

Magíster en Ingeniería de Software

Tesis

Ingeniería de Requerimientos

Proceso LEC

“Un Proceso definido para aplicar LEL, Escenarios y CRC”

Alumno: Lic. Javier A. Zuñiga

Director: Lic. Alejandro Oliveros

UNLP
Facultad de Informática

Abril de 2004

Abstract

En esta tesis se presenta un framework para un proceso de ingeniería de requerimientos basado en lenguaje natural. Utilizando Léxico Extendido del Lenguaje (LEL), Escenario y CRC s.

En el capítulo 1 se hace una introducción a la ingeniería de requerimientos, con el fin de comprender la importancia que tiene esta etapa en el proceso de ingeniería de software.

El segundo, tercero y cuarto capítulo, describen las técnicas de LEL, Escenarios y CRC respectivamente, según lo que proponen los autores.

En el quinto capítulo se describe como se define un proceso de ingeniería de software y la forma en que se pueden efectuar evaluaciones para poder realizar mejoras. En este capítulo también se describe el framework que se propone para cumplir con los objetivos fijados para el presente trabajo.

El capítulo sexto describe el proceso utilizando el framework propuesto en el capítulo anterior y las técnicas descritas en los capítulos dos, tres y cuatro.

El proceso propuesto guía la implementación de las técnicas y herramientas en un proyecto de desarrollo de software, como así también la capacitación de quienes no la conocen.

Para verificar el proceso y los beneficios que proporciona el trabajar con herramientas basadas en lenguaje natural, se aplico a un caso real.

El caso de estudio, es un sistema que permite a una universidad registrar y hacer un seguimiento de todas las consultas realizadas por los alumnos a la secretaría de Asuntos Estudiantiles que cotidianamente se presentan diferentes casos problemáticos de distinta índole algunos son aislados y otros comunes a una franja determinada de alumnos. Aquí se plantean problemas y sugerencias, esto exige comunicación directa con los responsables de cada sector para la resolución temprana de problemas detectados, además colabora con la imagen institucional mediante la explicación particular y personal en cada caso. Desde ella se facilita el monitoreo de la percepción de las actividades por parte de la comunidad estudiantil.

Como resultado final se obtuvo un producto de software que permite hacer el seguimiento de las consultas realizadas por los alumnos.

El realizar este caso permitió probar el proceso propuesto, la efectividad de comunicación de las herramientas basadas en lenguaje natural con el usuario final y las diferentes etapas de desarrollo de software.

Índice

<u>ABSTRACT</u>	I
<u>INGENIERÍA DE REQUERIMIENTOS</u>	1
1.1- OBJETIVOS Y ALCANCES DEL TRABAJO.....	2
1.2- INGENIERÍA DE REQUERIMIENTOS.....	3
1.3- IMPORTANCIA Y ROL DE LOS REQUERIMIENTOS.....	3
1.4- PROCESO DE LA INGENIERÍA DE REQUERIMIENTOS.....	5
1.4.1-PROCESO DETALLADO.....	5
1.4.2-TÉCNICAS DE ELICITACIÓN.....	8
<u>LÉXICO EXTENDIDO DEL LENGUAJE - LEL</u>	12
2.1- INTRODUCCIÓN.....	13
2.2- OBJETIVOS DEL LEL.....	13
2.3 PROCESO DE CONSTRUCCIÓN.....	15
<u>ESCENARIOS</u>	18
3.1- INTRODUCCIÓN.....	19
3.2-OBJETIVOS DE LOS ESCENARIOS.....	19
3.3-COMPONENTES DE LOS ESCENARIOS.....	20
3.4- TÉCNICA DE CONSTRUCCIÓN.....	22
<u>CLASE RESPONSABILIDADES COLABORACIONES - CRC</u>	35
4.1- CONCEPTO DE CRC.....	36
4.2- HEURÍSTICAS DE DERIVACIÓN DE CRC.....	36
4.2.1 HEURÍSTICAS DE IDENTIFICACIÓN DE CLASES PRIMARIAS Y SUS RESPONSABILIDADES.....	38
4.2.2 HEURÍSTICAS DE IDENTIFICACIÓN DE CLASES SECUNDARIAS Y SUS RESPONSABILIDADES.....	39
4.2.3. HEURÍSTICAS GENERALES PARA LA DEFINICIÓN DE LAS CLASES.....	42
4.2.5 DOCUMENTACIÓN DE LAS CLASES.....	46
4.2.6 EVALUACIÓN DE LAS CRCs.....	46
<u>FRAMEWORK DEL PROCESO DE CONSTRUCCIÓN</u>	48
5.1- INTRODUCCIÓN A PROCESOS DE SOFTWARE.....	49
5.1.1 DEFINICIÓN Y CONTENIDO DE UN PROCESO.....	49
5.1.2 CLASIFICACIÓN DE PROCESOS.....	49
5.1.3- FRAMEWORKS & TEMPLATES.....	50

5.2-DEFINICIÓN DE FRAMEWORK Y TEMPLATE PARA EL PROCESO LEC (LEL, ESCENARIOS Y CRC).	54
5.3- TEMPLATE DEL PROCESO	59
<u>DEFINICIÓN DEL PROCESO LEC</u>	<u>60</u>
6.1- INTRODUCCIÓN.	61
6.2- PROCESO LEC	61
6.3-DETERMINAR FUENTES DE INFORMACIÓN.	63
6.4- CONSTRUCCIÓN DEL LEL	67
6.5- CONSTRUCCIÓN DE ESCENARIOS	88
6.6- CONSTRUCCIÓN DE CRC S	108
<u>CASO DE ESTUDIO</u>	<u>122</u>
7.1- INTRODUCCIÓN	123
7.2- SÍNTESIS DEL CASO DE ESTUDIO	123
7.2.1-DETERMINACIÓN DE FUENTES DE INFORMACIÓN	125
7.2.2-CONSTRUCCIÓN DEL LEL	128
7.3.3-CONSTRUCCIÓN DE ESCENARIOS	135
7.3.4- CONSTRUCCIÓN DE CRC S	140
7.4 CONCLUSIÓN DEL CASO	142
<u>CONCLUSIONES Y TRABAJOS FUTUROS</u>	<u>143</u>
8.1- CONCLUSIONES.	144
8.2- APORTES REALIZADOS POR ESTE TRABAJO	145
8.3-TRABAJOS FUTUROS.	147
<u>DOCUMENTOS DEL CASO DE ESTUDIO</u>	<u>148</u>
WORK PRODUCTS	149
LISTADO DE STAKEHOLDERS	149
LISTADO DE FUENTES DE INFORMACIÓN ADICIONAL	149
LISTA DE SÍMBOLOS DEL LEL	149
LISTA DE SÍMBOLOS CLASIFICADOS	150
LISTA SÍMBOLOS DESCRIPTOS	151
ESCENARIOS.....	157
ESCENARIO CANDIDATOS	157
ESCENARIO INTEGRADOR.....	157
CRC S DERIVADAS	160
ESCENARIOS DE LA APLICACIÓN	163
DIAGRAMA DE CLASES DE LA APLICACIÓN	167
INTERFACES DEL SISTEMA	167
<u>REFERENCIAS</u>	<u>175</u>

Ingeniería de Requerimientos

Objetivos del Capítulo

El objetivo de este capítulo es fijar los objetivos del presente trabajo y hacer una introducción a la ingeniería de requerimientos.

Contenido

- 1.1-Objetivos del trabajo.
- 1.2-Ingeniería de requerimientos.
- 1.3-Importancia y rol de los requerimientos.
- 1.4-Proceso de generación de requerimientos.

1.1- Objetivos y alcances del trabajo.

Objetivos

El objetivo del presente trabajo es establecer un framework para un proceso de ingeniería de requerimientos basado en lenguaje natural, con LEL & Escenarios & CRC.

La utilidad que se pretende alcanzar es:

- Utilizar el framework como guía paso a paso para la implementación de un proyecto de ingeniería de software.
- Utilizar el framework como instructivo para quienes no conocen la técnica y deban capacitarse.
- Demostrar que partiendo de la aplicación de las herramientas basadas en lenguaje natural y siguiendo un proceso de desarrollo de software se llega a un producto que funciona.

Alcances

En el presente capítulo se hace una introducción a la ingeniería de requerimientos, con el fin de comprender la importancia que tiene esta etapa en el proceso de ingeniería de software.

El segundo, tercer y cuarto capítulo, describen las técnicas de LEL, Escenarios y CRC respectivamente, según la definición de sus autores.

En el quinto capítulo se describe como se define un proceso de ingeniería de software. También describe el framework propuesto en cumplimiento de los objetivos del presente trabajo.

El capítulo sexto describe el proceso utilizando el framework propuesto en el capítulo anterior y las técnicas descritas en los capítulos dos, tres y cuatro.

No está dentro de los alcances proponer modificaciones a las técnicas, sino darles un marco para facilitar su implementación, teniendo en cuenta cuáles son los aspectos fundamentales en los que se debe enfocar cada etapa, para poder realizar una implementación exitosa.

Aunque se construya un producto final para demostrar la aplicación del framework, no están dentro del alcance de este trabajo, las etapas del proceso de desarrollo de software, posteriores a la etapa de requerimientos.

Para probar el framework y su eficiencia en la transmisión de las técnicas, el mismo se utilizó para desarrollar 32 casos de estudio en trabajos prácticos de alumnos de tercer año, de la carrera de Ingeniería en informática.

Por último se aplicó en un caso real, que se describe en el capítulo siete.

En el capítulo 8 se presentan las conclusiones obtenidas.

1.2- Ingeniería de requerimientos.

Requerimientos Definición - IEEE-STD 610 (1990)

1. Condición o capacidad que necesita el usuario para resolver un problema o alcanzar un objetivo.
2. Condición o capacidad que debe satisfacer o poseer un sistema o una componente de un sistema para satisfacer un contrato, un estándar, una especificación u otro documento formalmente impuesto.
3. Representación documentada de una condición o capacidad como en 1 o 2.

Los requerimientos pueden clasificarse en:

Requerimientos funcionales: descripción de lo que el sistema debe hacer, qué información necesita ser mantenida y qué necesita ser procesado.

Requerimientos no funcionales: restricciones globales sobre cómo debe construirse y funcionar el sistema.

Requerimientos de empresa: razones de la creación del sistema, restricciones del ambiente en el que el sistema funciona, significado de los requerimientos del sistema.

Los requerimientos son sensibles a los tipos de sistemas, según estos sean: orientados al mercado, o específicos para un cliente.

Teniendo en cuenta los conceptos anteriores podemos definir ingeniería de requerimientos como:

“Es el proceso **sistemático** de desarrollar requerimientos a través de un proceso **cooperativo e iterativo** de **analizar** el problema, **documentar** las **observaciones** resultantes en una **variedad de formatos** de representación y **verificar** la precisión de la comprensión obtenida” [Loucopoulos 95]

1.3- Importancia y rol de los requerimientos.

En 1994 el Standish Group hizo un estudio sobre 350 compañías y cerca de 8000 proyectos de software para averiguar cómo se estaban desarrollando los procesos de software. Los resultados fueron los siguientes. El 31% de los proyectos de software fueron cancelados antes de completarse. En las grandes empresas, solo el 9% de los proyectos fue entregado dentro del costo y tiempo que se habían presupuestado; el 16% alcanzó estos criterios en las compañías pequeñas [Standish 94].

Para comprender por qué, [Standish 95] se pidió a los participantes en el estudio que explicaran las causas de los proyectos fallidos. Los principales factores informados fueron:

- requerimientos incompletos (13,1%)
- falta de compromiso del usuario (12,4%)
- falta de recursos (10,6%)
- expectativas no realistas (9,9%)
- falta de soporte ejecutivo (9,3%)

- requerimientos y especificaciones cambiantes (8,7%)
- falta de planeamiento (8,1%)
- fin de la necesidad del sistema (7,5%)

Notemos que cierta parte de las etapas de extracción, la definición y la gestión del proceso de los requerimientos puede llevar a una gran cantidad de problemas [Pfleeger 02]: construir un sistema que resuelve el problema equivocado, que no funciona como se espera o que presenta dificultades para que los usuarios puedan comprenderlo y utilizarlo. Es más, un proceso de requerimientos mediocre puede en realidad resultar muy caro. El trabajo de Bohem y Papaccio (1998) se consigna que si cuesta \$1 localizar y subsanar, durante el proceso de definición, un problema debido a requerimientos, puede costar \$5 repararlo durante el diseño, \$10 durante la codificación, \$20 durante la prueba unitaria y hasta \$200 después de la entrega del sistema. Así vemos que es rentable tomar el tiempo que sea necesario para comprender el problema y su contexto y obtener los requerimientos correctos desde el primer momento.

Impacto de los errores en la etapa de requerimientos

Los impactos de errores en esta etapa [Davis 93] pueden producir:

- El software resultante puede no satisfacer a los usuarios
- Las interpretaciones múltiples de los requerimientos pueden causar desacuerdos entre clientes y desarrolladores
- Puede gastarse tiempo y dinero construyendo el sistema erróneo.

Catarata de errores de Mizuno

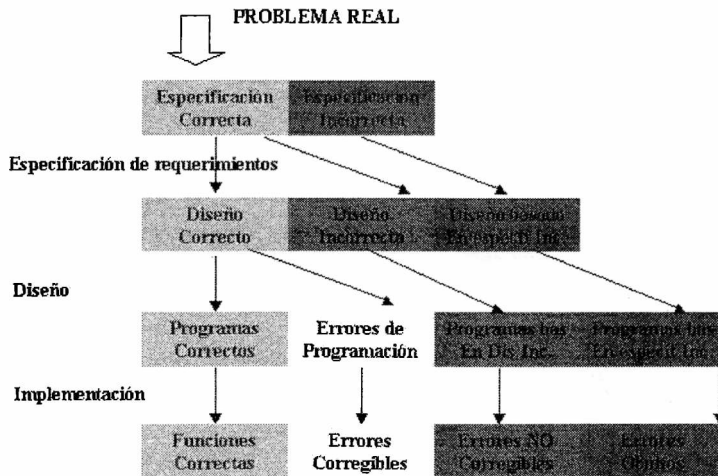


Figura 1.1

Este gráfico, de la Figura 1.1, transmite la importancia que tiene el partir de una especificación correcta de las necesidades del usuario y las consecuencias que puede tener durante el desarrollo del software, el no haberlo conseguido.

Rol de los requerimientos

El rol que tienen los requerimientos son los siguientes:

- Acuerdo desarrolladores-clientes-usuarios finales
- Base para el diseño del software
- Soporte para verificación y validación del software
- Soporte a la evolución del sistema

1.4- Proceso de la ingeniería de requerimientos.

Aspectos principales de la ingeniería de requerimientos [Loucopoulos 95]

Los aspectos fundamentales de la Ingeniería de Requerimientos son tres:

- comprender el problema
- describir el problema
- acordar sobre la naturaleza del problema

Cada uno de estos aspectos se corresponden con la elicitación, especificación y validación respectivamente

1.4.1-Proceso detallado

Esquema del proceso de Ingeniería de Requerimientos puede verse en la Figura 1.2

Cada proceso se describe en función de:

- propósito
- el input y sus orígenes
- las actividades que tienen lugar durante la ejecución del proceso y las técnicas usadas
- los entregables final e intermedios
- la interacción con otros procesos de la Ingeniería de Requerimientos

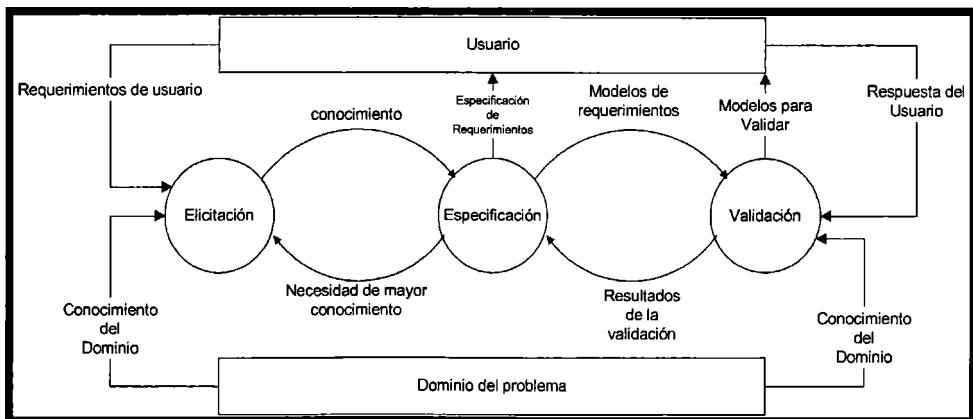


Figura 1.2
Ingeniería de Requerimientos – Framework
Loucopoulos 95

Elicitación :

Cuando se trata de resolver un problema la primero es descubrir más sobre él.

Propósito:

Obtener conocimiento relevante del problema, para producir una especificación rigurosa del software necesario para resolver el problema.

Input :

Fuentes del conocimiento del dominio:

- Expertos del dominio
- Literatura sobre el dominio
- Software existente en el dominio
- Software similar en otros dominios
- Estándares nacionales e internacionales
- Otros stakeholders

Actividades :

- Identificar fuentes de conocimiento
- Adquirir el conocimiento
- Decidir sobre la relevancia de un conocimiento
- Comprender la significación del conocimiento y su impacto

Técnicas más usadas:

- Entrevistas
- Prototipos
- Reuso de conocimiento

Productos:

Es un proceso de creación de modelos

- desde los modelos conceptuales hasta el modelo de la especificación de requerimientos.
- el analista comienza con modelos mentales con conocimiento dependiente del dominio
- al avanzar, los modelos son más orientados al software
- no se produce ningún modelo formal
- sucesión de modelos mentales del dominio del problema

Especificación:

Una especificación puede ser vista como un contrato entre usuarios y desarrolladores de software, que define el comportamiento funcional deseado del artefacto de software (y otras propiedades de este, tales como performance, confiabilidad, etc.) sin mostrar como será alcanzada tal funcionalidad

Propósito

- Acuerdo usuario-desarrolladores sobre el problema a resolver
- Pauta para el desarrollo de un sistema de software

Input

- Conocimiento sobre el dominio del problema
- Lo provee el proceso de elicitación

Actividades

- Análisis y asimilación del conocimiento de los requerimientos
- Síntesis y organización del conocimiento en un modelo de requerimientos coherente y lógico

Productos:

Se producen varios modelos:

- modelos orientados al usuario, que especifican comportamiento, características no funcionales, etc.
- modelos orientados al desarrollador, que especifican propiedades funcionales y no funcionales del software y restricciones

Relación con otros procesos:

Es el proceso principal de la Ingeniería de Requerimientos

- Controla los otros procesos
- Elicitación-Especificación:
 - puede requerir más información de la elicitación
 - cambios en el conocimiento del dominio del problema generan cambios en la especificación
- Especificación-validación: Cada parte de la especificación dispara necesidades de validación

Validación

Proceso que certifica que se ataca el problema correcto.

Propósito

- Certifica la consistencia del modelo de requerimientos con las intenciones de clientes y usuarios
- La necesidad aparece cuando se modifica el modelo actual
- Se aplica también a los modelos intermedios

Inputs

- Todo modelo está sujeto a validación, por lo tanto cada modelo es input
- El conocimiento sobre el dominio del problema debe ser validado

Actividades

- Interacción entre analistas, clientes del sistema y usuarios en el dominio del problema.
- Caracterizada por:
 - preparación del experimento
 - ejecución del experimento y análisis de los resultados.

Productos

- Modelo de requerimientos en línea con las expectativas de los usuarios
- No significa que el modelo sea correcto
- Compromiso entre lo deseado y lo posible y factible

Relación con otros procesos

La validación está en todos los procesos de la Ingeniería de Requerimientos, la dispara:

- un nuevo conocimiento sobre el dominio del problema (elicitación)
- la formulación de un modelo de requerimientos (especificación)

La validación se requiere en las etapas de análisis y síntesis dado que debe comprobarse la exactitud de la información

1.4.2-Técnicas de elicitación

Las técnicas de elicitación más utilizadas [Antonelli 02] son:

Tradicionales: entrevistas, cuestionarios, encuestas y análisis de formularios.

