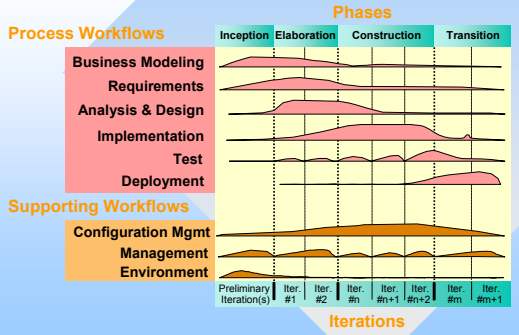
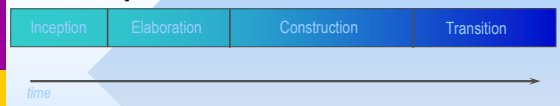
Unified Process structure

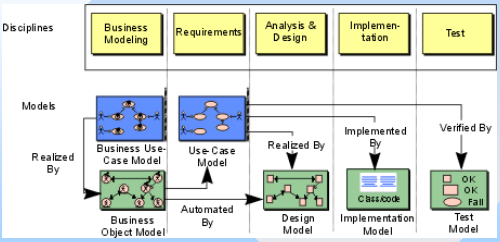


Lifecycle Phases

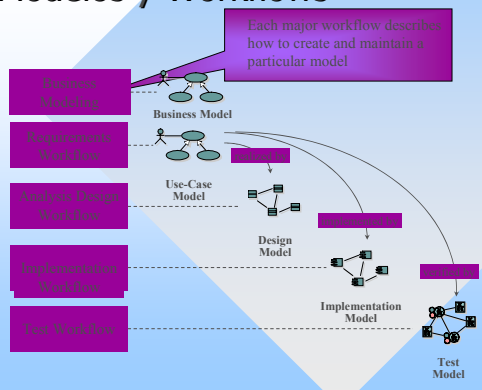


- Inception — Define the scope of the project and develop business case
- Elaboration — Plan project, specify features, and baseline the architecture
- Construction — Build the product
- Transition — Transition the product to its users

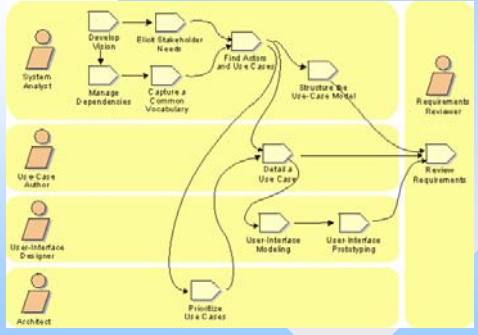
Cada Disiplina Contribuye a un conjunto de Modelos



Modelos y Workflows



Ejemplo de un Workflow



De la Visión a los Requerimientos

- Capturar los requerimientos es el proceso de encontrar que se debe construir.
- Aproximación tradicional → Lista de Requerimientos → cientos (miles) de paginas consistentes.
- Aún el usuario no entenderá lo que el sistema debería hacer hasta que esté casi terminado.
- Más allá de la interacción → el valor agregado al negocio.
- Propósito de los Requerimientos es construir el sistema correcto.
  - Descripción de los Requerimientos (condiciones y capacidades)
  - Alcances del sistema.
  - El usuario deben entender el resultado de la captura de los requerimientos -> lenguaje del cliente.

De la Visión a los Requerimientos

- Cada Proyecto es único (clases de sistemas, cliente, organización, tecnología, ...)
- Actividades, posiblemente paralelas:
  - Listar los requerimientos candidatos
  - Entender el contexto del sistema
  - Capturar los requerimientos funcionales
  - Capturar los requerimientos no funcionales
- Listar los requerimientos candidatos
  - Aspectos que contienen ideas generales.
  - Sirven para planificar el trabajo.
  - Contienen un nombre y una breve explicación.
    - Estado (propuesto, aceptado, incorporado, validado).
    - Costo estimado en desarrollo (recursos, horas-hombre)
    - Prioridad (crítica, importante, auxiliar)
    - Nivel de Riesgo (crítico, significante, común).

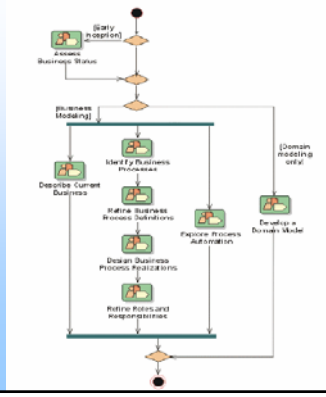
Comprender el Contexto del Sistema

- Modelo del Dominio
  - Describe los conceptos importantes dentro del contexto del dominio.
  - Ayuda a desarrollar un glosario de términos (están también las clases candidatas que son estarán en el modelo). Permiten establecer un vocabulario común.
  - Surgen de la especificación de requerimientos, de entrevistas con los expertos del dominio, workshops con los expertos del dominio.
  - Debe ayudar a entender el problema (no deben representarse detalles, sin visualizar la forma interna de solucionar el problema).
- Modelo del Negocio
  - Describe los procesos (existentes o no) de la organización.
  - Ingeniería de procesos busca mejorar los procesos.
  - Especifica los procesos que serán soportados por el sistema.
  - Establece las necesidades requeridas en cada proceso (trabajadores, responsabilidades y operaciones a ser ejecutadas).

