios finales especifican detalles técnicos innecesarios
2. **Problemas de alcance**
3. Problemas de entendimiento
4. Problemas de volatilidad
5. En este problema los clientes o usuarios no están completamente seguros de lo que se necesita, no comprenden las limitaciones de su ambiente de computo, tienen problemas para comunicar las necesidades al ingeniero de sistemas.

a) Problemas de alcance

**b) Problemas de entendimiento**

c) Problemas de volatilidad

1. Este problema ocurre porque los requerimientos cambian con el tiempo.
2. Problemas de alcance
3. Problemas de entendimiento
4. **Problemas de volatilidad**
5. Esta tarea de requerimientos busca desarrollar un modelo refinado de los requerimientos que identifique distintos aspectos de la función del software, su comportamiento e información.
6. **Elaboración**
7. Especificación
8. Administración
9. Indagación
10. En esta tarea de requerimientos se pide a clientes, usuarios y otros participantes que ordenen sus requerimientos según su prioridad y que después analicen los conflictos. Un enfoque iterativo que da prioridad a los requerimientos, se evalúa su costo y riesgo, y se enfrentan los conflictos internos.

a) Validación

b) Especificación

**c) Negociación**

d) Indagación

1. Ejemplos de esta tarea de ingeniería de requerimientos pueden ser un documento escrito, un conjunto de modelos gráficos, un modelo matemático formal, un conjunto de escenarios de uso, un prototipo o cualquier combinación de éstos

a) Elaboración

**b) Especificación**

c) Negociación

d) Concepción

1. En esta tarea de la ingeniería de requerimientos la calidad de los productos del trabajo de los requerimientos se evalúa y se analiza la especificación a fin de garantizar que todos ellos han sido enunciados sin ambigüedades; participan ingenieros de software, clientes, usuarios y otros participantes.
2. Administración
3. **Validación**
4. Concepción
5. Indagación
6. Esta tarea de la ingeniería de requerimientos consiste en un conjunto de actividades que ayudan al equipo del proyecto a identificar, controlar y dar seguimiento a los requerimientos y a sus cambios en cualquier momento del desarrollo del proyecto.

a) Elaboración

b) Especificación

c) Negociación

**d) Administración**

1. Cualquier persona que se beneficie en forma directa o indirecta del sistema en desarrollo
2. **Participante**
3. Cliente
4. Desarrollador
5. Inversor
6. Objetivos y metas que se establecen para un producto o sistema durante las reuniones con el cliente.
7. Requerimientos emocionantes
8. Requerimientos esperados
9. **Requerimientos normales**
10. Están implícitos en el producto o sistema y quizá sean tan importantes que el cliente no los mencione de manera explícita.

a) Requerimientos emocionantes

**b) Requerimientos esperados**

c) Requerimientos normales

1. Estas características van más allá de las expectativas del cliente.

**a) Requerimientos emocionantes**

b) Requerimientos esperados

c) Requerimientos normales

1. Narra una historia estilizada sobre cómo interactúa un usuario final con el sistema en circunstancias específicas, es decir, ilustra el software o sistema desde el punto de vista del usuario final.
2. **Caso de uso**
3. Lista de requerimientos
4. Prototipos desarrollados
5. Alcance del sistema
6. Estos actores interactúan para lograr la función requerida del sistema y obtienen el beneficio previsto de éste. Trabajan con el software en forma directa y con frecuencia.
7. **Actores principales**
8. Actores secundarios
9. Cada escenario de uso implica un conjunto de objetos que se manipulan cuando un actor interactúa con el sistema. El conjunto de objetos que tienen atributos similares y comportamientos comunes se les llama:
10. **Clases**
11. Escenarios
12. Requerimientos
13. Método de representación del comportamiento de un sistema que ilustra sus estados y los eventos que ocasionan que el sistema cambie de estado, también indica acciones (como la activación de un proceso, por ejemplo) tomadas como consecuencia de un evento en particular.
14. Diagrama de flujo
15. Diagrama de estado
16. Diagrama de clases