Grupales: brainstorming y focus group

Prototipación

Model-Driven: objetivos y escenarios

Cognitivas: análisis de protocolo, laddering, card storing, repertory grid

Contextuales: Observación de participantes, etnometodología, análisis de conversación

En este trabajo se hace una descripción breve de la técnica de entrevistas por ser la que más se utiliza para la implementación del proceso propuesto.

Entrevistas

Una entrevista para recolección de información, es una conversación dirigida con un propósito específico que usa un formato de preguntas y respuestas.

En la entrevista se quiere obtener la opinión del entrevistado y sus sentimientos acerca del estado actual del sistema, los objetivos de la organización, los personales y los procedimientos informales.

Planificación de la entrevista

Los cinco pasos principales para la entrevista [Kendall 97] son:

1. Leer material a fondo.
Leer y comprender información sobre el entrevistado y su organización tanto como sea posible.
Poner particular atención con el lenguaje que usan los miembros de la organización para describirse a sí mismos y a su organización.
2. Establecer objetivos de las entrevistas.
Usar la información que se recopiló, así como su propia experiencia, para establecer los objetivos de la entrevista.
Áreas que nos interesa obtener información: fuentes de información, formatos de la información, frecuencia de la toma de decisiones.
3. Decidir a quien entrevistar.
Cuando se decida a quien entrevistar elija personas clave del sistema que se debe modelar.
4. Preparar al entrevistado.
Avisar con anticipación a quien se va a entrevistar para permitirle pensar acerca de la entrevista.
5. Decidir sobre los tipos y estructura de preguntas.
Preparar las entrevistas pensando a quien va dirigida y su puesto jerárquico.
Los tipos básicos de preguntas son abiertas y cerradas. Cada una tiene beneficios y desventajas.

Preguntas abiertas.

Las preguntas abiertas incluyen aquellas tales como “¿Cómo ve los objetivos de este departamento?” “¿Cuál es su opinión del sistema de computadora actual?”
Las ventajas y desventajas se pueden ver en la Figura 1.1

Ventajas	Desventajas
Da más libertad al entrevistado	Preguntas que puedan dar como resultado muchos detalles relevantes
Permite que el entrevistador recoja el vocabulario del entrevistado.	Se puede perder el control de la entrevista
Se pueden recopilar muchos detalles	Las respuestas pueden requerir mucho tiempo y la utilidad que se le puede dar no lo justifica
Da información de guía para preguntas que fueron omitidas	Pueden mostrar potencialmente que el entrevistador no está preparado.
Hace que sea más interesante para el entrevistado	Puede dar la impresión de que el entrevistador está en una “expedición de pesca” sin un objetivo real para la entrevista
Permite más espontaneidad	
Hace que la construcción de frases sea más fácil para el entrevistador	

Figura 1.3

Preguntas cerradas.

Las preguntas cerradas, que tienen la forma básica “¿Qué tantos subordinados tiene?”. Las respuestas posibles están cerradas al entrevistado, debido a que solamente puede responder con un número finito, tal como “ninguno”, “uno”, o “quince”.

Las ventajas y desventajas de este tipo de preguntas se pueden ver en la Figura 1.4

Ventajas	Desventajas
Se ahorra tiempo.	Ser aburridas para el entrevistado.
Facilita la comparación de los resultados de las entrevistas.	No proporcionan grandes detalles.
Se llega al punto que se quiere tratar.	Se pierden ideas principales por la razón anterior.
Se mantiene control sobre la entrevista.	No se llega a establecer una relación armoniosa entre el entrevistado y el entrevistador.
Se obtienen datos relevantes.	

Figura 1.4

Entrevista de comienzo y final abierto:

Forma más simple de interacción analista-usuario

- el analista deja que el usuario hable de su tarea
- ambiente informal
- útiles para obtener visiones generales
- no son útiles para obtener información detallada
- preguntas cerradas, abiertas, de sondeo y de guía

Entrevistas estructuradas o cerradas:

Dirige al usuario hacia aspectos específicos de requerimientos a elicitar

- son útiles para información detallada
- preguntas cerradas, abiertas, de sondeo y de guía
- información para gaps, obstáculos y soporte

Capítulo 2

Léxico Extendido del Lenguaje - LEL

Objetivos del Capítulo

El objetivo de este capítulo es plantear los objetivos, componentes y método de construcción del LEL.

Contenido

- 2.1- Introducción
- 2.2-Objetivos del LEL
- 2.3-Proceso de construcción del LEL

2.1- Introducción

El Léxico Extendido del Lenguaje es una representación de los símbolos del lenguaje del dominio de la aplicación, que intenta capturar el vocabulario de una aplicación. Su objetivo principal es que el ingeniero de software entienda el lenguaje que habla el usuario, entendiendo los términos que utiliza, sin preocuparse por entender el problema [Leite93] [Leite97b].

El LEL involucra la denotación y la connotación de cada símbolo descubierto como una palabra o frase relevante del dominio de la aplicación.

La información recopilada en este capítulo será utilizada en el capítulo 6 para describir la actividad Construcción del LEL

2.2- Objetivos del LEL

El objetivo que se propone en la construcción del LEL no es solamente mejorar la comunicación entre los usuarios y el equipo de desarrollo sino también hacer más sencilla la construcción de escenarios y su descripción, facilitando su validación por utilizar el lenguaje del usuario.

Lenguaje Natural

El LEL permite registrar el vocabulario, reflejando las palabras particulares y más usadas en el dominio de la aplicación.

El uso de lenguaje natural para describir situaciones cumple con el objetivo de mejorar la comunicación con los stakeholders. El uso del LEL y Escenarios para la elicitación de requerimientos y su utilización a través de todo el proceso de desarrollo de software permite y facilita la validación con el usuario. El propósito principal del LEL es capturar el vocabulario de la aplicación y su semántica, posponiendo la comprensión de la funcionalidad de la aplicación para etapas posteriores.

El uso del LEL como soporte para la construcción de los escenarios mejora el problema de la ambigüedad en las descripciones basadas en lenguaje natural. Cada vez que se utiliza un símbolo del LEL en la descripción de un escenario, ese símbolo se convierte en hipervínculo, permitiendo así un vínculo entre las dos representaciones.

Símbolos del LEL

Este léxico se construye utilizando lenguaje natural y está compuesto, por símbolos que pueden ser objetos activos o Sujetos (realizan acciones), objetos pasivos u Objetos (las acciones se realizan sobre ellos), Verbos (acciones del sistema) y estados significativos del sistema [Leite97a].

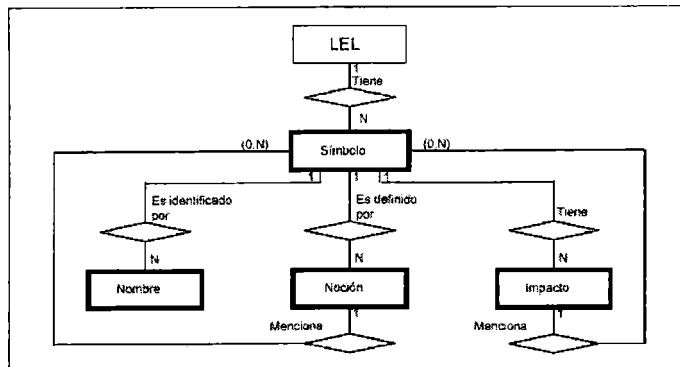


Figura 2.1

En la Figura 2.1 se presenta el diagrama Entidad-Relación [Chen76] para el modelo del Léxico Extendido del Lenguaje. Cada símbolo en el léxico tiene uno o más nombres o frases que lo identifican y dos tipos de descripciones, la noción y el impacto. Cada entrada puede contener una o más nociones y uno o más impactos [Leite96].

En la descripción de los símbolos deben cumplirse simultáneamente dos reglas básicas: **Principio de circularidad:** en la descripción de la noción o impacto de los símbolos se debe maximizar el uso de otros símbolos del léxico. De esta manera, el conjunto de símbolos determina una red, que permite representar al LEL mediante un hipertexto que puede ser navegado para conocer todo el vocabulario del problema.

Principio del vocabulario mínimo: se debe minimizar el uso de símbolos externos al lenguaje de la aplicación. De este modo, se acota el lenguaje al menor conjunto de símbolos posible. Si se utilizan símbolos externos, éstos deben pertenecer al vocabulario básico del lenguaje natural que se está utilizando [Leite97b].

Meta Modelo de LEL [Leite 00], es el utilizado para definir los símbolos en esta etapa.

LEL: Representación de los símbolos en el lenguaje del dominio de la aplicación

Sintaxis:

$$\{\text{Símbolo}\}_1^N$$

Símbolo: Entrada del léxico que tiene un significado especial en el dominio de la aplicación

Sintaxis:

$$\{\text{Nombre}\}_1^N + \{\text{Noción}\}_1^N + \{\text{Impacto}\}_1^N$$

Nombre: Identificación del símbolo. Más de uno representa sinónimos Sintaxis:

Palabra | Frase

Noción: Denotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo Sintaxis:

Sentencia

Impacto: Connotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo Sintaxis:

Sentencia

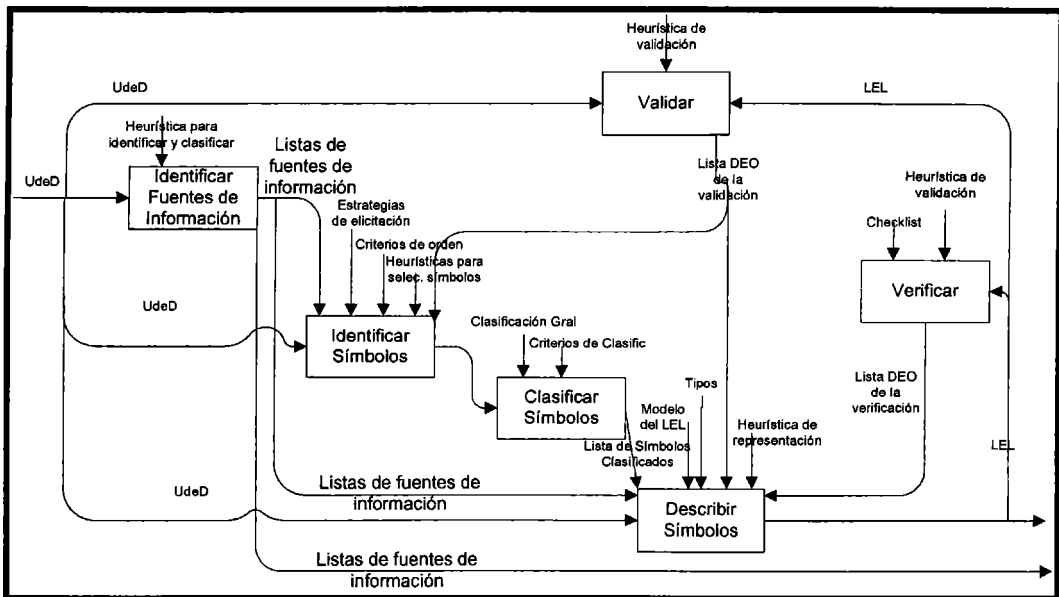
Sentencia está compuesto por Símbolos y No-Símbolos pertenecientes al vocabulario mínimo.

+ significa composición, {x} significa cero o más ocurrencias de x, () es usado para agrupamiento, | significa or, y [x] significa que x es opcional

2.3 Proceso de Construcción

El proceso de construcción consta de 6 etapas dependientes entre sí, y que, en algunos casos, se desarrollan en forma simultánea.

1. Identificar las fuentes de información.
2. Identificar los símbolos.
3. Clasificar los símbolos.
4. Describir los símbolos.
5. Verificar el LEL.
6. Validar el LEL con los clientes-usuarios.



SADT de las etapas del proceso de construcción del LEL
 Figura 2.2

El modelo SADT Figura 2.2 [Kaplan 00] del proceso de construcción del LEL muestra un flujo principal compuesto por cuatro tareas: Identificar las fuentes de información, Identificar los símbolos, Clasificar los símbolos y Describir los símbolos. Existe un ciclo entre las etapas de Validación y de Verificación. Después de verificar el LEL, el proceso vuelve a la etapa de Descripción de los símbolos donde las correcciones son realizadas teniendo en cuenta las listas DEO (Discrepancias, Errores y Omisiones). Después de la tarea de Validación, el proceso regresa a la actividad Identificar símbolos y/o a la actividad Describir símbolos, dependiendo de la lista DEO de validación, para hacer los cambios necesarios. Por ejemplo, mientras se describe un símbolo puede descubrirse un tipo mal asignado, lo que obliga a volver atrás en el proceso y reclasificarlo [Garcia99].

El primer paso para la construcción del LEL es la identificación de las fuentes de información. Estas fuentes están contenidas en el UdeD, cuyos límites son raramente

precisos, siendo incluso redefinidos, en algunos casos, a lo largo del proceso de desarrollo.

Las principales fuentes de información son los documentos y las personas involucradas en el dominio de la aplicación, libros de temas relacionados y otros sistemas disponibles en el mercado.

El siguiente paso es identificar los símbolos, para lo cual se seleccionan las estrategias de elicitación, se recolectan y organizan los símbolos. Las principales técnicas para recolectar información son la lectura de documentos, entrevistas, observaciones, reuniones, anotaciones del diseño de software y otras. La estrategia depende fundamentalmente de las fuentes identificadas en la etapa previa, aunque las entrevistas son las más utilizadas para la recolección de símbolos. En las entrevistas se hace más sencillo reconocer el vocabulario que el usuario emplea. Esta técnica se combina normalmente con lectura de documentos principalmente formularios, manuales y políticas organizacionales. Aplicando las técnicas seleccionadas, el ingeniero de requisitos identifica los símbolos y construye una lista de símbolos candidatos. Después de las primeras validaciones, la lista de símbolos candidatos generada inicialmente es ampliamente modificada. Se incorpora un gran número de símbolos y sinónimos, se eliminan símbolos erróneos y se aclaran símbolos confusos.

Posteriormente, se procede a clasificar los símbolos. Esta etapa asegura la integridad y la homogeneidad de las descripciones. Existe un criterio de clasificación general, que agrupa los símbolos en cuatro grandes categorías: Sujetos, Verbos, Objetos y Estados. Al finalizar esta etapa, cada uno de los símbolos de la lista queda identificado con un tipo que guiará su posterior descripción.

La descripción de los símbolos implica la definición de sus nociones e impactos, basadas en el modelo LEL y los tipos obtenidos en la clasificación. El ingeniero de requerimientos, de ser necesario, recurre al paquete de información previamente obtenido.

Los símbolos se deben describir según su tipo de la siguiente forma:

Para un símbolo del tipo Sujeto

Noción: define quién es el sujeto

Impacto: deben registrarse las acciones que ejecuta o recibe.

Para un símbolo del tipo Verbo

Noción: debe decir quién ejecuta la acción, cuándo ocurre y las actividades involucradas con dicha acción.

Impacto: debe identificar las situaciones que impiden la ocurrencia de la acción, qué otras acciones son disparadas en el ambiente y qué situaciones son causadas por la acción.

Para un símbolo del tipo Objeto

Noción: debe definir al objeto e identificar otros símbolos del mismo tipo con los cuales se relaciona.

Impacto: describe las acciones que pueden aplicarse a este objeto.

Para un símbolo del tipo Estado

Noción: define su significado y las acciones que llevan a ese estado.

Impacto: debe identificar otros estados y acciones que pueden ocurrir a partir de la situación específica.

En la figura 2.3 se presentan a modo de ejemplo de símbolos del LEL correspondiente al caso de estudio Agenda de Reuniones [Hadad98] un símbolo de tipo sujeto.

Una vez que se describieron los símbolos, se procede a la Verificación con el objeto de controlar que el LEL producido sea consistente y homogéneo.

Se realiza utilizando heurísticas de verificación y una lista de chequeo. Los defectos que se pueden encontrar realizando la verificación son [Kaplan 00]:

- De descripción: Símbolos mal descriptos, Símbolos incompletos y Descripción incompatible con el tipo.
- De clasificación: Clasificación incorrecta.
- De Identificación: Símbolos omitidos, Sinónimos incorrectos y Símbolos incorrectamente incluidos
- De Referencia: Falta de referencia a otros símbolos y mal uso de símbolos.

A partir de esta tarea se obtiene una lista DEO, que es utilizada posteriormente como control de la etapa Descripción de Símbolos.

Mientras se identifican y describen los símbolos tiene lugar una validación informal.

Estas validaciones tratan de chequear la lista candidata y, además, permiten al ingeniero de requisitos percibir el significado de los símbolos. Las siguientes validaciones son usadas para chequear el conocimiento obtenido y para mejorar el LEL mediante el esclarecimiento de dudas, corrección de las definiciones de los símbolos y la exclusión y agregado de símbolos. La tarea de Validación generalmente consiste en entrevistas estructuradas o reuniones con usuarios. Esta tarea es llevada a cabo fácilmente con los usuarios ya que éstos no tienen dificultades en la comprensión del LEL que está escrito en lenguaje natural y empleando su propio vocabulario.

Como resultado, se genera la lista DEO, que es utilizada para hacer correcciones en la Identificación y Descripción de los símbolos.

<p>CONVOCADO</p> <p>Noción: -Persona invitada a la <u>reunión</u></p> <p>Impactos: - Puede informar sus <u>horarios disponibles</u> - Puede dar el <u>aviso de concurrencia</u> - Debe estar en el lugar, <u>fecha</u> y <u>horarios</u> establecidos en la <u>convocatoria</u> - Puede dar el <u>aviso de no concurrencia</u> - Asigna un <u>reemplazante</u> en caso de no poder asistir y lo informa al <u>convocante</u> o <u>secretaria</u> - Puede definir el <u>material para repartir</u> - Registra el tiempo de traslado de la <u>reunión</u></p>
--

Símbolo del LEL de tipo Sujeto

Figura 2.3

Capítulo 3

Escenarios

Objetivos del Capítulo

El objetivo de este capítulo es introducir el concepto de escenarios, componentes y distintas formas de construcción.

Contenido

- 3.1-Introducción
- 3.2-Objetivos de los Escenarios
- 3.3-Componentes de los Escenarios
- 3.4-Técnica de construcción

3.1- Introducción

Los escenarios son descripciones parciales del funcionamiento del sistema. Los escenarios son usados para entender la aplicación y su funcionalidad: cada escenario describe una situación específica de la aplicación centrando la atención en su comportamiento [Leite00].

Un escenario es una descripción parcial y concreta del comportamiento de un sistema en una determinada situación. Es una descripción parcial, porque no necesita describir todas las características de las entidades involucradas, sólo se describe aquello que está relacionado con un comportamiento particular del sistema analizado. Más allá de estar acotados a un determinado comportamiento, describen todo el contexto que involucra a esa actividad: recursos del sistema, objetivos de los usuarios, contexto en que se desarrolla, entidades involucradas. Proveen una “foto” de cómo esa actividad se lleva a cabo.

Los escenarios no son formales, se pueden representar de diferentes maneras: narrativa textual, story boards, video mock-ups, prototipos escritos o situaciones físicas.

El nivel de detalle con que se describen depende de dos factores:

- Importancia de los hechos específicos.
- Fase en la que se encuentra el proceso de desarrollo.

Si bien cada escenario es una descripción parcial del comportamiento de la aplicación que se modela, ningún escenario es totalmente independiente del resto. Cada escenario tiene una relación semántica con otros escenarios [Booch 94].

3.2-Objetivos de los Escenarios

Los escenarios logran diferentes objetivos dependiendo de la fase donde se usen durante el proceso de desarrollo de software. En la fase de Ingeniería de Requerimientos, los objetivos del escenario son:

- capturar los requerimientos,
 - Evitar abstracciones orientadas a una solución.
 - Brindar una visión más amplia del comportamiento del macro sistema.
- proveer un medio de comunicación entre los stakeholders,
 - Asegurar la comunicación entre los miembros del equipo de desarrollo
 - Facilitar la validación de los requerimientos con el usuario
 - Comprometer al usuario en el desarrollo
 - Validar el LEL
- proveer un soporte para trazabilidad
 - Documentar los requerimientos
 - Capacitar a nuevos miembros del equipo para comprender la aplicación
 - Promover la evolución de los escenarios a medida que avanza el proceso de desarrollo
 - Generar casos de prueba
 - Vínculos con el LEL

3.3-Componentes de los Escenarios

Los escenarios están compuestos por: Título, Objetivo, Contexto, Recursos, Actores, Episodios, Excepciones y el atributo restricción.