Modelo del Dominio vs Negocio

- El modelado del dominio es una variante simplificada del modelado del negocio, donde solo se mira "las cosas".
- Clases del Dominio: son extraidas del experto, de otras clases, de las especificaciones de requerimiento, ...
- Entidades del Negocio: se comienza con los clientes, identifica los CU del negocio, y luego las entidades.
- Clases del Dominio tienen atributos, y ninguna o muy pocas operaciones. No así las entidades del negocio. Identifica los trabajadores que usarán las entidades a través de operac.
- Cada CU puede ser identificado en componentes que implementan el sistema.
- Cuando solamente un modelo del dominio es usado, no hay una manera obvia de relacionar el modelo del dominio con los CU del sistema.

Framework



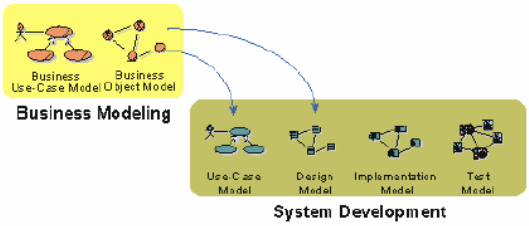
Captura de Requerimiento Funcionales y No Funcionales

- Requerimientos Funcionales -> Casos de Uso.
  - Cada usuarios quiere que el sistema realice algo para él.
  - Para el usuario un CU es una forma de utilizar el sistema.
  - Se requiere conocimiento de las necesidades de los usuarios y clientes -> entrevistas, discusión de propuestas,...
- Un documento "requerimientos suplementarios" contiene los requerimientos no funcionales, en un formato tradicional.
  - Interfaz: especifica la interfaz con un item externo con el cual sistema debe interactuar.
  - Físico: especifica los requerimientos de HW.
  - Diseño: restricciones de diseño (extensible y de mantenimiento).
  - Implementación: restricciones de codificación y construcción.
  - Otros: seguridad, disponibilidad, facil de aprender, etc.

De la Visión a los Requerimientos

Trabajo a Realizar	Resultado
Listar los Requerimientos Candidatos	Lista de Características
Entender el Contexto del Sistema	Modelo del Dominio o Negocio
Capturar los Requerimientos Funcionales	Modelo de Caso de Uso
Capturar los Requerimientos No Funcionales	Requerimientos Suplementarios o Casos de Uso

Relación entre el Negocio y el Sistema



Ciclo de Vida - Requerimientos

- Fase de inicio: identificar la mayoría de los casos de usos, el alcance del proyecto, y detallar los más críticos.
- Fase de Elaboración: capturar la mayoría de los requerimientos → medir el tamaño del esfuerzo de desarrollo. La meta es capturar el 80% de los requerimientos y describir la mayoría de los casos de usos.
- Fase de Construcción: Requerimientos restantes son capturados e implementados.
- Fase de Transición: No hay requerimientos, a menos que cambien.

Captura de Requerimientos como Casos de Uso

- Los requerimientos funcionales son estructurados como CU.
- Los CU especifican una secuencia de acciones, incluyendo variantes, que el sistema debe ejecutar produciendo un resultado de valor observable a un actor particular.
- Los CU permiten llegar a un acuerdo entre los desarrolladores y los clientes respecto de los requerimientos (condiciones y capacidades).
- Actores corresponden a trabajadores o actores del negocio en el marco de un proceso de negocio. Rol.
- Cada manera en que el actor usa al sistema es representado como un Caso de Uso.
- Se utilizan otros diagramas para dar mayor precisión (Diagrama de Estados, Actividades, Colaboración, Secuencia).
- Una instancia de un caso de uso es una ejecución.
- Un CU es atómico.

Análisis

- Alcanzar un entendimiento más preciso de los requerimientos.
- Es probable todavía permanezcan problemas no resueltos con respecto a los requerimientos del sistema.
- Se analizan los requerimientos en mayor profundidad.
  - Los CU deben ser independientes uno de otro, tanto como sea posible. Análisis: se pueden reflejar las interacciones internas, incluido los recursos internos compartidos.
  - CU deben ser descriptos utilizando el lenguaje del cliente. Análisis: se usa el lenguaje del desarrollador.
  - Cada CU debe ser estructurado para formar una especificación intuitiva y completa de la funcionalidad. Análisis: estructurar los requerimientos para facilitar reusarlos y mantenerlos.

Comparación

Modelo de CU	Modelo de Análisis
Usa el lenguaje del cliente	Usa el lenguaje del Desarrollador
Vista externa del sistema.	Vista Interna del Sistema
Estructurado por CU.	Estructurado por clases estereotipadas y paquetes.
Contrato entre el cliente y los desarrolladores.	Usado por los desarrolladores para entender como el sistema sería amoldado.
Puede contener redundancia e inconsistencia entre requerimientos.	No debería tener redundancia ni inconsistencia entre requerimientos.
Captura la funcionalidad del Sistema	Bosqueja cómo realizar la funcionalidad dentro del sistema.
Define CU que son analizados extensamente en el modelo del Análisis.	Define realizaciones de CU, cada uno representando el análisis de un CU del modelo de CU.