Actores y Recursos son enumeraciones. Título, Objetivo, Contexto y Excepciones son sentencias declarativas, mientras Episodios es un conjunto de sentencias expresadas en un lenguaje simple que da una descripción operacional de comportamiento.

El modelo de escenarios debería ser visto como una guía estructural para:

- obtener un estilo homogéneo de descripción para el conjunto de escenarios.
- demostrar los diferentes aspectos que los escenarios pueden cubrir, y
- facilitar la verificación de escenarios (principalmente por un proceso automatizado)

Un escenario debe satisfacer un objetivo que se alcanza mediante la ejecución de los episodios. Éstos representan el curso de acción principal, pero incluyen también variaciones o alternativas posibles.

Mientras se ejecutan los episodios puede surgir una excepción, que señala un obstáculo para lograr el objetivo. El tratamiento de la excepción puede satisfacer o no el objetivo original. La comprensión de un escenario se ve facilitada por el uso de lenguaje natural, situaciones bien limitadas y el uso de subescenarios. Estos permiten manejar el problema de la explosión de escenarios señalada en [Cockburn95].

Un subescenario es usado cuando:

- se detecta comportamiento común en varios escenarios,
- aparecen cursos de acción condicionales o alternativos complejos en un escenario, y
- se detecta en un escenario la necesidad de mejorar una situación con un objetivo concreto y preciso

El atributo restricción es usado para caracterizar requisitos no funcionales aplicados a contexto, recursos y episodios.

Un escenario puede ser interrumpido por excepciones. Cada excepción se describe como una sentencia simple que especifica la causa de la interrupción. Si se incluye el título de otro escenario, la excepción será tratada por dicho escenario.

El contexto se describe detallando una ubicación geográfica, una ubicación temporal y precondiciones. Cada uno de estos subcomponentes puede ser expresado por una o más sentencias simples vinculadas por los conectores lógicos and, or.

Los episodios puede ser simples, condicionales y opcionales. Los episodios simples son aquellos necesarios para completar el escenarios. Episodios condicionales son aquellos cuya ocurrencia depende de una condición específica. La condición puede ser interna o externa al escenario. Las condiciones internas pueden ser debidas a precondiciones alternativas, restricciones de actores o recursos y episodios previos. Los episodios opcionales son aquellos que pueden ocurrir o no dependiendo de condiciones pueden no estar detalladas explícitamente.

Independientemente de su tipo, un episodio puede ser expresado como una acción simple o puede ser concebido en si mismo como un escenario, posibilitando de esa manera la descomposición de un escenario en subescenarios.

El modelo de escenarios provee la descripción de comportamientos con diferentes órdenes temporales. Una secuencia de episodios implica un orden de precedencia, pero

un orden no secuencial requiere agrupar dos o más episodios según la sintaxis del meta modelo de escenarios [Hadaad 99] mostrada en la figura 3.1.

<p>Escenario: descripción de una situación en el dominio de la aplicación. Sintaxis: Título + Objetivo + Contexto + {Recursos} ₁^N + {Actores} ₁^N + {Episodios} ₂^N + {Excepciones}</p> <p>Título: identificación del escenario. En el caso de un subescenario, el título es el mismo que la sentencia del episodio (ver en la definición de Episodios más abajo), sin las restricciones. Sintaxis: Frase ([Actor Recurso] + Verbo + Predicado)</p> <p>Objetivo: meta a ser alcanzada en el dominio de la aplicación. El escenario describe el cumplimiento del objetivo. Sintaxis: [Actor Recurso] + Verbo + Predicado</p> <p>Contexto: compuesto al menos por uno de los siguientes subcomponentes:</p> <p>Ubicación Geográfica: ubicación física del escenario. Ubicación Temporal: especificación de tiempo para el desarrollo del escenario. Precondición: estado inicial del escenario. Sintaxis: (Ubicación Geográfica) + {Ubicación Temporal} + {Precondición} donde Ubicación Geográfica es: Frase + {Restricción} donde Ubicación Temporal es: Frase + {Restricción} donde Precondición es: [Sujeto Actor Recurso] + Verbo + Predicado + {Restricción}</p> <p>Recursos: elementos físicos relevantes o información que debe estar disponible en el escenario. Sintaxis: Nombre + {Restricción}</p> <p>Actores: personas, dispositivos u organización que tienen un rol en el escenario. Sintaxis: Nombre</p> <p>Episodios: conjunto de acciones que describen al escenario y proveen su comportamiento. Un episodio también puede ser descrito como un escenario. Sintaxis (usando una representación tipo BNF): <episodios> ::= <serie de grupos> <serie de episodios> <serie de grupos> ::= <grupo> <grupo> <grupo no secuencial> <serie de grupos> <grupo> <grupo> ::= <grupo secuencial> <grupo no secuencial> <grupo secuencial> ::= <sentencia básica> <grupo secuencial> <sentencia básica> <grupo no secuencial> ::= # <serie de episodios> # <serie de episodios> ::= <sentencia básica> <sentencia básica> <serie de episodios> <sentencia básica> <sentencia básica> ::= <sentencia simple> <sentencia condicional> <sentencia opcional> <sentencia simple> ::= <sentencia episodio> CR <sentencia condicional> ::= IF <condición> THEN <sentencia episodio> CR <sentencia opcional> ::= [<sentencia episodio>] CR donde <sentencia episodio> es descrita como: ((([Actor Recurso] + Verbo + Predicado) ([Actor Recurso] + [Verbo] + Título)) + {Restricción})</p> <p>Excepciones: usualmente reflejan la falta o mal funcionamiento de un recurso necesario. Una excepción impide el cumplimiento del objetivo del escenario. El tratamiento de la excepción puede ser expresado por medio de otro escenario. Sintaxis: Causa [(Solución)] donde Causa es: Frase ([Sujeto Actor Recurso] + Verbo + Predicado) donde Solución es: Título</p> <p>Restricción: alcance o requisito de calidad referido a una entidad dada. Es un atributo de Recursos, Episodios básicos o subcomponentes de Contexto. Sintaxis: ([Sujeto Actor Recurso] + [No] Debe + Verbo + Predicado) Frase</p>

+ significa composición, {x} significa cero o más ocurrencias de x, () es usado para agrupamiento, | significa or, y [x] denota que x es opcional

Meta Modelo de Escenarios Figura 3.1

3.4- Técnica de construcción

Si bien existen varias técnicas de construcción de escenarios se toma como base para este trabajo la de Leite. En este proceso de construcción “Middle out” [Leite 00] propone realizar las actividades de la siguiente manera:

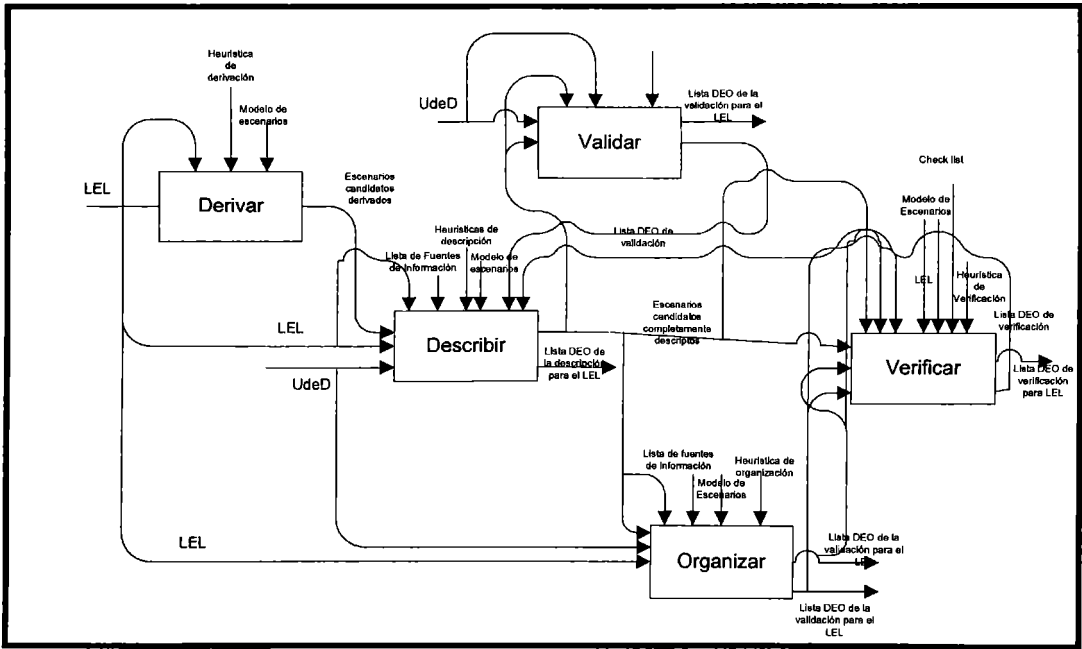
El proceso de construcción de escenarios comienza en el léxico del dominio de la aplicación, produciendo una primera versión de los escenarios derivada exclusivamente desde el LEL. Estos escenarios son mejorados utilizando otras fuentes de información y organizados para obtener un conjunto consistente de escenarios que representen al dominio de la aplicación. Durante o después de estas actividades propone realizar la verificación y validación con los clientes / usuarios para detectar Discrepancias, Errores y Omisiones (DEO).

En el SADT de la figura 3.2 [Leite 00], se puede ver las siguientes actividades:

1. Derivar
2. Describir
3. Organizar
4. Verificar
5. Validar

Aunque el proceso de Construcción de Escenarios de la 3.2 muestra un flujo principal compuesto por tres tareas: Derivar, Describir y Organizar, estas actividades no son estrictamente secuenciales. Algunas actividades pueden ser concurrentes debido al mecanismo de feedback, siempre presente en estas situaciones [Goguen92]. Hay un feedback cuando se llevan a cabo las actividades Verificar y Validar, retornando a la tarea Describir, donde se hacen correcciones en base a las listas DEO. La actividad Verificar ocurre después de describir los escenarios, y después de la organización cuando aparecen nuevos escenarios o se generan los escenarios de integración. Los escenarios son validados con los clientes / usuarios después de la verificación.

El proceso de construcción de escenarios podría producir también tres listas DEO que actuarán como feedback para el proceso de construcción del LEL para mantener la consistencia entre el vocabulario de los escenarios y el LEL propiamente dicho. “Estas listas DEO son creadas durante las etapas Describir, Verificar y Validar, y pueden ser descubiertas en este punto porque el conocimiento sobre el dominio de la aplicación comienza a incrementarse y mejorarse fuertemente.” Se pueden detectar posiblemente impactos omitidos, algunas discrepancias entre símbolos y raramente nuevos símbolos del UdeD.



SADT de las etapas del proceso de Construcción de los escenarios
 Figura 3.2

Derivar

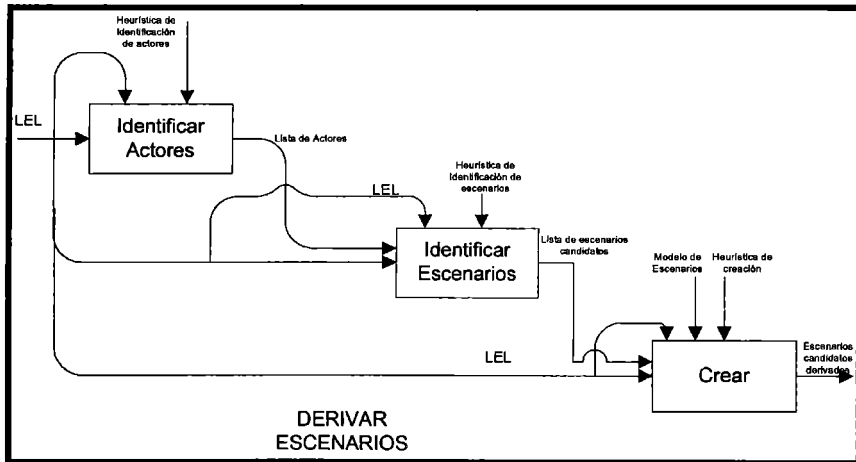


Figura 3.3

Esta tarea apunta a generar los escenarios candidatos a partir de la información del LEL usando el modelo de escenario y aplicando las heurísticas de derivación. La etapa de Derivación de escenarios consta de tres pasos: identificar los actores del UdeD, identificar los escenarios candidatos y crearlos usando el LEL. La Figura 3.3 muestra el modelo SADT [Leite 00] que descompone la actividad Derivar.

Identificar actores

Se identifican dentro del LEL los símbolos que representan actores del UdeD. Deben pertenecer al tipo sujeto. Los actores son clasificados en actores principales y secundarios.

Actores principales son aquellos que ejecutan acciones directamente en el dominio de la aplicación, y los actores secundarios son los que reciben y/o dan información, pero no comparten responsabilidades en la acción.

Identificar escenarios

Se extraen del LEL los impactos de los símbolos elegidos como actores principales y secundarios. Cada impacto representa un posible escenario, y es incorporado a la lista de escenarios candidatos. El título del escenario se construye con la acción incluido en el impacto, pero en infinitivo más un predicado también tomado del impacto.

Cuando diferentes actores ejecutan la misma acción, es muy probable que dos o más escenarios de la lista tengan el mismo título. En ese caso todos ellos excepto uno deberían ser eliminados.

Crear escenarios

El objetivo de esta etapa es construir el escenario aprovechando la información del LEL en la mayor medida posible, aplicando las heurísticas de creación.

El resultado de esta etapa son los escenarios candidatos derivados.

El contenido de cada impacto del símbolo de tipo Sujeto que llevó a un escenario candidato es analizado para encontrar símbolos del léxico del tipo Verbo.

Si el impacto contiene un símbolo Verbo:

El objetivo se define según el título y la noción del símbolo Verbo.

Los actores y recursos del escenario son identificados a partir de la información contenida en el símbolo Verbo y deberían ser símbolos de tipo Sujeto y Objeto respectivamente.

Los episodios se derivan a partir de cada uno de los impactos del símbolo Verbo.

Si el impacto no contiene un símbolo Verbo:

Los símbolos del LEL contenidos en el impacto son identificados y considerados como posibles fuentes de información.

El objetivo se define de acuerdo al título del escenario y el punto de vista de la aplicación.

Leyendo la definición completa de los símbolos mencionados, se seleccionan posibles actores y recursos. Los primeros se derivan de los símbolos de tipo Sujeto y los segundos de los de tipo Objeto.

Los episodios no se derivan del LEL. Su definición se posterga hasta una etapa posterior.

En ambos casos, la ubicación geográfica y temporal del contexto puede ser extraída del impacto del símbolo del LEL que originó el escenario (símbolo Sujeto). La información importante debería ser registrada en las precondiciones

del contexto podría estar disponible no sólo en ese impacto, sino también en la relación de secuencia con otros impactos del mismo símbolo.

Describir

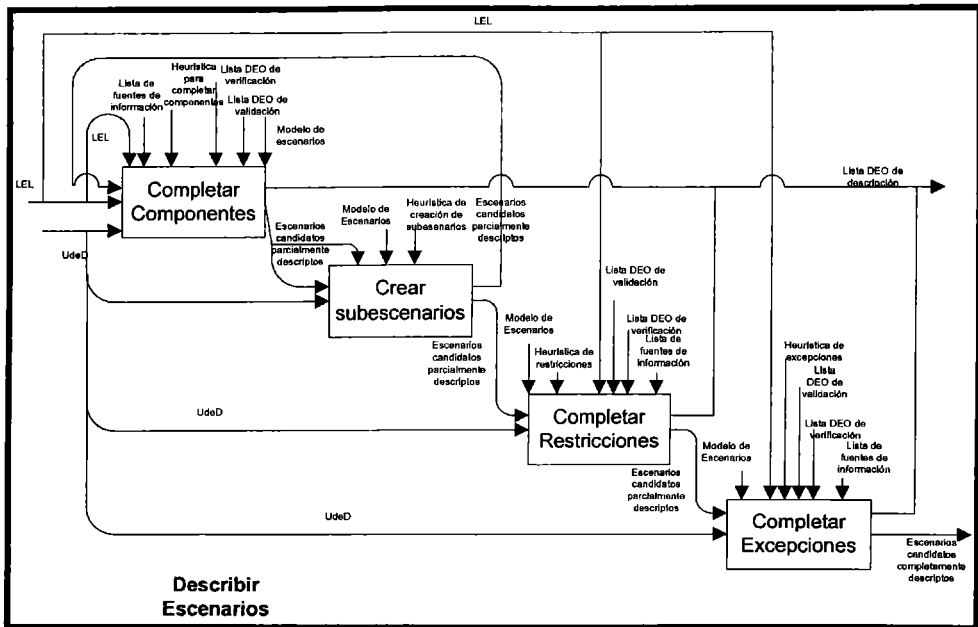


Figura F3.4

En esta actividad se completan los escenarios candidatos agregando información del UdeD utilizando las heurísticas de descripción y tomando como base el modelo de escenario y los símbolos del LEL. El resultado es un conjunto de escenarios candidatos completamente descriptos. Esta etapa consta de cuatro actividades, como puede verse en la figura 3.4[Leite 00]: Completar componentes, Crear subescenarios, Completar restricciones y Completar excepciones.

A partir de la Derivación, se obtiene un conjunto incompleto de escenarios que debe ser completado en dos sentidos:

- se debe mejorar el contenido de los escenarios candidatos
- y se pueden agregar escenarios nuevos.

Para lograr esto, si está disponible, la información obtenida durante el proceso de construcción del LEL pero no incluida en él, y si es necesario, el ingeniero de requerimientos puede retornar a las fuentes de información para obtener más datos y confirmar los existentes.

Como puede verse en la 3.2[Leite 00], existe un feedback entre las etapas de Verificación y Validación, representado por las listas DEO. Estas listas pueden originar cambios en la descripción del escenario.

En la actividad Describir también puede generarse una lista DEO para el LEL.

Completar componentes

Esta actividad generalmente se basa en entrevistas estructuradas, observaciones y lectura de documentos. La información obtenida permite confirmar y mejorar el

comportamiento modelado en los escenarios. Dado que algunos escenarios pueden estar parcialmente descritos en este punto, en esta etapa deberían revisarse las descripciones iniciales.

Después de terminar de describir el conjunto de escenarios, los episodios deben ser revisados cuidadosamente para confirmar el orden secuencial o detectar paralelismos, y para encontrar episodios opcionales.

Crear sub escenarios

Los sub escenarios son una posible solución al problema de la explosión de escenarios señalada en [Cockburn95].

Se utilizan cuando:

- se detecta comportamiento común en varios escenarios,
- aparecen cursos de acción condicionales o alternativos complejos en un escenario,
- se detecta en un escenario la necesidad de mejorar una situación con un objetivo concreto y preciso

Completar restricciones

Las restricciones se utilizan para caracterizar requisitos no funcionales aplicados a Contexto, Recursos y Episodios. Algunas pueden ser elicidadas desde el UdeD y otras pueden surgir examinando los episodios.

Completar excepciones

Como actividad final, se deben detectar los casos alternativos y excepciones. Algunas causas de excepción son elicidadas desde las fuentes de información mientras que otras pueden deducirse analizando los episodios y la no disponibilidad o malfuncionamiento de los recursos. Cuando se descubren las causas de una excepción, los ingenieros de requerimientos deberían investigar cómo es tratada la excepción en el UdeD, surgiendo así una nueva situación que podría necesitar ser descrita a través de un escenario separado.

Organizar

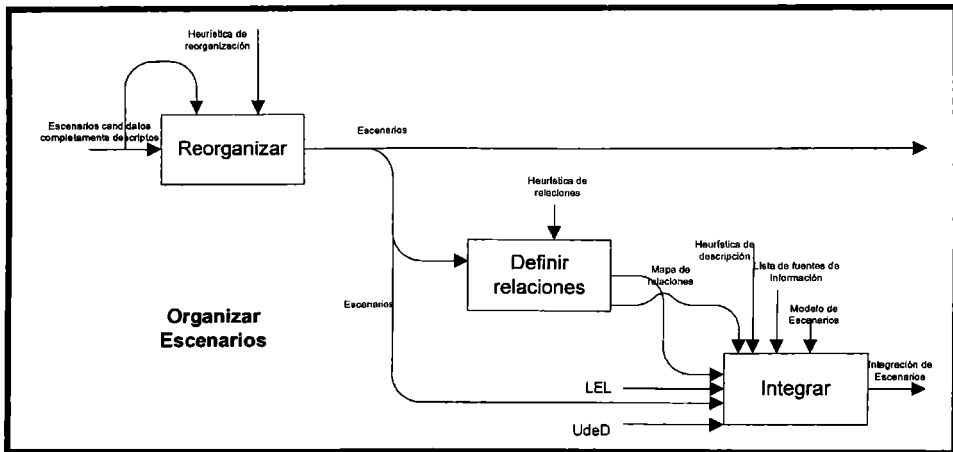


Figura 3.5

“Esta actividad es la más compleja y sistematizada del proceso de construcción de escenarios. Se basa en la idea de Escenarios Integradores, descripciones “artificiales” cuyo único objetivo es lograr que el conjunto de escenarios sea más manejable y comprensible.”[Leite 00]

Los escenarios integradores proporcionan una visión global de la aplicación.

Cuando las aplicaciones son grandes, el número de escenarios puede ser inmanejable y puede ocurrir que el ingeniero de requerimientos se vea inmerso en detalles, perdiendo la visión global de la aplicación. Para resolver este inconveniente, se propone la construcción de escenarios integradores, que dan una visión global de la relación existente entre varias situaciones, dado que cada episodio de un escenario integrador corresponde a un escenario. Por lo tanto, el propósito principal de los escenarios integradores es vincular escenarios dispersos,proveyendo una visión global de la aplicación, preservando siempre el formato de lenguaje natural utilizado para la descripción de los escenarios.

Así que, esta etapa comienza con los escenarios completamente descritos, y los organiza mediante la reorganización y la integración, esto puede observarse en la figura 3.5 [Leite 00]

Si bien la actividad Organizar está ubicada en tercer lugar , las actividades Verificar y Validar deben ocurrir antes de iniciar la organización de los escenarios completamente descritos. En otras palabras, esta es una tarea que se lleva a cabo sólo cuando se confía en tener un conjunto de escenarios bien definidos. Lo ideal es comenzar con las listas DEO vacías. No obstante, los escenarios completamente descritos pueden tener aún algunas debilidades importantes, como:

- falta de homogeneidad,
- problemas semánticos menores, y
- falta de perspectiva global

En esta etapa se utilizan, además, el LEL, el modelo de escenarios y es posible que sea necesario volver al UdeD.

Estos problemas pueden ser minimizados al aplicar las heurísticas de organización. Las heurísticas que se aplican en esta etapa contienen taxonomías de operaciones y relaciones, que se describen a continuación.

Reorganizar

Los escenarios pueden tener diferente grado de detalle o solapamiento de información, especialmente cuando son contruidos por más de un ingeniero de requerimientos. Esta situación se complica más a medida que se manejan más escenarios y el equipo de trabajo es más numeroso.

Las heurísticas de reorganización, aplicadas a escenarios completamente descriptos, tienen por objeto obtener escenarios consistentes y homogéneos.

Básicamente la reorganización consiste en juntar dos o más escenarios en uno, o dividir un escenario en uno o más.

La primera situación se da cuando un único escenario ha sido artificialmente dividido durante las etapas Derivar o Describir.

La segunda se plantea cuando el escenario contiene más de una situación.

Para establecer la necesidad de operaciones de composición/descomposición, algunas propiedades y relaciones entre escenarios deberían ser previamente identificadas.

Cuando un escenario presenta propiedades especiales, principalmente en sus episodios, se puede aplicar la descomposición. La composición, mientras tanto, puede ser usada cuando se descubren relaciones específicas entre escenarios: superposiciones, orden de precedencia o relaciones de jerarquía.

Debe notarse que tanto estas dos operaciones involucran el riesgo de degradación semántica. La descomposición crea dos o más escenarios a partir de uno. Este procedimiento no debe crear situaciones “artificiales”; en otras palabras, la descomposición debe ser aplicada sólo cuando los escenarios resultantes van a describir mejor al UdeD. La composición, por el contrario, reemplaza dos o más escenarios por uno. Esto hace parecer que una situación del UdeD es eliminada. Sin embargo, esto sólo puede ocurrir cuando uno de los escenarios describe algo que no es una situación real, o cuando una única situación real del UdeD fue descrita por dos escenarios.

Las operaciones de reorganización son sugeridas como guías para mejorar la comprensión y administración de los escenarios. Por lo tanto, esta etapa no debería verse como una actividad obligatoria, sino como una buena práctica.

A continuación se describen las diferentes operaciones [Leite 00] de reorganización:

Empotrar subescenarios: Puede ser aplicada cuando se detectan subescenarios no relevantes con pocas ocurrencias en otros escenarios. Esta operación incorpora los episodios del subescenario en cada escenario que lo menciona. El subescenario original se borra cuando se han empotrado todas sus ocurrencias.

Agregar subescenarios: Puede ser aplicada cuando se detecta un conjunto de episodios muy relevantes o un conjunto de episodios con diferente nivel de detalle en relación con el resto. Se crea un subescenario que contiene episodios de uno o más escenarios, y el grupo de episodios es reemplazado por el título del subescenario que los contiene.

Consolidar escenarios: Puede ser aplicada cuando dos o más escenarios están fuertemente superpuestos si sus objetivos y contextos son similares y tienen muchas coincidencias en los episodios.

Esta operación copia los episodios comunes de los escenarios originales a un nuevo escenario y crea nuevos episodios condicionales usando los episodios no compartidos, siendo la condición la parte correspondiente de la precondition del escenario original. Los escenarios originales son eliminados.

Dividir escenarios: Puede aplicarse cuando se detecta la presencia de varios episodios condicionales con la misma condición. Dividir produce dos nuevos escenarios que tendrán a la condición disparadora como precondition. Los episodios disparados son movidos al escenario correspondiente sin la condición, en tanto que los episodios no disparados por la condición son copiados a ambos escenarios. El escenario original es eliminado.

Fusionar escenarios: Puede ser aplicada a escenarios que presentan un orden de precedencia contigua o secuencial, objetivos complementarios y contextos acoplados. No puede existir una brecha temporal entre los escenarios. Esta operación copia los episodios de cada uno de los escenarios originales al nuevo escenario en el orden correspondiente. Los escenarios originales son eliminados.

Partir escenarios: Puede ser aplicada a un escenario si hay una brecha temporal entre los episodios o cuando se detecta un contexto temporal muy largo. Esta operación copia todos los episodios que preceden a la brecha temporal a un nuevo escenario, y aquéllos que le siguen a otro nuevo escenario. Luego, se elimina el escenario original. Las precondiciones del segundo escenario pueden reflejar el orden de precedencia secuencial respecto del primero.

Heurísticas que ayudan en la etapa Reorganizar Escenarios:

- Cuando se aplica composición, los objetivos del escenario resultante deben abarcar toda la situación.
- En el caso de consolidación el objetivo se extiende por generalización.
- Cuando se aplica descomposición, el objetivo del nuevo escenario puede ser menos global que el original. Solo se puede aplicar cuando el objetivo puede ser desacoplado.
- Consolidar debe aplicarse si pocos episodios condicionales aparecen en el escenario resultante.
- Cuando las ubicaciones geográficas y temporales podrían ser fragmentadas, se recomienda la operación Partir Escenarios

Puede verse que para aplicar las operaciones de composición se requiere la existencia de relaciones específicas que luego desaparecerán entre los escenarios. Las operaciones de descomposición, mientras tanto, generan relaciones entre los escenarios resultantes del mismo tipo que las eliminadas por la composición. El conjunto de relaciones resultante se usará en la siguiente etapa.

Definir relaciones entre escenarios

Varios autores han propuesto relaciones entre escenarios, tales como las asociaciones y asociaciones extendidas de [Jacobson94b], conexiones semánticas de [Booch94]. Vínculos temporales de [Dano97], y otros. Estas relaciones fueron identificadas y

utilizadas con otros propósitos. En este caso, el objetivo de las relaciones a definir es la construcción de escenarios integradores, así que se necesita un nuevo conjunto de relaciones. En esta etapa se identifican diferentes relaciones entre escenarios con el fin de integrarlos.

A continuación se presentan los tipos de relaciones identificados en [Leite00]:

Relación jerárquica: El comportamiento de un escenario se describe a través de un conjunto de episodios, que podrían ser acciones simples o subescenarios. Un escenario puede contener más de un subescenario o ninguno. Un subescenario puede estar incluido en uno o más escenarios y puede a su vez, contener subescenarios. Así, una jerarquía puede definirse como el conjunto compuesto por un escenario y sus subescenarios.

Relación de superposición: Se establece entre escenarios con porciones comunes. Esta relación se observa principalmente cuando varios episodios comunes se presentan en diferentes escenarios.

Relación de orden: Se establece entre dos jerarquías de escenarios cuando una precede a otra. Una jerarquía puede preceder a otras jerarquías estableciendo un orden temporal parcial con ellas. Se establece una secuencia cuando una jerarquía es precedida inmediatamente por otra. Puesto que la segunda puede a su vez estar precedida por una tercera, un gran número de jerarquías puede estar involucrado en una secuencia comenzando al menos con una jerarquía inicial.

Relación a través de excepciones: Se establece entre un escenario y aquellos escenarios que tratan sus excepciones. Cuando un escenario tiene una excepción, se describe su causa y si se especifica un tratamiento de la excepción, se menciona un escenario para tratarla. Un escenario puede estar relacionado con uno o más escenarios que resuelven excepciones. Un escenario que resuelve excepciones puede tratar excepciones que ocurren en diferentes escenarios.

Integrar

Esta actividad implementa el comportamiento “middle out” del proceso de construcción de escenarios [Leite 00], siendo los tres primeros pasos bottom-up y los últimos dos top-down. La integración de escenarios como tal podría ser vista como un procedimiento de cinco pasos:

- Construir jerarquías de escenarios
Una jerarquía es un grupo de escenarios relacionados por relaciones jerárquicas.
Cada jerarquía tiene un escenario raíz que no es referenciado como subescenario en otro.
- Detectar orden parcial entre jerarquías Es realizado en base a comparaciones entre precondiciones o recursos o restricciones de la jerarquía contra los títulos.

Si una precondición de la jerarquía identifica un estado inicial esto es resuelto por otra jerarquía. Lo mismo puede ocurrir con los recursos necesarios en una jerarquía y producido por otra. Episodio y restricciones de recursos pueden ser satisfecho por otra jerarquía, estableciendo así un orden parcial.

Estas restricciones pueden usarse para la comparación cuando ellas señalan a estados del recurso o ciertas condiciones que deben reunirse previamente.

Para establecer las precondiciones de la jerarquía, deben unificarse las precondiciones de todos los escenarios que pertenecen a la jerarquía, entonces, las precondiciones requeridas por un sub escenario y satisfecho por otro sub escenario deben eliminarse de la unión. Los recursos y restricciones de la jerarquía se obtienen de la misma manera. El objetivo de una jerarquía es simplemente el objetivo del escenario raíz y la comparación debe detenerse en cuanto un orden parcial se encuentre o se detectó como imposible.

Las comparaciones usadas en este paso son:

- La precondición de la jerarquía contra los títulos de otras jerarquías;
 - Las restricciones de la jerarquía contra los títulos de otras jerarquías;
 - Los recursos de la jerarquía contra los títulos de otras jerarquías;
 - La precondición de la jerarquía contra las metas de otras jerarquías;
 - Las restricciones de la jerarquía contra las metas de otras jerarquías;
 - Los recursos de la jerarquía contra las metas de otras jerarquías;
 - Las precondiciones de la jerarquía contra los episodios de otras jerarquías;
 - Las restricciones de la jerarquía contra los episodios de otras jerarquías;
 - Los recursos de la jerarquía contra los episodios de otras jerarquías.
- Construir secuencia de jerarquías
Comienza con la revisión de la relación de la clasificación parcial descubierta entre las jerarquías. Esto se hace para clarificar cual es el set de jerarquías que preceden a uno dado. Para hacer esto, se eliminan todos los ordenes parciales derivados por transitividad . Se dice que entre las jerarquías A, B y C hay un transitividad , si A y B precede a C pero también A precede B.
 - Construir el esqueleto de la integración
Se hace partiendo de las sucesiones que obtuvieron en el paso anterior. Este paso construye sólo los episodios componentes de los escenarios de integración. Primero se construye un escenario principal y después escenarios de nivel de integración intermedio. Todas las secuencias se colocan en el escenario raíz de integración con el indicador no-secuencial (#). Si una secuencia está compuesta por más de un escenario, se crea un fragmento para un escenario intermedio y un título nominal se usa en el escenario raíz y en el fragmento.
 - Proponer Título, Objetivo y Contexto para los escenarios integradores
Es el paso dónde la nueva información semántica necesita ser agregada. Esto se hace mirando los escenarios involucrados en el escenario integración. Este paso actúa como una clase de última verificación, el título y los objetivos deben escribirse conservando la cohesión. Las descripciones deben escribirse basándose en el LEL.

Los cuatro pasos iniciales se basan principalmente en información de sintaxis, mientras el último requiere más conocimiento semántico. En esta actividad los escenarios son agrupados en jerarquías que luego son agrupadas en secuencias. Finalmente, estas

secuencias se usan para construir escenarios integradores. La figura 3.6 muestra un ejemplo de un escenario integrador producido para el caso de estudio “Agenda de Reuniones”[Hada 98].

Debe notarse que dado que un escenario integrador es sólo producido para organizar un conjunto de escenarios, no usa las entidades recursos y actores.

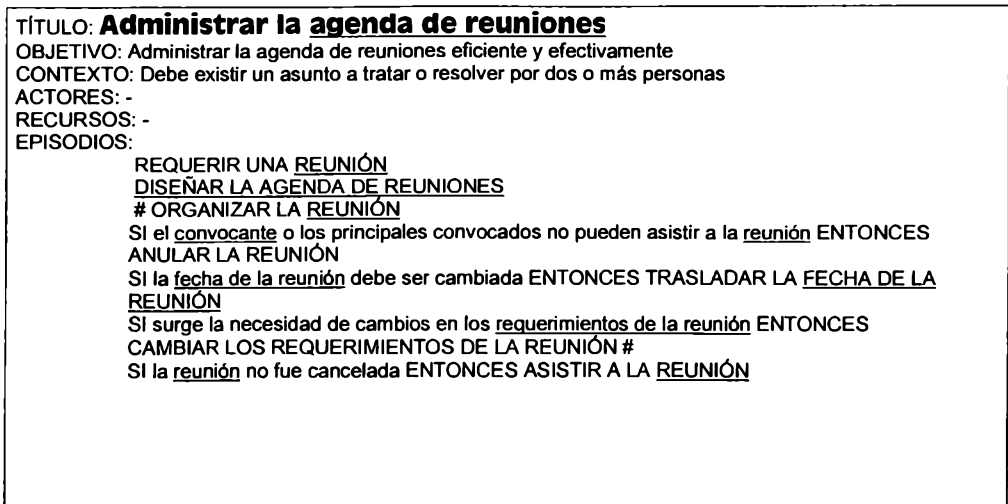


Figura 3.6

Verificar

Esta actividad es llevada a cabo al menos dos veces durante el proceso de construcción de escenarios; la primera sobre el conjunto de escenarios completamente descriptos y la segunda luego de la actividad Organizar. Esta etapa se realiza siguiendo un checklist con heurísticas de verificación de sintaxis, referencias cruzadas entre el LEL y los escenarios.

En el checklist se va registrando lo que se aprende en términos de verificación al utilizar el modelo de escenarios.

Como resultado, se producen dos listas DEO, una utilizada en la actividad Describir, y la otra en el proceso de construcción del LEL.

La verificación se realiza en tres niveles: intraescenarios, interesenarios y contra el LEL.

La verificación intraescenarios incluye:

Verificar sintaxis:

- chequear la completitud de cada componente
- chequear la existencia de más de un episodio por escenario
- chequear la sintaxis de cada componente según lo establecido en el modelo de escenario

Verificar relaciones entre componentes:

- chequear que cada actor participe en al menos un episodio
- chequear que cada actor mencionado en los episodios esté incluido en el componente Actores
- chequear que cada recurso sea usado en al menos un episodio

- chequear que cada recurso mencionado en los episodios esté incluido en el componente Recursos

Verificar semántica:

- chequear coherencia entre el Título y el Objetivo
- asegurar que el conjunto de episodios satisface el objetivo y está dentro del contexto
- asegurar que las acciones presentes en las precondiciones ya han sido realizadas
- asegurar que los episodios sólo contienen acciones a ser realizadas
- asegurar el mismo nivel de detalle dentro del escenario

La verificación interescenarios incluye:

Verificar existencia de subescenarios:

- chequear que cada episodio identificado como un subescenario existe dentro del conjunto de escenarios
- chequear que el conjunto de episodios de cada subescenario no haya sido incluido en otro escenarios
- chequear que cada excepción sea tratada por un escenario

Verificar contexto:

- chequear que cada precondición es un hecho incontrolable o es satisfecha por otro escenarios
- chequear la coherencia entre las precondiciones de los subescenarios y las precondiciones de los escenarios
- chequear que las ubicaciones geográfica y temporal de los subescenarios son iguales o más restringidas que las de los escenarios

Verificar equivalencia en el conjunto de escenarios:

- chequear que las coincidencias de objetivo sólo se den en situaciones diferentes
- chequear que las coincidencias de episodios sólo se den en situaciones diferentes
- chequear que las coincidencias de contexto sólo se den en situaciones diferentes

La verificación contra el LEL incluye:

- chequear que cada símbolo del léxico esté identificado
- chequear el uso correcto de los símbolos del léxico
- chequear que los actores sean preferentemente símbolos del tipo Sujeto
- chequear que los recursos sean preferentemente símbolos del tipo Objeto
- chequear que los impactos de los símbolos del tipo Sujeto estén cubiertos por los escenarios

Al usar las listas DEO de verificación, se modifican los escenarios y el LEL. Si se necesitan correcciones importantes, podría requerirse una nueva verificación. Cuando

un ciclo Describir-Verificar ha finalizado, la actividad Organizar debería proveer un conjunto más grande de escenarios, que a su tiempo serán verificados iniciando un posible ciclo Describir-Organizar-Verificar. Esta actividad es efectuada actualmente por medio de inspecciones [Doom98].

Validar

Los escenarios son validados con los clientes/usuarios efectuando entrevistas estructuradas o reuniones. Durante la validación, debe prestarse especial atención al componente Dudas para aquellos escenarios que lo incluyen.

Es importante destacar que aunque se use una descripción estructurada, los escenarios son escritos en lenguaje natural, empleando el propio vocabulario de los clientes/usuarios y describiendo una situación específica bien delimitada.

La tarea de Validación permite la detección de DEO en los escenarios. Los errores son principalmente detectados al leer cada escenario al cliente/usuario. Algunas omisiones pueden aparecer también durante esa lectura, y otras al cuestionar la falta de información o detalles en los escenarios. Las discrepancias pueden aparecer durante las entrevistas pero surgen generalmente más adelante, cuando se analiza la información recolectada.

Debe notarse que el objetivo principal de la Validación es confirmar la información elicitada y detectar DEO. Sin embargo, un efecto lateral de esta etapa es la elicitación de nueva información. La validación de un conjunto de escenarios, después de un proceso de verificación, debería confirmar que las situaciones del UdeD han sido modeladas de acuerdo con la percepción de los clientes/usuarios del UdeD a cargo de la lectura y discusión de los escenarios.

Clase Responsabilidades Colaboraciones - CRC

Objetivos del Capítulo

El objetivo de este capítulo es introducir el concepto de CRC y la heurística para derivarlas partiendo de los símbolos del LEL y los Escenarios.

Contenido

4.1-Concepto de CRC.

4.3-Heurísticas de derivación de CRC.

4.1- Concepto de CRC

El producto de esta etapa es un conjunto de tarjetas, asociadas a cada una de las clases, en las que se especifican las responsabilidades y sus colaboraciones con otras clases, como se plantea en [Wirfs-Brock R 90].

Las responsabilidades son sentencias de alto nivel acerca de las acciones que realiza un objeto como también del conocimiento que él mantiene y provee. Por lo tanto, constituyen el grupo de servicios que brinda la clase correspondiente. El conjunto de las responsabilidades de cada clase representa la forma en que se distribuye lo que el sistema global debe realizar.