Análisis

- Modelo del Diseño: amoldar el sistema y contrar la forma, incluyendo la arquitectura.
  - Una forma con componentes de código que son compiladas e integradas en el release ejecutable.
  - Una forma que puede ser mantenible a lo largo del tiempo.
  - Decisiones de requerimientos de performance y distribución (requerimientos no funcionales).
- Análisis puede ser visto como un primer corte al diseño.
  - Puede ser usado para planificar el diseño e implementación (distribución del trabajo de desarrollo).
  - Provee una vista global del sistema. Muy valioso para nuevos desarrolladores o quienes deben mantenerlo.
  - Es un modelo compartido para diferentes diseños o implementaciones (distintos lenguajes o plataformas).
  - Reingeniería en términos del modelo del análisis, para evitar entrar en detalles de diseño e implementación.

El rol del Análisis en el Ciclo de Vida

- El proyecto usa el modelo del análisis para describir:
  - el resultado del análisis, y mantener la consistencia del modelo.
  - el resultado del análisis pero visualiza el modelo como una herramienta transitoria e intermedia. En la fase de elaboración, el modelo ya no se mantiene más. Los problemas de análisis que aparezcan son resueltos e integrados como parte del trabajo del diseño.
  - No lo usa para describir los resultados. El proyecto analiza los requerimientos como parte integrada del:
    - Modelo de CU. Requiere más formalismo en el modelo de CU. Justificable si el cliente los puede entender.
    - Modelo del diseño. Complica más el trabajo del diseño. Justificable si los requerimientos son simples o bien conocidos.
- Los dos primeros tienen un costo de mantenimiento del modelo del análisis a través de varias iteraciones. La tercer variante solo usable en sistemas muy simples.

Clases del Análisis

- Una clase del análisis representa una abstracción de una o varias clases y/o subsistemas. Las clases del análisis:
  - Se focalizan sobre la manipulación de los requerimientos funcionales, y pospone los no funcionales.
  - Rara vez proveen alguna interfaz en términos de operaciones y signaturas. Su comportamiento es definido por responsabilidades a alto nivel de abstracción (descripción textual de un subcjto cohesivo del comportamiento).
  - Definen atributos también a alto nivel. Los tipos son del dominio del problema, mientras que en el diseño e implementación son tipos del lenguaje de programación. Generalmente, atributos del análisis se convierten en clases en el diseño e implementación.
  - Están involucradas en relaciones conceptuales.
  - Pueden ser de tres estereotipos básicos: límite, de control, y entidad.

Estereotipos

- Clase límite: modela la interacción entre el sistema y el actor
  - Recepción y presentación de información.
  - Separan la interfaz del usuario o comunicación con el usuario.
  - Representan abstracciones de ventanas, forms, paneles, sensores, API (sistemas externos).
  - No describe cómo la interacción es realizada físicamente.
  - Está relacionada con al menos un actor, y un actor está relacionado con al menos una clase límite.
- Clase entidad: modela información persistente.
  - En la mayoría de los casos son derivadas directamente de las clases entidad del negocio o dominio. Estos pueden capturar información que no es manipulada dentro del sistema.
  - Muestran una estructura de datos lógica y contribuyen a entender que información manipular.

Estereotipos (II)

- Clase control: representa coordinación, secuencia, transacción y control de otros objetos.
  - Son usadas para encapsular el control relacionado a un CU.
  - Usadas para representar derivaciones y cálculos complejos, tal como lógica del negocio, que no puede ser representada por ninguna clase entidad específica.
  - La dinámica del sistema son modeladas por las clases control, dado que manejan y coordinan los flujos de control y acciones principales y delegan trabajo a otros objetos (clases entidad y límite).
- Un objeto límite aparece en una ventana y puede participar en dos o más CU. Es creado y terminado dentro de una realización de CU simple.
- Un objeto entidad no es específico de una realización de CU.
- Un objeto control encapsula el control relacionado a un CU específico. Hay excepciones donde participa en más de una realización de CU, o una realización no tiene obj. control.

Diseño

- Propósito:
  - Entender en profundidad requerimientos no funcionales y restricciones relacionadas a los leng. de prog. reuso de componentes.
  - Descomponer el trabajo de implementación en partes manejables. (posiblemente por distintos grupos de desarrollo concurrentemente)
  - Capturar las grandes interfaces entre subsistemas.

Modelo de Análisis	Modelo del Diseño
Modelo Conceptual. Evita ideas de implementación.	Modelo Físico.
Aplicable a distintos diseños	Específico a una implementación
Tres estereotipos de Clases	Estereot. depend. del Leng Prog
- Formal; + Barato; - Capas	+ Formal; + Caro; + Capas.
Dinámico sin focalizar secuencia	Dinámico indicando secuencia.
Puede no ser mantenido durante el ciclo de vida completo.	Debe ser mantenido.