Los objetos no están aislados sino que colaboran unos con otros. Una colaboración es una solicitud hecha por un objeto a otro. Para identificarlas, se debe examinar cada par clase-responsabilidad con el fin de determinar si es necesario que la clase interactúe con otra/s para llevar a cabo esa responsabilidad. Para ello, puede ser apropiado realizar las siguientes preguntas, para cada responsabilidad definida en una clase:

- 1) ¿La clase es capaz de concretar la responsabilidad por sí misma?;
- 2) Si no es así, ¿de qué otra clase puede adquirir lo que necesita?;
- 3) ¿Qué otras clases necesitan la información que la clase conoce o el resultado que produce?.

En este punto se puede verificar si se ha obtenido alguna clase que no interactúe con las demás, ya que en este caso debería eliminarse. Sin embargo, antes de hacerlo, es necesario revisar las etapas anteriores con el fin de comprobar que tales interacciones no se hayan pasado por alto.

Una tarjeta CRC contiene el nombre de la clase que representa, sus responsabilidades y las clases con las que colabora.

4.2- Heurísticas de derivación de CRC.

Para derivar CRC s [Lonardi 01] propone una serie de heurísticas que se aplican partiendo de los modelos basados en lenguaje natural.

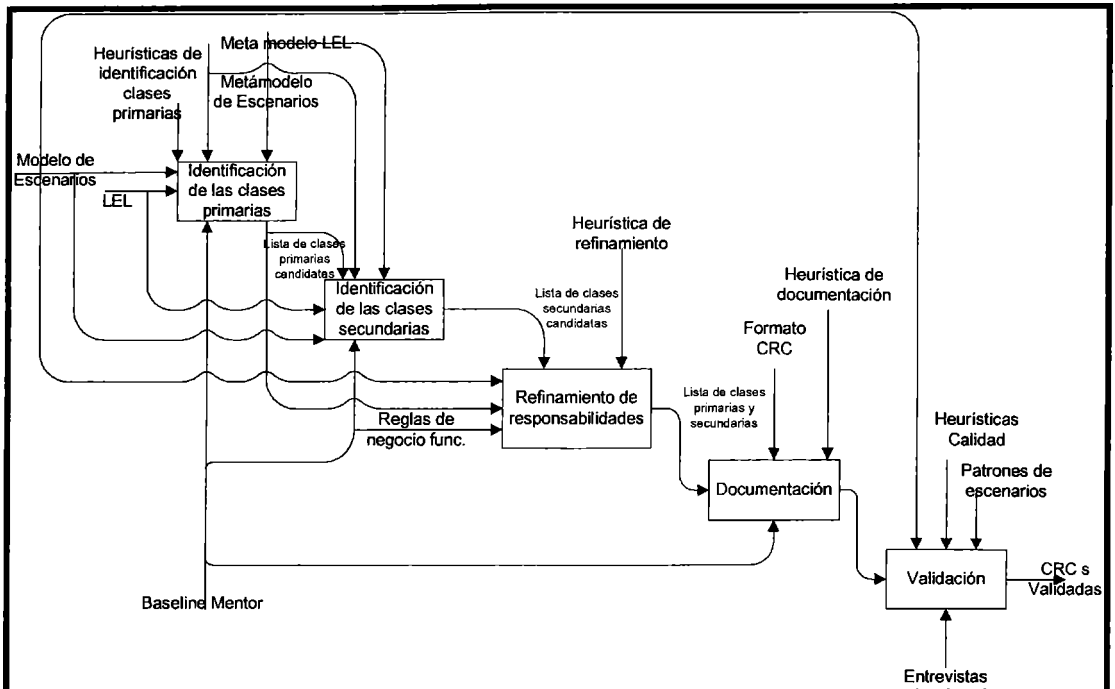
La definición se hace principalmente partiendo del LEL, utilizando también el modelo de escenarios para completar las responsabilidades y colaboraciones de una clase así como también el modelo de reglas funcionales para reflejar las políticas de la organización ya sea agregando nuevas clases o modificando existentes.

En este caso las reglas funcionales no serán tenidas en cuenta.

Al analizar el Macrosistema, se puede distinguir entidades que participan activamente y entidades que son utilizadas para poder llevar a cabo alguna acción. Esta distinción se refleja en la clasificación de actor / recurso considerada en el modelo de escenario y objeto / sujeto considerada en el modelo del LEL. Como el modelo de clases que se esta definiendo refleja al Macrosistema, tendrá estas mismas categorías. De ahí se puede hablar de clases Primarias y Secundarias.

En la primera etapa de la estrategia propone identificar las clases primarias y luego las clases secundarias. La información se modela en tarjetas estilo CRCs [Beck 89]. La identificación se hace principalmente a partir de los términos del LEL, pero luego se refinan las colaboraciones siguiendo los escenarios.

Para poder asignar responsabilidades Lonardi dice que se debe tenerse en cuenta, qué significa el impacto dentro del término según sea su clasificación, ya que debido al significado del impacto para cada término la asignación puede variar. El siguiente SADT figura 4.1 [Lonardi 01] muestra las entradas y salidas de las sub_ etapas.



Modelo SADT – Derivar CRC s
Figura 4.1

Para la descripción de la heurística Lonardi propone utilizar el siguiente patrón:

Identificador: Se utilizará una Sigla para identificar a la heurística. Todas las heurísticas relacionadas a la definición de CRCs empiezan con HC, las del modelo lógico como HM y las relacionadas con la definición de límites con HL.

Esta sigla es utilizada para las relaciones de trace que surgen en el modelo a partir de la aplicación de las heurísticas.

En este caso solo se tomaran las heurísticas propuestas para derivar CRC s por ser las de interés para este trabajo.

Descripción: Presenta la descripción y justificación de la heurística. Pueden aparecer sub-heurísticas, que se identifican con un orden relativo a la heurística a la que pertenece.

Ejemplo: En general, se ejemplificarán las heurísticas con el caso de estudio "Círculo de ahorro

Para mejorar la calidad tanto del proceso de construcción como del producto final del modelo de objetos de software, se pueden utilizar distintos principios de calidad de modelado, como por ejemplo [Meyer 97; Cockburn 99] en donde se mencionan heurísticas para modelar clases con calidad, principios para establecer asociaciones entre clases, y cuestiones de reusabilidad, entre otros. En [Lonardi 01] se mencionan aquellas cuestiones en donde el uso de los modelos de escenarios, LEL y reglas de Negocio ayudan en este proceso. Por consiguiente, esta estrategia debe ser considerada junto con principios generales de modelado de objetos.

La aplicación de estas heurísticas genera relaciones de traceability entre los modelos generados y los generadores, permitiendo obtener información que justifica la existencia de las CRCs y sus responsabilidades.

4.2.1 Heurísticas de Identificación de clases primarias y sus responsabilidades

En este punto se toman las heurísticas propuestas y se mantiene su formato en cuanto al patrón utilizado para su descripción.

Se entiende por clases primarias aquellas que representan una abstracción de comportamiento significativo para el UdeD, es decir aquellas que modelan parte de la funcionalidad del mismo. Como todavía no existe un modelo de software en donde existen factores de diseño que puede modificar la solución, el comportamiento más relevante es el realizado por las personas, organizaciones, o sectores del Macrosistema que se está modelando. Estas entidades son los Actores de los Escenarios y los Sujetos del LEL.

H1-Modelar los sujetos del LEL como clases candidatas

Descripción: En los escenarios estos sujetos son actores. Los sujetos corresponden a roles o partes de una organización. Al estar definiendo un modelo del Macrosistema, estas entidades son las que realizan las principales acciones del mismo por lo que su modelado como clases es automática.

Ejemplo: El término del LEL Administradora clasificado como Sujeto corresponde a la organización que se está modelando, por consiguiente existirá una clase “Administradora”. El término del LEL Adherente que representa a una persona con un determinado rol dentro de la organización (pertenecer a un grupo de Ahorro), es también modelado como una clase.

H2-Determinar las responsabilidades de los impactos.

Descripción: Se analiza cada uno de los impactos del LEL correspondientes al término para el cual se ha definido la clase. Al ser clases que han surgido a partir de los Sujetos del LEL, por definición, todos los impactos son acciones que el sujeto realiza, entonces se define al impacto como una responsabilidad del objeto.

Existe un caso a tener en cuenta, y es cuando el impacto representa a una frase verbal que ha sido modelada como un término del LEL. Este será analizado en la identificación de las clases secundarias, pudiendo o no ser modelado como una clase, de todos modos, en principio la responsabilidad queda asignada en esta nueva clase y colaborará con la clase que representa a la acción. Puede haber modificaciones en las siguientes etapas,

dependiendo de como se modelará el término del LEL que aparece en esa responsabilidad.

Ejemplo: Para la clase Administradora el impacto Expulsar Adherente se modela como responsabilidad. En el impacto Evaluar Licitación existe un término Licitación en el LEL, por el momento queda como responsabilidad de la clase Administradora sin embargo es candidata a ser una clase secundaria ya que aparece un término del LEL correspondiente a una Frase Verbal que define la Licitación.

4.2.2 Heurísticas de Identificación de clases secundarias y sus responsabilidades

Se entiende por clases secundarias aquellas que representan recursos para que las clases primarias puedan realizar sus responsabilidades. Son entidades pasivas dentro del Macrosistema, generalmente son repositorios de datos, a excepción de las clases provenientes de Frases Verbales.

H3- Analizar los términos del LEL que corresponden a Objetos, Estados y Frases verbales.

Descripción: Se construye una lista con los términos del LEL que representen objetos, frases verbales y estados. La mayoría de estos términos figuran en cada una de las responsabilidades de las clases definidas previamente. Sin embargo, pueden aparecer más términos que no están directamente relacionados con los términos que se han convertido en clase candidata. Se analiza cada uno de estos términos para determinar si serán modelados o no como objetos.

H3.1- Objetos del LEL como atributos o clases

Descripción: para cada objeto se analizan sus impactos:

- Si el término no tiene un comportamiento relevante se modela como un atributo. Para esto se busca en las nociones e impactos de los términos correspondientes a Sujetos que han sido modelados como clases, para determinar la clase a la que corresponde. Si existen dudas de la clase que lo debe contener, se revisan los escenarios en donde aparece este término. En etapas posteriores puede convertirse en objeto, pero en esta etapa no justifica que lo sea. Para decidir si será modelado como atributo se debe analizar si los impactos de los términos no están describiendo acciones similares a alguna clase primitiva (por ejemplo, Integer, String) es decir, que si bien es una abstracción propia del sistema, en términos de objetos, no justifica ser una nueva clase [Meyer'97]. Generalmente ocurre que es un término de valor simple o un conjunto de términos de valores del mismo tipo.
- Si el término tiene comportamiento significativo para el Macrosistema se convierte en clase, ya que define una porción de comportamiento importante que puede ser modelado como una abstracción independiente necesaria para que las clases primarias puedan cumplir sus responsabilidades.

Ejemplo: El término del LEL Cuota Mensual, no tiene un comportamiento propio que justifique que sea modelado como un objeto del modelo conceptual.

Por otro lado, el término Comunicación Fehaciente es un término que representa un mecanismo de comunicación muy utilizado dentro de la Administradora, por esta razón es modelado como clase para independizar las diferentes formas de comunicación de las entidades que lo utilizan.

H3.2- Frases verbales del LEL como clases o responsabilidades

Descripción: Si el término corresponde a una acción, se debe analizar la posibilidad o no de representarla como una clase. Esto ocurre generalmente cuando tiene características propias que no pueden ser asignadas a otras clases.

Luego, al pasar al diseño preliminar y marcar los límites del software, parte de la funcionalidad de los objetos quedan adentro y parte afuera de estos límites. Por esta razón, al desaparecer los objetos, responsabilidades que antes correspondían a algunos objetos del modelo conceptual son reasignadas a otro objeto, o bien son modeladas como un objeto de software.

Existen estrategias de diseño que, ante determinadas características de esa acción, sugieren que sea modelada como un objeto, como por ejemplo el “*pattern strategy*” [Gamma 94]. El modelar un comportamiento como una clase, permite refinar al comportamiento a través de una jerarquía de herencia sin que se vean involucradas las clases que lo utilizan. Estos objetos son llamados “coordinadores o controladores” [Wirfs 95], y desde una perspectiva de Negocios y Orientación a Objetos, son conocidos como “objetos del proceso de negocio” [Jacobson 95]. En general, debieran surgir a partir de una decisión de diseño y no del modelado del Macrosistema. Si una clase es descrita solo por la capacidad de hacer una acción, ¿no es más adecuado que se convierta en una responsabilidad de otra clase? Solo podría justificarse que esa acción sea modelada como clase si representa un comportamiento complejo, o si guarda propiedades que le corresponde a esa acción y no a ninguna otra clase.

Ejemplo: el término Licitación tiene un conjunto de impactos que justifica sea representada como clase ya que describen a una de las principales acciones de la Administradora. Además, se necesita guardar información propia de la Licitación, que no puede almacenarse en otra clase. Por esta razón se modela como clase Licitación, la cual colaborará con la clase administradora.

Sin embargo, el término Expulsar Adherente no tiene impactos que justifiquen sea modelado como clase, por lo que queda como responsabilidad de la clase Administradora.

H3.3- Estados como clases o atributos

Descripción: Si el término corresponde a un estado, por definición del mismo, ese estado afecta a todo el Macrosistema o a algunas de sus componentes, reflejados como términos del LEL. El hecho de haber sido modelado como un término diferente, implica que tiene nociones o impactos que lo caracterizan y distinguen.

Si el término del LEL al cual afectan fue modelado como clase, se analiza primero si el estado no puede convertirse en un atributo (con una responsabilidad que lo modifique), pero si tiene impactos que pueden ser modelados como responsabilidades, entonces es candidato a ser modelado como una clase diferente.

Ejemplo: En el caso de estudio del Meeting Scheduler existe el término posible reunión ya que contiene características propias que la diferencian de una reunión. Se modela como clase ya que la reunión también lo es y posibleReunion tiene responsabilidades bien diferenciadas de la reunión. Sin embargo, en el Sistema de Pasaportes los términos correspondientes a Pasaporte inválido y Pasaporte renovado se modelan como atributos de pasaporte ya que indican un estado del mismo sin anexar comportamiento extra. (en el caso de pasaporte inválido, invalida al Pasaporte pero no anexa comportamiento)

H4- Definir las responsabilidades de las clases secundarias utilizando el LEL

Descripción: Al igual que con las clases primarias, se analiza los impactos de los términos del LEL correspondientes a las clases definidas. Es necesario tener en cuenta la categoría del término para el cual se está determinando las responsabilidades ya que según sea la categoría cambia el sentido de los impactos pudiendo, en algunos casos, agregar responsabilidades a otras clases.

H4.1- Determinar las responsabilidades de los términos correspondientes a objetos

Descripción: los impactos de un Objeto describen qué acciones se le pueden aplicar al mismo. En términos del paradigma de orientación a objetos, esto no necesariamente implica que siempre el objeto va a tener una responsabilidad para responder a estas acciones. Depende de la semántica del impacto. En general, siempre habrá una responsabilidad asociado al impacto, pero a veces puede ser que el objeto sea un recurso para otro (sujeto u objeto) y no tenga una responsabilidad asociada. En este caso se debe buscar otro término dentro del impacto que representa a la clase en cuestión o analizar el conjunto de escenarios en donde aparece este término para buscar las acciones asociadas a este impacto y determinar las clases involucradas.

Ejemplo: Si consideramos el impacto de una Solicitud “es llenada”, nos damos cuenta que se debe definir una responsabilidad para la clase Solicitud que permita ser llenada, en este caso una responsabilidad básica de creación y modificación de los valores de la solicitud. Por otro lado, si analizamos el impacto “es pagada mensualmente” del término Cuota, nos damos cuenta que no debe modelarse como una responsabilidad “serPagada” ya que la cuota no lo tiene que registrar, sino que esto queda registrado en el sujeto que realizó la acción o en algún objeto encargado de registrar el pago de Cuotas, en este caso se asocia como una cuotaPaga a la clase Adherente.

H4.2- Determinar las responsabilidades de los términos correspondientes a frases verbales

Descripción: en el caso de las frases verbales, los impactos describen restricciones de la acción y nuevas acciones. Si es una restricción de las acciones pueden derivar responsabilidades así como también pueden afectar a responsabilidades existentes. Si es una acción, se debe analizar si no existe un término del LEL en el impacto que sea el ejecutor de la acción. En este caso existe una colaboración entre ambas clases. Si no es el caso se modelará como responsabilidad de la clase que está representando a la clase verbal.

Ejemplo: analizando el término Sorteo para definir las responsabilidades de la clase correspondiente, se ve que el impacto “se define el adherente ganador...” queda como una responsabilidad de la clase. Por otro lado, el impacto “el adherente que gana el sorteo puede rechazar el bien tipo...” establece una acción de rechazo para el Adherente y una aceptación de rechazo para el Sorteo que a su vez lo manejará con la Administradora.

H4.3- Determinar las responsabilidades de los términos correspondientes a estados

Descripción: si el término es un estado se aplican algunas de las dos heurísticas presentadas anteriormente (H4.1 y 4.2) dependiendo de la clasificación del término al que está afectando ese estado.

Ejemplo: En el caso de la clase posibleReunion del Meeting Scheduler se aplica la heurística H4.1 estableciendo responsabilidades que permiten determinar los datos básicos de la Reunión que son utilizados por las clases Secretaria y Convocante

4.2.3. Heurísticas generales para la definición de las clases

Tanto para la identificación de las clases primarias como para las secundarias propone tener en cuenta las siguientes pautas:

H5- Elegir el nombre más corto y significativo para definir a la clase

Descripción: El nombre debe ser el sinónimo más corto que se haya definido para cada término del LEL, excepto que sea similar a otro término que también se ha modelado como clase. En [Cockburn 99] se presentan heurísticas simples para elegir un nombre correcto teniendo como premisa que se pueden hacer suposiciones sobre una clase basándose en el nombre, el cual debe reflejar claramente la abstracción que la clase representa. Un nombre no debiera estar escrito como una frase verbal, ya que esto es indica una posible mala definición de clases [Meyer 97].

Ejemplo: el término Administradora tiene el sinónimo Administradora del Circulo de Ahorro para la Compra de un automotor. Este nombre es muy largo y como solo se está modelando ese Macrosistema aparecería como redundante. Se elige como nombre de la clase a la palabra Administradora.

H6- Definir los nombres de las responsabilidades con verbos en infinitivo

Descripción: Las responsabilidades deben estar escritas en verbos infinitivos, dejando bien en claro la semántica de la acción.

Ejemplo: El término Expulsar Adherente tiene el sinónimo Adherente es expulsado este término ha sido modelado como responsabilidad de la clase Administradora por lo que se utiliza el sinónimo infinitivo. El término correspondiente a la Administradora tiene un impacto que es Aceptación de solicitud de adhesión tiene. La responsabilidad será aceptarSolicitud.

H7- Analizar si el impacto no refleja más de una acción

Descripción: Si un impacto describe más de una acción relacionada debe analizarse la posibilidad de que se conviertan en dos responsabilidades. Para esto se puede analizar los escenarios en donde aparecen los términos para poder decidir si se justifica modelar como dos responsabilidades diferentes o son variantes de un mismo comportamiento. Esta situación no es común por la propia definición de los impactos de los términos del LEL.

Ejemplo: en el término Adherente aparece Pagar cuota en Banco o pagar cuota vencida. *Serán modeladas como dos responsabilidades diferentes.*

H8- Asignación de responsabilidades redundantes

Descripción: Con respecto a la asignación de las responsabilidades a las nuevas clases, se debe tener en cuenta la redundancia del LEL. Esto significa que una misma acción puede estar descrita en distintos términos pero pertenecer a uno (esto sucede cuando pertenece por ejemplo a un Sujeto y una Frase Verbal). Se debe establecer claramente la colaboración, quien ejecuta la acción y quien es el cliente o si ambos son responsables.

Ejemplo: esta situación fue reiterativa en el caso de la Auto-aplicación de la metodología. Por ejemplo se puede tomar el impacto: Realizar la detección de excepciones Este impacto se encuentra en varios símbolos del LEL, a saber: Construcción de escenarios, Ingeniero de requisitos y Construcción del LEL. Claramente se puede determinar que la responsabilidad corresponde a la clase Ingeniero de Requisitos.

4.2.4 Refinamiento de colaboraciones y responsabilidades

Al definir las responsabilidades de las clases primarias y secundarias a partir de los impactos del LEL, aparecen naturalmente las clases que colaboran ya que son los términos del LEL que están en los impactos analizados que se han convertido en clases. En esta etapa propone refinar las colaboraciones y responsabilidades, utilizando los escenarios

H9- Refinar responsabilidades y colaboraciones analizando el modelo de escenarios

Descripción: Construir una lista con los escenarios en donde aparece cada una de las clases definidas. Al identificar las responsabilidades a partir de los impactos del LEL, estas pueden ser muy generales. Algunas de las responsabilidades deben ser refinadas a partir de los escenarios definiendo las acciones o sub-responsabilidades asociadas. Se analizan los episodios y se refinan las responsabilidades con las acciones que la clase lleva a cabo para cumplir con esa responsabilidad. Al refinar la responsabilidad se refinan las colaboraciones porque se definen las colaboraciones a nivel de acciones de la clase. Para cada acción se especifica la colaboración y se verifica si existe en la otra clase alguna responsabilidad que puede llevar a cabo la colaboración.

Ejemplo: analizando el escenario “Licitar” en el cual aparece la responsabilidad licitar de la clase Adherente, se determina que esta responsabilidad puede ser refinada como crear sobreLicitacion y enviarSobre. Para la primera colabora con la clase sobreLicitacion , mientras que para la segunda con la Administradora. Se verifica que para ambas clases existan responsabilidades que las cumplan. En este caso, se agregó crearSobre a la clase sobreLicitacion.

H9.1- Analizar las estructuras de los escenarios que indiquen cursos alternativos de acción

Descripción: En esta etapa se debe analizar si asociadas a cada responsabilidad que aparece en los escenarios hay excepciones, restricciones o cláusulas condicionales para evaluar las distintas alternativas de comportamiento. Se refina la colaboración o se generan nuevas responsabilidades para poder llevar a cabo el comportamiento detallado en el curso alternativo.

Ejemplo: analizando el escenario Evaluar Licitación se observa que existe una sentencia condicional que indica: Si hay ofertas similares se toma el primero en el orden establecido en el Sorteo.... En este caso, si bien no se crea una nueva responsabilidad, se refina la responsabilidad Evaluar Licitación de la clase Licitación para que obtenga de la clase Sorteo el orden del sorteo y determine el ganador (posteriormente, durante la definición del modelo de objetos se podrá decidir si este refinamiento se convierte o no en un método privado de la clase que realice esta acción)

H9.2- Agregar responsabilidades básicas

Descripción: En los escenarios se encuentran nuevas responsabilidades que tratan aspectos de acceso y modificación del estado interno del objeto. Estos no siempre figuran en el LEL porque no encierran una semántica significativa para el sistema. Sin embargo, desde el punto de vista del modelo de solución es una responsabilidad de la clase. Si no se quiere agregar complejidad en esta etapa se puede dejar para la etapa del Diagrama de Objetos.

Ejemplo: en los escenarios Armar Grupo y Asegurar Adherente, se observa que la Administradora toma los datos de la Planilla de Adhesión Se crea una responsabilidad básica darDatos en la clase la Planillas de Adhesión con los datos necesarios.

H10- Definición de clases que no pertenecen al LEL

Descripción: Pueden modelarse como objetos entidades que aparecen en los escenarios interactuando con objetos pero que no figuran en el LEL por no ser términos propios del Macrosistema. Las responsabilidades se definen analizando los escenarios en donde aparecen.

Ejemplo: en el caso de estudio del “Meeting Scheduler” surge el directorio telefónico , que si bien no aparece como término en el LEL por ser un término de uso común, es utilizado como recurso en algunos escenarios.

H11- Analizar verbos con semántica de registrar

Descripción: Se debe analizar los tipos de verbos cuya semántica es registrar información. Si los impactos de un término describen alguna acción relacionada a registrar, se debe analizar si la información que se está registrando no puede modelarse como objeto. En general, esta heurística no se aplica ya que al tener el LEL aquellos términos que son relevantes para el Macrosistema, existirá un término por cada elemento significativo que haya en el UdeD. Sin embargo es útil considerarla para ser usada como validación del LEL.

Ejemplo: en la introducción se mencionó un ejemplo extraído de Meyer'97
Se debe guardar un registro cada vez que un elevador se mueve de un piso a otro.
Se debe guardar un registro cada vez que se produce un movimiento de un elevador de un piso a otro.
La palabra mover aparece como una frase verbal en una sentencia y como sustantivo en otra. Si bien en la primera parecería registrar como una posible clase, claramente es el registro de un movimiento el que interesa guardar, y esta abstracción es más correcta que la anterior.

H12- Analizar si la clase representa a un individuo o una colección

Definición: Tanto para las clases primarias que modelan Sujetos como para secundarias que modelan Objetos del LEL, debe analizarse si en los escenarios aparece una instancia o un conjunto de instancias. Si aparece un conjunto se modelará como una colección, sin crear otra clase, solo se indica con el plural. Se considera que es una colección con operaciones básicas, por lo que en las siguientes etapas, puede modelarse con clases predefinidas que representan a colecciones. Si tuviera otro comportamiento / significado en el Macrosistema, tendría que haber aparecido como un término independiente del LEL.

Ejemplo: si se analiza el escenario Evaluar licitación se ve que como recurso se utiliza un conjunto de Sobres de Licitación , que son los que serán evaluados. Al no tener una semántica diferente, más allá que la propia de una colección, se utiliza el plural para diferenciarla de la clase que representa a un miembro de la colección pero no se crea una nueva clase. En el caso de los adherentes, existe el término del LEL Grupo que si bien se define como un conjunto de adherentes, tiene su propia semántica dentro del Macrosistema; por esta razón es modelado como clase.

H13- Utilizar el modelo de reglas funcionales para completar las clases

Esta heurística no es tenida en cuenta por no involucrar reglas funcionales en el proceso que se propone.

4.2.5 Documentación de las clases

H14- Documentar las clases con tarjetas estilo CRCs

Descripción: Las *CRCs* inicialmente propuestas en [Beck 89] son tarjetas que describen las clases, sus responsabilidades y colaboraciones. En la propuesta de Lonardi fueron modificadas, pero para este trabajo se utiliza el formato original como estándar y no como heurística.

4.2.6 Evaluación de las CRCs

El objetivo de esta etapa es determinar junto con el cliente si el conjunto de *CRCs* obtenidas modelan, desde una perspectiva de orientación a objetos, el comportamiento del Macrosistema. Además se buscan posibles errores u omisiones en las *CRCs* con respecto a los modelos de reglas, escenarios y LEL. Esto último puede causar la modificación de los modelos de la *Requirements Baseline*.

H15- “Ejecutar” los escenarios sobre la base de las CRC s

Descripción: Si la heurística H5 es aplicada, H15 sería redundante, salvo que las validaciones las realice un equipo diferente al que formuló las *CRCs*. Las *CRCs* pueden ser evaluadas siguiendo los pasos propuestos en [Cockburn 99], es decir tomar cada escenario y “avanzar “ por él determinando si las clases y sus responsabilidades llevan a cabo las acciones de los episodios. Dentro de esta heurística, se puede analizar los patrones de escenarios [Ridao 00]. Según sea el tipo de patrón al que corresponde cada escenario en donde aparecen las *CRCs* se puede validar si existen colaboraciones con las otras *CRCs* para llevar a cabo los tipos de episodios pedidos por cada tipo de patrón. Por el momento esta actividad sólo sirve para ver si se necesita una colaboración y cuantas *CRCs* como mínimo debe intervenir (a partir de la cantidad mínima de participantes que exige cada tipo de patrón). Por otro lado, es importante tener en cuenta que si una determinada responsabilidad involucra clases servidoras, éstas deben tener responsabilidades que puedan llevar a cabo la tarea mencionada. Con respecto a las clases clientes, tiene que aparecer alguna responsabilidad en la *CRC* correspondiente que tenga a la clase como servidora. Este análisis completa la estrategia anterior, ya que en un escenario puede figurar implícitamente las responsabilidades de las clases. Por ejemplo, el cliente reserva un libro. Esta oración esta bien escrita y es válida dentro de un episodio. Desde el punto de vista de objetos, tanto el cliente como el libro deben tener una responsabilidad que describa el comportamiento que le corresponde de la reserva.

H16- Aplicar principios de diseño orientado a objetos

Descripción: Se deben tener en cuenta distintos aspectos de calidad en el modelado con objetos [Cockburn 99; Meyer 97] y métricas de orientación a objetos, por ejemplo [Lorenz 94]. En esta etapa de la estrategia se utilizan aquellas métricas que se refieren sólo a aspectos de cantidad de responsabilidades y colaboraciones de una clase, como por ejemplo: ¿ tiene demasiadas responsabilidades una componente? ¿Tiene pocas responsabilidades? ¿Qué tipo y grado de cohesión existen entre las clases? (En esta

etapa sólo se puede analizar la cohesión de una clase por la relación de sus responsabilidades, ya que la parte interna de la clase aun no ha sido definida). Esta clase de preguntas mejora la calidad de las CRCs. En etapas posteriores se deben aplicar de una forma más sistemática alguno de los sistemas de métricas para objetos propuestos como por ejemplo [Chidamber 94; Lorenz 94], así como también principios generales de diseño OO por ejemplo [Martin 00; Meyer 97].

H17- Validar el modelo de CRC s con los stakeholders

Descripción: Esta heurística ayuda en la validación con el stakeholder, analizando si el modelo conceptual de objetos corresponde a las expectativas y necesidades del mismo, de una forma similar a la propuesta en *XP*. El ingeniero de Software conduce una entrevista con los clientes. Los modelos de LEL, reglas y los escenarios, junto con las relaciones de trace que les permite relacionarse con el modelo de objetos, son usados para chequear si los términos y comportamiento claves del Microsistema fueron bien interpretados en las CRCs. Por ejemplo: ¿Cómo esta considerada la regla XX en el modelo de objetos? ¿Cómo se modela un determinado escenario? ¿Cómo se “implementa” este concepto del LEL en el modelo? entre otras preguntas. Si hay un cambio en alguna CRC producto de una mala definición de los escenarios o LEL, éste debe ser reflejado en los modelos correspondientes. Las relaciones backward de traceability de las CRCs ayudan a determinar los modelos involucrados ya que permiten obtener las componentes de los modelos de LEL, escenarios y Reglas involucradas en la creación de cada CRC, sus responsabilidades y colaboraciones.

Partiendo de la información recopilada en este capítulo, se utiliza para armar la actividad 4.00 del proceso propuesto en el capítulo 6.

Framework del Proceso de Construcción

Objetivos del Capítulo

El objetivo de este capítulo es definir el framework que se utilizará para definir el proceso LEC.

Contenido

- 5.1-Introducción a procesos de software.
- 5.2-Definición de framework y template para el proceso LEC
- 5.3-Template para el proceso LEC

5.1- Introducción a procesos de software

5.1.1 Definición y contenido de un proceso.

Como primer punto analizaremos que se entiende por proceso citando definiciones de distintos autores.

Proceso es: “Conjunto de pasos que involucran actividades, restricciones y recursos que producen un output de algún tipo. Involucra: Técnicas, Herramientas, Procedimientos”

Técnica: conjunto de pasos que se deben realizar para conseguir un resultado.

Herramienta: sistema automatizado que nos permite lograr algo de una mejor manera.

Método/Procedimiento: “combinación de herramientas y técnicas que en conjunto producen un producto particular” [Basili]

Otra definición de proceso es la que propone la IEEE en el STD 610.

Proceso: “Secuencia de pasos ejecutadas con un propósito dado” [IEEE-STD-610].

La definición de proceso de Olson dice: “Un proceso es una serie de eventos o fases que tienen lugar con el tiempo y tienen un propósito o resultado identificado. Un proceso tiene entry criteria, inputs, actividades, exit criteria, outputs, roles, etc.” [Olson 93]

De las tres definiciones propuestas de proceso, se puede observar que aunque tienen distinto grado de detalle en lo que proponen que debe tener un proceso, coinciden en que son pasos a desarrollar para obtener un resultado determinado.

En el caso de la definición de la IEEE es muy abstracta, las definiciones de Basili y de Olson son las más detalladas, donde además de definir que un proceso es una secuencia de pasos nos dice que debe tener. Estas dos definiciones si tomamos textualmente lo que dicen podemos decir que tienen un aspecto en común, que son las actividades. Para esta tesis la definición adoptada es la propuesta por Olson, por ser la más amplia y más puntual en sus “características”.

5.1.2 Clasificación de procesos.

En este punto se propone una clasificación de procesos propuesta por Sommerville. Esta clasificación la propone en un contexto de mejora de procesos, donde habla de proceso que se han implementado y los clasifica en que estado se encuentran.[Sommerville 02].

Los diferentes tipos de procesos planteados son los siguientes.

Procesos informales:

El proceso no está definido de forma estricta y es elegido por el equipo de desarrollo.

Procesos administrados:

El proceso está definido, define procedimientos, duración y relaciones entre los procedimientos.

Procesos metodológicos:

Se utilizan algunos métodos o modelos de desarrollo definidos.

Procesos de mejora:

Son procesos que tienen objetivos de mejora inherentes. Cuentan con mediciones cuantitativas del proceso.

El framework propuesto en esta tesis en la medida que se implemente con las características que se proponen, permite contar con un proceso administrado.

5.1.3- Frameworks & Templates

En principio describiremos el template propuesto por Basili para definir un modelo de proceso. “Un modelo de ciclo de vida define el orden de aplicación de varios métodos y documentos anexos, pero no define las técnicas y métodos necesarios para ser aplicados”.

En este contexto es que Basili propone considerar las siguientes características para definir un proceso [Basili]:

Input: define los objetos que se necesitan para trabajar y el estado del ambiente para que se pueda aplicar.

Output: los objetos producidos por el método y los resultados o cambios que produce en el ambiente la aplicación del método.

Técnica: son las técnicas básicas para transformar las entradas en salidas

Formalidad: es el grado de rigor con que deben ser aplicadas las técnicas

Énfasis: enfoque que se debe dar a los métodos y técnicas

Definición del método: el modelo de proceso asociado a la aplicación de las técnicas, incluyendo cosas tales como dirección y estructura organizacional.

Perspectiva: es el enfoque con que el método y las técnicas son aplicados.

Calidad del producto: es el grupo de características de calidad requeridas para el producto.

Calidad del proceso: es el grupo de características de calidad requeridas para el proceso.

Por ejemplo, el aplicar el template a una especificación de testing, resulta [Basili]

Input: sistema, subsistema, módulos.

Output: resultado del testeo.

Técnica: estructura, funcional.

Formalidad: adhesión total o parcial a las reglas definidas para las técnicas de testing.

Énfasis: detección de fallas, rendimiento, integridad.

Definición del método: como esta especificado en el plan de testing.

Perspectiva: varias clases de clientes / configuraciones de hardware.

Calidad del producto: integridad, eficiencia.

Calidad del proceso: adhesión al método, integración de procesos.

Otro enfoque es el Software Process Framework (SPF) [Olson 93] que es el utilizado como base para el framework propuesto en esta tesis.

El SPF tiene dos usos principales:

El primero es una herramienta para revisar y analizar un proceso de software para asegurarse que es consistente con CMM.

El segundo ayuda a diseñar nuevos procesos de software que sean consistentes con CMM.

El SPF reorganiza las key practices de CMM para facilitar el diseño, análisis y revisión de procesos. Las key practices son reagrupadas de acuerdo con el “Operational Framework”, que se describe en el siguiente punto.

El contenido del SPF es tomado directamente de CMM, y no se incorporan nuevas key practices o requerimientos de procesos. Reorganiza la información recomendada por CMM en un formato que es útil para definir tareas del proceso de software.

SPF identifica políticas, estándares, procesos, procedimientos, capacitación, y herramientas recomendadas por CMM, y provee un checklist para asegurarse que los procesos documentados son consistentes con CMM.

Operational Framework

El SPF, esta basado en un *Operational Framework* que se describe a continuación. A fin de mejorar las operaciones, una organización debe definir documentos y comunicar sus políticas, estándares, procesos, procedimientos, capacitación y herramientas. De manera conjunta, estos elementos y sus relaciones pueden ser referenciados como “*Operational Framework*” ellos guían el desarrollo de productos.

El *Operational Framework* lo constituyen seis componentes:

Políticas: proporciona las pautas que guían y restringen las operaciones.

Estándar: provee las “definiciones operativas” o “Criterios de aceptación” para productos finales o intermedios, o procesos.

Procesos: “..serie de eventos o fases que tienen lugar con el tiempo y tienen un propósito o resultado identificado” describen como construir productos siguiendo el estándar y las políticas.

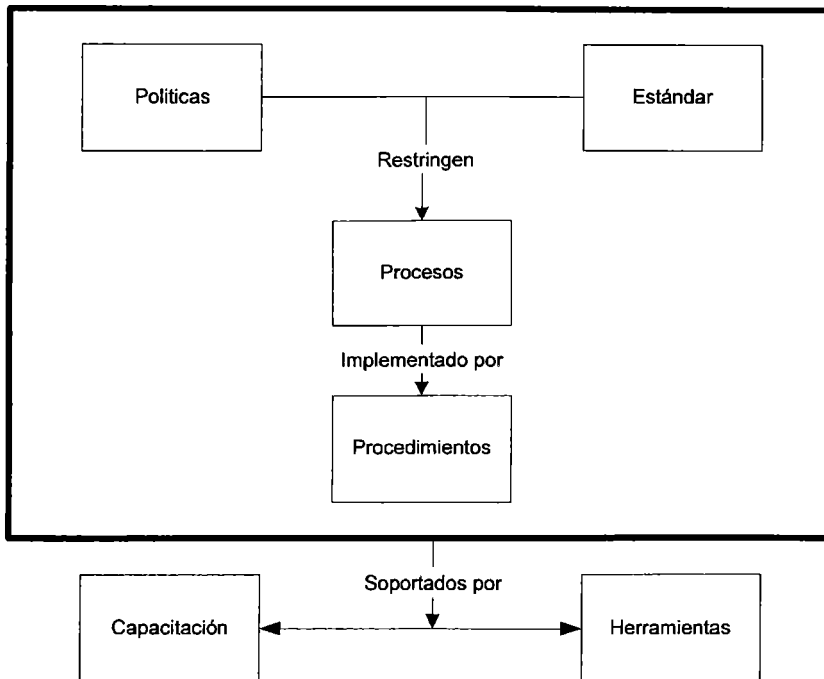
Procedimientos: describe “como” o “paso a paso” implementar los procesos.

Capacitación: brinda el conocimiento y habilidades necesarias al equipo de desarrollo.

Herramientas: brinda el apoyo necesario para construir los productos de software.

Estas componentes se relacionan de la siguiente forma. Las *políticas* guían y restringen los *procesos*. Por ejemplo los requerimientos deben ser documentados. Los *estándar* guían y restringen los *procesos*, con respecto a sus *output*. Por ejemplo, todas las especificaciones de requerimientos deben respetar el estándar IEEE / ANSI 830 1984. Los *procesos* son implementados por los *procedimientos*. Por ejemplo explicar de que manera completar la especificación de requerimientos para cumplir con el estándar. La *capacitación* brinda el conocimiento y habilidades necesarias para poder implementar las *políticas, estándares, procesos y procedimientos*. Por ejemplo realizar un curso de capacitación sobre el estándar de la IEEE 830. *Herramientas* brindan soporte para desarrollar el *proceso*. Por ejemplo utilizar una herramienta CASE que de soporte al estándar.

En la Figura 5.1 resume y muestra las relaciones entre los componentes del *Operational Framework*.



Relación de los componentes de un “Operational Framework”
Figura 5.1

Los beneficios que se obtienen de aplicar un “*Operational Framework*” [Olson 93] son:

- **“Separa la información usada para diferentes propósitos:”**
Desagregar la información teniendo en cuenta cual es su finalidad. Por ejemplo, si algún integrante de la organización quiere utilizar un documento de capacitación se debe contar con el material específico y no que se encuentre como parte de un documento mucho más extenso que no este preparado con ese fin.
- **“Identifica y usa solamente la información relevante para cada parte:”**
Solo debe tener la información relevante para esa parte del *Operational Framework*. Por ejemplo, información que se utiliza para capacitación solo debe estar en los documentos de capacitación.
- **“Administra las partes operacionales para que trabajen todas juntas como un sistema.”**
Cuando la información fue identificada y particionada , puede ser diseñada como un sistema. Las relaciones entre los componentes del “*Operational Framework*” pueden ser optimizadas dado que trabajan en conjunto para transmitir como se deben realizar las actividades.

- **“Administra cambios y mejoras:”**
Es más fácil hacer cambios dado que la información esta separada en partes. Por ejemplo, una vez definidas las políticas no cambian frecuentemente, en cambio los documentos de capacitación cambian pero al estar separados en partes es más sencillo realizar las modificaciones.
- **“Administra y mejora la comunicación:”**
Permite conocer donde obtener la información que se necesita y como se relaciona con el resto, como los cambios se aíslan mas a las partes operativas, se necesita menos comunicación y solo los cambios relevantes necesitan ser administrados y comunicados.

En el SPF se utilizan una serie de conceptos con el significado que les asigna el SEI:

Actividad: “Accion tomada para crear o alcanzar un work product, servicio o resultado.”

Entry Criteria: “Describe las condiciones bajo las cuales una actividad puede comenzar. Entry criteria toma la forma de un predicado simple o compuesto sobre el estado de un work product, agente, o actividad.”

Exit Criteria: “Describe las condiciones bajo las cuales una actividad puede ser declarada completa. Exit criteria toma la forma de un predicado simple o compuesto sobre el estado de un work product, agente, o actividad”

Input: “La relación o vínculo entre una actividad y un work product. Inputs son los resultados producidos por una actividad anterior y usados por la actividad actual, y pueden ser calificados por el estado de los work products”

Output: “La relación o vínculo entre una actividad y un work product. Outpus son los resultados producidos por la actividad actual y usados por la actividad siguiente, y pueden ser calificados por el estado de los work products”

Rol: “El realizador o actor que lleva a cabo las acciones para alcanzar o crear un work product, servicio, o resultado. Un rol puede consistir en usar computadoras o ser totalmente automatizado”

Work product: “Cualquier producto final o intermedio, servicio, o resultado de un proceso o actividad”

Estos conceptos del SEI son adoptados para el desarrollo de esta tesis.

El Software Process Framework (SPF) que es el *Operational Framework* propuesto por Olson, tiene un template con los siguientes elementos: roles, entry criteria, inputs, actividades, outputs, exit criteria, revisiones y auditoria, administración y control de work products, mediciones, procedimientos documentados, capacitación, herramientas.

Roles: enumera los roles que participan en las actividades del proceso.

Criterio de entrada (Entry criteria):determina cuando el proceso puede comenzarse.

Inputs: menciona los work products necesarios para desarrollar el proceso.

Actividades: accion tomada para crear o alcanzar un work product, servicio o resultado.

Outputs: describe los work products que se obtienen luego de desarrollar el proceso.

Criterio de salida (Exit criteria): determina cuando un proceso se puede dar por terminado.

Revisiones y auditoria: enumera las revisiones y auditorias necesarias.

Administración y control de work products: enumera los work products que deben ser administrados y controlados.

Mediciones: determina las mediciones requeridas por el proceso.

Procedimientos documentados: enumera las actividades que deben completarse de acuerdo al procedimiento.

Capacitación: enumera la preparación necesaria para poder desarrollar el proceso.

Herramientas: enumera las herramientas necesarias para desarrollar el proceso.

El template de Olson se toma como base para este trabajo por considerar que es más amplio y detallado que el de Basili. Si bien en el template de Basili algunas características como *criterio de entrada y salida*, de gran importancia para definir un proceso, están incluidas dentro de la definición de *definición del método*, otra característica como *capacitación* para determinar el conocimiento necesario para quienes tienen que desarrollar cada uno de los roles no las tiene en cuenta.

5.2-Definición de framework y template para el proceso LEC (LEL, Escenarios y CRC).

Esto se realiza proponiendo un template que se orienta a ciertas características del proceso.

Estas características se corresponden parcial o totalmente con las componentes del Operational Process Framework de Olson.

La componente *Política*, es la que menos desarrollada está por ser la que depende más de la implementación particular que se haga del framework propuesto en este trabajo.

Teniendo en cuenta las características planteadas en el punto 5.1, se seleccionaron las siguientes características para el template con el cual se describe el proceso propuesto:

- Código
- Nombre
- Objetivos
- Justificación
- Actividades
- Recursos
- Inputs
- Outputs
- Estándares
- Restricciones
- Variables a Medir
- Observaciones

Estas características fueron elegidas pensando en su implementación y en poder utilizarlo como base para la capacitación.

A continuación se explica el significado de cada uno de los componentes seleccionados que forman parte del template propuesto.

Código:

Es el número con que se identificara el proceso y da secuencia. Formato #.##

Nombre :

Nombre que identifica el proceso.

Objetivos:

Son las metas que nos proponemos alcanzar al desarrollar las actividades del proceso.

El contar con esta información nos ayuda a realizar el proceso conociendo cuales son los resultado esperados, y de esta forma desarrollar las actividades teniendo en cuenta los metas fijadas.

Por ejemplo: Representar el vocabulario del dominio de la aplicación.

Justificación:

Razones por las cuales se tiene que realizar el proceso.

Actividades:

Este componente se utiliza con el mismo criterio que en el témplate propuesto por Olson, pero ajustándolo al proceso que se describe en esta tesis.

Enumeran en orden las acciones tomadas, por Ingeniero de requerimientos y/o el usuario, para obtener un *work product*.

Para esta característica se registra: Paso, Nombre Actividad, Objetivo, y Rol responsable de la ejecución.

Paso: es el orden en que se debe ejecutar la actividad.

Nombre Actividad: la identifica dentro del proceso.

Objetivo: que es lo que nos proponemos obtener de realizar esta actividad.

Rol responsable de la ejecución: quien es el encargado de desarrollar la actividad.

Dentro de esta característica se determina que rol es el responsable de realizar la actividad.

Por ejemplo: Paso: 1, Nombre Actividad: Describir los símbolos, Objetivo:

Completar la noción e impacto según la clasificación, Rol responsable de la ejecución: analista de sistemas.

Recursos:

Este componente agrupa a roles, herramientas y método. El motivo por el cual se proponen agrupados es por hacer más sencillo la lectura de los documentos y cuando se analizan los recursos necesarios para realizar el proceso se encuentran todos en la misma sección.

En el témplate propuesto estos recursos se desagregan en tres tipos: humanos, tecnológicos y métodos.

A continuación se explican cada uno en detalle.

Humanos

Para los recursos humanos se registra: Rol y Capacitación requerida.

Rol: es el responsable de realizar la actividad.

Capacitación requerida: Conocimientos y habilidades necesarias para el rol.

En el caso de Olson la capacitación es propuesta como otra componente del *Operational Framework*, para la cual utiliza otro template para definirlo.

En este caso se propone incluirlo en el mismo template para determinar que recursos humanos y capacitación deben tener para poder realizar las actividades. Esto permite a quien este asignando ó seleccionando los recursos humanos para el proceso conocer el rol conjuntamente con la capacitación requerida.

En la figura 5.2 se proponen los participantes a modo de ejemplo, los distintos tipos de participantes dependen de la envergadura de cada proyecto en particular y los recursos disponibles.

Equipo de Desarrollo	Usuario / Cliente
Líder de proyecto	Estratégico (Dirección)
Analista de Sistemas	Gerencial (Gerencias o Jefaturas)
	Operativo (Operación)

Figura 5.2
Recursos Humanos del Proyecto

Por ejemplo el Analista de sistemas debe estar capacitado para realizar entrevistas.

Tecnológicos

Herramientas, con las que se debería contar para dar soporte a las actividades del proceso.

Este recurso Olson también lo toma como un componente del *Operational Framework* y lo implementa en otro template.

No es un recurso indispensable, pero contar con herramientas adecuadas ayuda a obtener una mayor estandarización en el proceso y a mejorar la comunicación, sobre todo cuando trabajamos con equipos de desarrollo numerosos.

Por ejemplo BMW [Antonelli 99]

Métodos

Son técnicas y herramientas que al aplicarse de manera conjunta generan un producto determinado.

Por ejemplo: Heurísticas para determinar los símbolos a representar.

Forma de representación de los símbolos.

Input:

Son información, conocimiento o *Work Products* que se necesitan tener disponibles para poder comenzar con el proceso. Sin contar con estos, no se puede desarrollar el proceso.

Esta característica incluye el criterio de entrada que debe satisfacer el input.

Para los Inputs se registra: Nombre, Origen, Criterio de entrada, Tipo.

Nombre : Nombre con que se identifica a los Input requeridos.

Origen : Que actividad genera ese *Work Product*.

Criterio de entrada: Es el estado o condición que debe tener el input

Tipo: Si es Información, Conocimiento o *Work Product*

Output:

Son información, conocimiento o *Work Products* con los que se debe contar al final del proceso. El no haber obtenido estos resultados hace que no se pueda dar por finalizado el proceso.

Esta característica incluye el criterio de salida que debe satisfacer el output.

Para los Outputs se registra: Nombre, Destino, Criterio de salida, Tipo.

Nombre : Nombre con que se identifican a los output generados por la actividad.

Destino : Que actividad se alimenta con ese output.

Criterio de salida: Es el estado o condición que debe tener el output.

Tipo: Si es Información, Conocimiento o *Work Product*.

Estándares:

Utilizados para los productos y procesos.

Esta característica Olson la trata como otra componente del *Operational Framework* que implementa en otro template, pero los *estándares* están íntimamente relacionados con las actividades que se desarrollan en el proceso, por este motivo considero que es mucho más práctico para implementar el contar con esta información en el mismo template que el tener que consultar otro documento para conocerlos.

Restricciones:

Condiciones sin las que el proceso no se puede ejecutar.

Se aclaran las limitaciones a los métodos para su implementación.

Variables a Medir

Variables que, por medio de sus valores, nos permiten analizar el resultado de la aplicación del proceso.

De esta forma podemos analizar cómo es el comportamiento del proceso de un proyecto a otro, y poder detectar cambios que deben realizarse para obtener mejoras.

En este punto se sugieren algunas de las variables a medir, esta lista puede ser ampliada según se requiera.

Para las variables a medir se registra su nombre.

Observaciones

En esta característica se indicaran los comentarios o aclaraciones que se consideren necesarias para el proceso

Ahora que tenemos definido el framework propuesto para definir un proceso, podemos determinar si fueron definidas todas las características necesarias para que el proceso pueda ser implementado.

Olson en su trabajo propone una lista de preguntas mínimas que deben ser respondidas a fin de determinar si un proceso fue definido con los elementos necesarios y si es capaz de ser implementado. Este grupo de preguntas las llama “Key Process Questions” (KPQ).

Las respuestas a estas preguntas son “elementos de proceso” que deben estar documentados.

Las KPQ que propone son las siguientes:

- ¿Qué artefactos son producidos?
- ¿Qué actividades son desarrolladas?
- ¿Qué agentes están involucrados?
- ¿Cómo son implementadas las actividades?
- ¿Cuándo las actividades comienzan y terminan?
- ¿Qué artefactos son producidos (ó utilizados) por que actividad?
- ¿Qué agente es responsable de desarrollar cada actividad?
- ¿Qué actividad es la próxima?

En este trabajo estas preguntas se utilizan para verificar el template propuesto, si responde a estas y determinar si es una definición de proceso capaz de ser implementado.

En el cuadro, figura 5.3, se ve que característica del template propuesto responde a la KPQ.

Key Process Questions	Característica que la responde
¿Qué artefactos son producidos?	Output
¿Qué actividades son desarrolladas?	Actividades
¿Qué agentes están involucrados?	Recursos: Recursos humanos
¿Cómo son implementadas las actividades?	Actividades / Métodos
¿Cuándo las actividades comienzan y terminan?	Input / Output estas dos características incluyen criterio de entrada y salida
¿Qué artefactos son producidos (ó utilizados) por que actividad?	Input / Output incluyen origen y destino respectivamente de los work product
¿Qué agente es responsable de desarrollar cada actividad?	Actividades: Rol responsable de la ejecución.
¿Qué actividad es la próxima?	Actividades: Orden

Figura 5.3

El analizar las respuestas del cuadro, nos permite decir que utilizar este framework para describir el proceso permite que sea implementado.

La demostración real de que este proceso puede ser implementado se hace en el capítulo 7 de esta tesis, aplicándolo a un caso real.

5.3- Template del proceso

<u>Código:</u>		<u>Nombre:</u>	
<u>Objetivos:</u>			
<u>Justificación:</u>			
<u>Actividades:</u>			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
<u>Recursos:</u>			
<u>Humanos:</u>			
<u>Rol:</u>		<u>Capacitación requerida:</u>	
<u>Tecnológicos:</u>			
<u>Método:</u>			
<u>Input:</u>			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
<u>Output:</u>			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
<u>Estándares:</u>			
<u>Restricciones:</u>			
<u>Variables a medir:</u>			
Nombre de la variable			
<u>Observaciones</u>			

Capítulo 6

Definición del proceso LEC

Objetivos del Capítulo

El objetivo de este capítulo es describir como se construye con el framework propuesto en el capítulo anterior el proceso LEC.

Contenido

- 6.1-Introducción
- 6.2-Proceso LEC
- 6.3-Determinación de fuentes de información.
- 6.4-Construcción del LEL
- 6.5-Construcción de Escenarios
- 6.6-Construcción de CRC

6.1- Introducción.

Teniendo en cuenta la información que se describe en los capítulos 1 a 4 por cada uno de los autores, y el framework propuesto en el capítulo 5, se instanciará el framework para el proceso de construcción de LEC (LEL, Escenarios y CRC)

6.2- Proceso LEC

El proceso LEC tiene como objetivo instanciar el framework propuesto con la información existente sobre LEL, Escenarios y CRC.

Contar con un proceso documentado de este estilo nos brinda los siguientes beneficios.

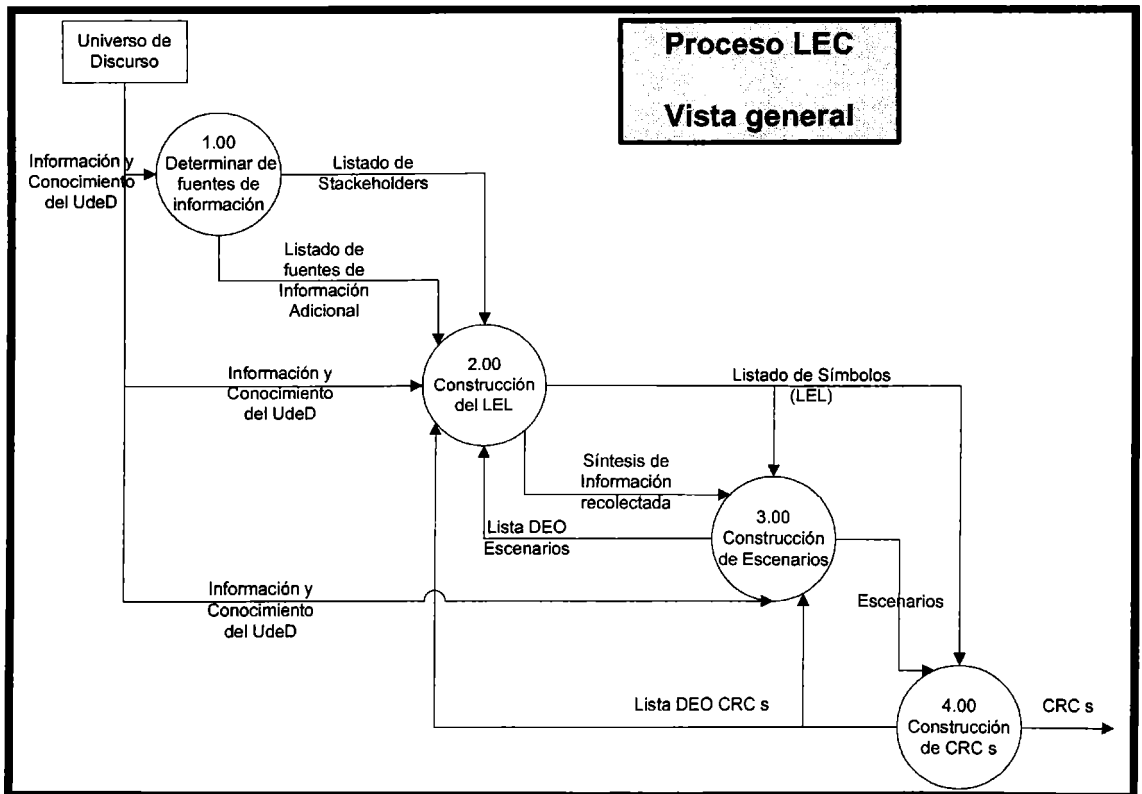
- Ayuda a una comprensión común de todo el equipo de desarrollo
Tener el proceso documentado de esta manera nos da una fuente de información única a la cual consulta el equipo de desarrollo para conocer, por ejemplo, cual es la próxima actividad, cuales son los objetivos que se quieren alcanzar.
- Facilita encontrar inconsistencias, redundancias y omisiones.
Tener el proceso documentado hace que sea mucho más sencillo, seguirlo y mejorarlo con las lecciones aprendidas de un proyecto a otro.
- Permite la adecuación a cada situación en particular.
Brinda la posibilidad, según cada caso en particular, de ajustar el proceso para poder ser aplicado. Por ejemplo: si se trata de implementar en una organización que no tiene una estructura de recursos humanos con el número necesario para el proceso, puede ajustarse respetando los roles que se deben asumir concentrándolos en menos personas.
- Herramienta de capacitación.
En este documento contamos con toda la información necesaria para poder armar el plan de capacitación para quienes no conozcan el proceso.

Este proceso consta de las siguientes actividades:

- 1. Determinación de fuentes de información**
- 2. Construcción del LEL**
- 3. Construcción de Escenarios**
- 4. Construcción de CRC s**

En la primera actividad, **Determinación de fuentes de información**, partiendo del universo de discurso, se determina cuales son las fuentes de información en las que nos vamos a basar para construir. Este resultado alimenta a la siguiente actividad, **Construcción del LEL**, en la cual se realiza la recolección y análisis de la información teniendo en cuenta las fuentes seleccionadas en la actividad anterior. Como resultado de esta segunda actividad se obtiene el LEL del dominio de la aplicación a partir del cual se construirán los escenarios, en la actividad **Construcción de Escenarios**. Una vez construidos los escenarios y con el LEL obtenido en la actividad **Construcción del LEL**, se construyen las CRC s.

El DFD de la figura 6.1 muestra la relación que existe entre las distintas actividades planteadas para el proceso el flujo de datos entre un actividad y otra.



Proceso LEC – Vista General
 Figura 6.1

6.3-Determinar fuentes de información.

Esta actividad es tratada como un proceso en sí mismo, debido a que es uno de los factores críticos de éxito de la implementación de la Ingeniería de requerimientos. El no involucrar los stakeholders adecuados y demás fuentes de información puede llevarnos a obtener productos no deseados por el cliente.

El tenerla como una actividad de la misma jerarquía que, por ejemplo, construcción del LEL hace que tome una mayor importancia al desarrollarla y no que sea tomada como una “sub-actividad” dentro de otra de mayor jerarquía.

Código:		Nombre:	
1.00		Determinar fuentes de información	
Objetivos:			
Determinar las fuentes de información en las cuales nos vamos a basar para elicitar.			
Justificación:			
El no detectar los stakeholders puede llevarnos a una falta de completitud y no poder validar la consistencia del conocimiento que reflejamos.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Seleccionar stakeholders del dominio a modelar.	Determinar dentro de la organización que estamos analizando quienes son los potenciales usuarios del sistema y seleccionar los que participaran del proceso de elicitación.	Líder de proyecto
2	Crear lista de stakeholders	Crear un documento que establezca quienes serán los stakeholders seleccionados	Analista de sistemas
3	Seleccionar otras fuentes de información para elicitar.	Determinar que otro medio para adquirir conocimiento sobre el dominio utilizaremos	Líder de proyecto
4	Crear lista de otras fuentes de información.	Crear un documento que establezca cuales serán las fuentes de información adicionales y la justificación de la elección	Analista de sistemas
5	Validar con el usuario la lista de stakeholders y de fuentes de información.	Obtener la aprobación del usuario con respecto a quienes serán nuestro principales referentes dentro de la organización y fuentes de información adicional	Líder de proyecto

Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Técnicas de elicitación Alto nivel de abstracción Negociador	
Analista de sistemas		Técnicas de elicitación Alto nivel de abstracción	
Usuario Estratégico / Gerencial		Visión global de empresa Expectativas del sistema a modelar	
Tecnológicos:			
Procesador de textos.			
Método:			
Entrevistas Análisis de documentación y de contenidos en general			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Información del dominio a modelar	Universo de Discurso	Información global de la empresa y del sistema a modelar	Información
Output:			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
Listado de stakeholders	Recolectar y analizar información	Listado con función y responsabilidad relacionado con el sistema a modelar validado por el usuario.	Work product
Listado de fuentes de información adicional	Recolectar y analizar información	Nombre de la fuente y que información debe aportar validado por el usuario.	Work product
Estándares:			
Restricciones:			
Las provenientes del dominio y de las formas que asume su información y stakeholders.			

Variables a medir:	
Nombre de la variable	
Total de usuarios detectados por jerarquía / área / función	
Usuarios seleccionados por jerarquía / área / función	
Observaciones	
<p>El involucrar el nivel estratégico y gerencial en el proceso hace que se obtenga el compromiso político que se requiere para poder implementar un proceso de manera exitosa.</p> <p>El que debe ocuparse de crear y mantener esta relación es el Líder de proyectos.</p> <p>Las entrevistas y análisis de documentos de esta actividad apuntan a conocer la estructura y roles dentro de la organización, principalmente de los sectores involucrados en el sistema a modelar.</p>	

6.4- Construcción del LEL

Es la segunda actividad del proceso, por la complejidad que tiene, se desagrega en varias actividades.

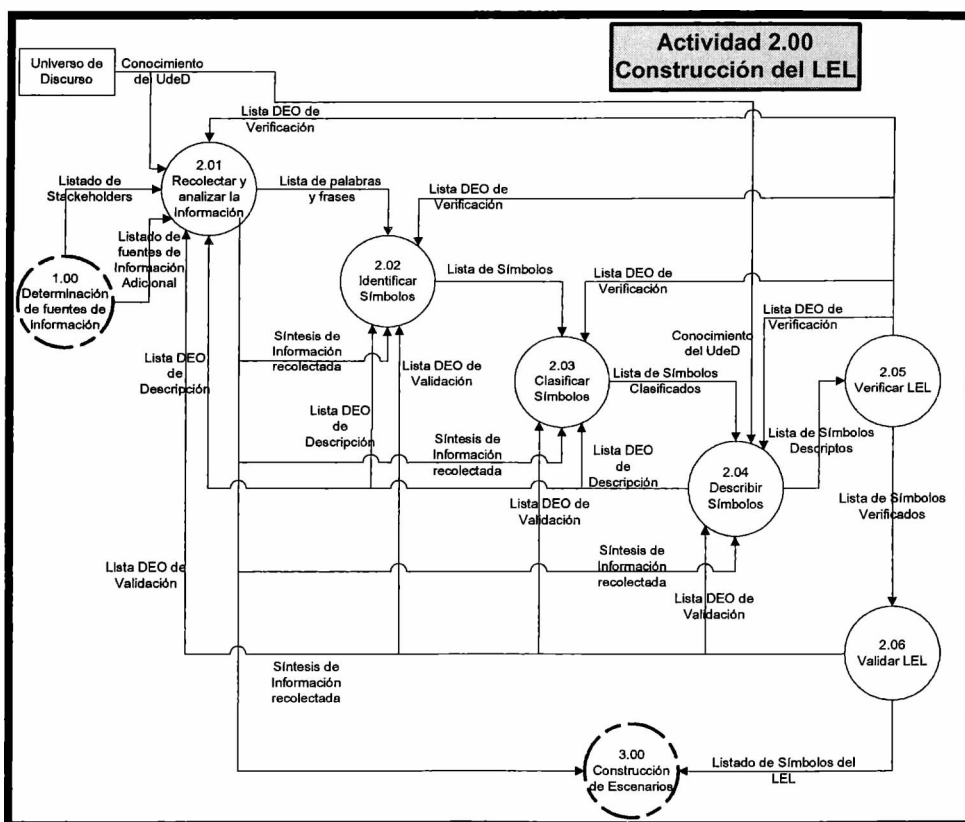
Las actividades que involucra son las siguientes:

- Recolectar y analizar la información
- Identificar símbolos
- Clasificar símbolos
- Describir símbolos
- Verificar LEL
- Validar LEL

En principio se describe la actividad Construcción del LEL, con un alto grado de abstracción para que permita tener una visión global de lo que involucra, es por eso que dentro de esta actividad, la característica actividades menciona cada una de las actividades que involucra pero sin especificar el rol responsable.

Luego se describe de forma detallada cada una de la actividades involucradas.

En el DFD de la figura 6.2 se muestra la relación entre las actividades que involucra la creación del LEL.



Proceso LEC – Construcción del LEL

Figura 6.2

<u>Código:</u>		<u>Nombre:</u>	
2.00		Construcción del LEL	
<u>Objetivos:</u>			
Representar el vocabulario del Dominio de la aplicación .			
<u>Justificación:</u>			
Una de las dificultades mayores que tenemos en la comunicación con el cliente / usuario es utilizar el lenguaje natural por ser muy ambiguo. El definir el LEL nos permite determinar claramente el significado de cada término dentro del universo de discurso que estamos modelando, con lo cual nos permite mejorar la comunicación con el cliente / usuario.			
<u>Actividades:</u>			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Recolectar y analizar la información	“Aprender” sobre el dominio que estamos analizando.	
2	Identificar símbolos	Determinar cuales son los símbolos que representan el lenguaje del dominio en análisis.	
3	Clasificar símbolos	Agrupar los símbolos según la clasificación seleccionada.	
4	Describir símbolos	Describir los símbolos con su noción e impacto teniendo en cuenta la clasificación.	
5	Verificar LEL	Asegurar que las pautas establecidas para la construcción del LEL se desarrollaron según lo planificado.	
6	Validar LEL	Asegurar que los símbolos modelados en el LEL son consistentes con lo esperado por el usuario y el equipo de desarrollo.	

Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Técnicas de elicitación Alto nivel de abstracción Conocimientos del dominio de la aplicación Negociador Técnicas de construcción de LEL & Escenarios	
Analista de sistemas		Técnicas de elicitación Alto nivel de abstracción Conocimientos del dominio de la aplicación Negociador Técnicas de construcción de LEL & Escenarios	
Usuario Estratégico / Gerencial		Visión global de la empresa Expectativas del sistema a modelar	
Usuario Operativo		Conocimiento específico de la aplicación	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			
Método:			
<ul style="list-style-type: none"> • Heurística para determinar los símbolos a representar. • Criterios de clasificación de símbolos. • Forma de representación. • Heurística de validación. • Heurística de verificación. 			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condíc. Y/O	Tipo
Listado de stakeholders	Recolección y análisis de información	Listado con función y responsabilidad relacionado con el sistema a modelar validado por el usuario.	Work product
Listado de fuentes de información adicional	Recolección y análisis de información	Nombre de la fuente y que información debe aportar.	Work product
Conocimiento del UdeD	Universo de Discurso	Conocimiento del sistema a modelar	Conocimiento

Output:			
Nombre	Destino	Criterio de Salida : Estado o Condición Y/O	Tipo
Lista de símbolos del LEL	Construcción de Escenarios	LEL validado	Work product
Síntesis de información	Construcción de Escenarios	Resumen del conocimiento recolectado en la actividad de construcción del LEL	Información

Estándares:

LEL: Representación de los símbolos en el lenguaje del dominio de la aplicación

Sintaxis:

$$\{\text{Símbolo}\}_1^N$$

Símbolo: Entrada del léxico que tiene un significado especial en el dominio de la aplicación

Sintaxis:

$$\{\text{Nombre}\}_1^N + \{\text{Noción}\}_1^N + \{\text{Impacto}\}_1^N$$

Nombre: Identificación del símbolo. Más de uno representa sinónimos

Sintaxis:

Palabra | Frase

Noción: Denotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo

Sintaxis:

Sentencia

Impacto: Connotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo

Sintaxis:

Sentencia

Sentencia está compuesto por Símbolos y No-Símbolos pertenecientes al vocabulario mínimo.

+ significa composición, {x} significa cero o más ocurrencias de x, () es usado para agrupamiento, | significa or, y [x] significa que x es opcional

Clasificar símbolos:

Símbolos de tipo Sujeto: son actores principales o secundarios del dominio de la aplicación. Los primeros ejecutan acciones directamente en el dominio de la aplicación, y los actores secundarios reciben y/o brindan información al sistema.

Símbolos de tipo Verbo: son acciones que se desarrollan en el dominio de la aplicación.

Símbolos de tipo Objeto: son los elementos sobre los cuales los cuales se realizan las acciones.

Símbolos de tipo Estado: son utilizados cuando asociado al dominio en análisis el estado que tome alguno de sus valores posibles es importante modelar.

Descripción de símbolos:

Para un símbolo del tipo Sujeto

Noción: define quién es el sujeto

Impacto: deben registrarse las acciones que ejecuta o recibe.

Para un símbolo del tipo Verbo

Noción: debe decir quién ejecuta la acción, cuándo ocurre y las actividades involucradas con dicha acción.

Impacto: debe identificar las situaciones que impiden la ocurrencia de la acción, qué otras acciones son disparadas en el ambiente y qué situaciones son causadas por la acción.

Para un símbolo del tipo Objeto

Noción: debe definir al objeto e identificar otros símbolos del mismo tipo con los cuales se relaciona.

Impacto: describe las acciones que pueden aplicarse a este objeto.

Para un símbolo del tipo Estado

Noción: define su significado y las acciones que llevan a ese estado.

Impacto: debe identificar otros estados y acciones que pueden ocurrir a partir de la situación específica.

Restricciones:

- Estrategias de elicitación seleccionadas en base a las fuentes de información.
- **Principio de circularidad:** en la descripción de la noción o impacto de los símbolos se deben maximizar el uso de otros símbolos del léxico.
- **Principio de vocabulario mínimo:** se debe minimizar el uso de símbolos externos al lenguaje de la aplicación. De este modo, se acota el lenguaje al menor conjunto de símbolos posibles. Si se utilizan símbolos externos, éstos deben pertenecer al vocabulario básico del lenguaje natural [Leite97b]

Variables a medir:

Nombre de la variable	
Usuarios entrevistados	
Entrevistas realizadas por usuario	
Documentos consultados por tipo	
Reuniones grupales	
Símbolos identificados	
Símbolos del LEL por categoría	
Símbolos reclasificados	
Símbolos en la lista definitiva del LEL	
Símbolos corregidos	
Símbolos eliminados	

Símbolos agregados	
Símbolos que deben ser modificados.	
Símbolos con errores	
Observaciones	

Código:		Nombre:	
2.01		Recolectar y analizar la información .	
Objetivos:			
Aprender sobre el dominio que estamos analizando.			
Justificación:			
El desarrollo de esta actividad nos permite aprender sobre el dominio de la aplicación que debemos modelar.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Analizar fuentes de información	Aplicar las técnicas de elicitación seleccionadas para obtener información para preparar entrevistas y conocimiento del dominio	Líder de proyecto /Analista de sistemas
2	Preparar entrevistas	Realizar todas las actividades de planificación de entrevistas	Líder de proyecto /Analista de sistemas
3	Aplicar técnicas de elicitación	Obtener la información necesaria para poder modelar los símbolos	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Técnicas de elicitación seleccionadas Alto nivel de abstracción Técnicas de construcción de LEL & Escenarios	
Analista de sistemas		Técnicas de elicitación seleccionadas Alto nivel de abstracción Técnicas de construcción de LEL & Escenarios	
Usuario Estratégico y/o Gerencial		Visión global de la empresa Expectativas del sistema a modelar	
Usuario Operativo		Conocimiento específico de la aplicación	
Tecnológicos:			
Procesador de textos			
Método:			
Técnicas de elicitación seleccionadas			

Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Listado de stakeholders	Recolección y análisis de información	Listado con función y responsabilidad relacionado con el sistema a modelar validado por el usuario.	Work product
Listado de fuentes de información adicional	Recolección y análisis de información	Nombre de la fuente y que información debe aportar.	Work product
Información del UdeD	Universo de Discurso	Información del sistema a modelar	Información
Lista DEO Descripción Verificación Validación	Describir Símbolos Verificar Símbolos Validar Símbolos	Si no esta vacía.	Work product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
Lista de palabras y frases	Identificar símbolos	Lista con todos las palabras, frases más repetidas o enfatizadas por el usuario o relevantes para es sistema en estudio	Work product
Conocimiento del dominio	Para todas las actividades posteriores	Miembros del equipo de desarrollo con conocimiento adquirido	Conocimiento
Síntesis de información	Identificar símbolos / Clasificar símbolos / Describir símbolos / Construcción de Escenarios	Resumen del conocimiento recolectado en la actividad	Información
Estándares:			
No aplicable			
Restricciones:			
Restricciones que surgen de las técnicas de elicitación seleccionadas y de la forma en las que se encuentra la información.			
Variables a medir:			
Nombre de la variable			
Usuarios entrevistados			
Entrevistas realizadas por usuario			

Documentos consultados por tipo	
Reuniones grupales	
Observaciones	

Código:		Nombre:	
2.02		Identificar Símbolos	
Objetivos:			
Determinar cuales son los símbolos que representan el lenguaje del dominio en análisis.			
Justificación:			
De la gran cantidad de términos que se utilizan en el lenguaje, debemos seleccionar los que tienen un significado particular, comprender el significado que tiene dentro de este contexto nos permitirá entender el dominio que estamos analizando			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Analizar lista de palabras y frases.	Generar una lista con los símbolos que tienen un significado particular dentro del dominio.	Líder de Proyecto / Analista de sistemas
2	Analizar fuentes adicionales de información	Confirmar y/o agregar más símbolos a la lista	Líder de Proyecto / Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs . 			
Método:			
LEL & Escenarios			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condición Y/O	Tipo
Lista de palabras y frases	Recolectar y analizar la información	Lista con todas las palabras, frases más repetidas o enfatizadas por el usuario o relevantes para es sistema en estudio	Work product

Síntesis de información	Recolección y análisis de información	Resumen del conocimiento recolectado en la actividad	Información
Lista DEO Descripción Verificación Validación Escenarios CRC	Describir Símbolos Verificar Símbolos Validar Símbolos Construcción de Escenarios Construcción de CRC	Si no esta vacía.	Work product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
Lista de símbolos	Clasificar símbolos	Lista de símbolos solamente con los que tiene un significado particular para el dominio.	Lista de símbolos
Estándares:			
<p>Nombre: Identificación del símbolo. Más de uno representa sinónimos</p> <p>Sintaxis: Palabra Frase</p>			
Restricciones:			
Solo modelar los símbolos que tienen un significado particular para el dominio en analisis			
Variables a medir:			
Nombre de la variable			
Símbolos identificados			
Observaciones			

Código:		Nombre:	
2.03		Clasificar símbolos	
Objetivos:			
Agrupar los símbolos según la clasificación seleccionada.			
Justificación:			
En esta actividad se agrupan los símbolos según la clasificación seleccionada, el resultado es la entrada a la siguiente en la que se describirán los símbolos. Para cada símbolo en particular varía su descripción según a que categoría pertenecen.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Generar lista de símbolos con clasificación	Agrupar los símbolos según la clasificación elegida	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Analista de sistemas		Técnicas de construcción de LEL & Escenarios Conocimiento del dominio de la aplicación	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			
Método:			
LEL & Escenarios Clasificación seleccionada para los símbolos			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Lista de símbolos	Identificar símbolos	Lista de símbolos solamente con los que tiene un significado particular para el dominio.	Work product
Síntesis de información	Recolección y análisis de información	Resumen del conocimiento recolectado en la actividad	Información
Lista DEO Descripción Verificación	Describir Símbolos Verificar Símbolos	Si no esta vacía.	Work product

Validación Escenarios CRC	Validar Símbolos Construcción de Escenarios Construcción de CRC		
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
Lista de símbolos	Describir símbolos	Todos los símbolos de la lista deben pertenecer a una categoría.	Work product
Estándares:			
<p>Nombre: Identificación del símbolo. Más de uno representa sinónimos</p> <p>Sintaxis: Palabra Frase</p> <p>Clasificación de símbolos propuesta en 2.00.</p>			
Restricciones:			
Los símbolos deben ser clasificados dentro de alguna de las categorías establecidas.			
Variables a medir:			
Nombre de la variable			
Símbolos del LEL por categoría			
Símbolos reclasificados			
Observaciones			
La clasificación que se toma como estándar, no es la única que se puede utilizar. Se puede crear una clasificación propia y utilizarla, de todas formas, esta nueva clasificación debe tener su relación con la estándar.			

Código:		Nombre:	
2.04		Describir símbolos	
Objetivos:			
El objetivo es describir los símbolos con su noción e impacto teniendo en cuenta la clasificación.			
Justificación:			
En esta actividad se describen los símbolos teniendo en cuenta las pautas para cada tipo, el resultado de esta actividad nos permite crear un medio de comunicación entre el equipo de desarrollo y el usuario para comprender el lenguaje del dominio.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Completar noción e impacto de los símbolos	Obtener la lista de símbolos completamente descriptos	Analista de sistemas
2	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Analista de sistemas		Técnicas de construcción de LEL & Escenarios Conocimiento del dominio de la aplicación	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			
Método:			
LEL & Escenarios Clasificación de símbolos Descripción de símbolos según su clasificación			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condición Y/O	Tipo
Lista de símbolos	Clasificar símbolos	Todos los símbolos de la lista deben pertenecer a una categoría de clasificación.	Work product

Síntesis de información	Recolección y análisis de información	Resumen del conocimiento recolectado en la actividad	Información
Conocimiento del UdeD	Universo de Discurso	Conocimiento del sistema a modelar	Conocimiento
Lista DEO Verificación Validación Escenarios CRC	Verificar Símbolos Validar Símbolos Construcción de Escenarios Construcción de CRC	Si no esta vacía.	Work product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
Lista de símbolos	Verificar LEL	Lista de símbolos completamente descripta según el estándar.	Work product
Lista DEO de Descripción (Discrepancias, Errores, u Omisiones)	Recolectar y analizar la información / Identificar Símbolos / Clasificar Símbolos	Si no esta vacía.	Work product
Estándares:			
<p>Nombre: Identificación del símbolo. Más de uno representa sinónimos</p> <p>Sintaxis: Palabra Frase</p> <p>Noción: Denotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo</p> <p>Sintaxis: Sentencia</p> <p>Impacto: Connotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando un vocabulario mínimo</p> <p>Sintaxis: Sentencia</p> <p><i>Sentencia está compuesto por Símbolos y No-Símbolos pertenecientes al vocabulario mínimo.</i></p> <p>Descripción de símbolos propuesta en 2.00</p>			
Restricciones:			
<p>Describir los símbolos con el formato establecido en el estándar</p> <p>Completar la noción e impacto según la clasificación del símbolo</p>			

Variables a medir:	
Nombre de la variable	
Símbolos en la lista definitiva del LEL	
Símbolos corregidos	
Símbolos eliminados	
Símbolos agregados	
Observaciones	

Código:		Nombre:	
2.05		Verificar LEL	
Objetivos:			
Asegurar que las pautas establecidas para la construcción del LEL se desarrollaron según el proceso definido.			
Justificación:			
En esta etapa se debe asegurar que fueron respetadas todas las pautas establecidas para la construcción del LEL, y la revisión de los símbolos en todos sus aspectos (Clasificación y Descripción) antes de realizar la validación con el usuario. Esta etapa apunta más al proceso que al producto en sí mismo.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Verificar clasificación	Comprobar que se clasificaron los símbolos de acuerdo a la clasificación adoptada	Líder de proyectos
2	Verificar descripción	Comprobar que se describieron los símbolos completamente y teniendo en cuenta la noción y el impacto según la clasificación elegida	Líder de proyectos
3	Verificar identificación	Comprobar que todos los símbolos pertenecientes al universo de discurso estén incluidos	Líder de proyectos
4	Verificar referencias	Comprobar el adecuado uso del principio de circularidad y vocabulario mínimo	Líder de proyectos
5	Confeccionar lista DEO	Construir una lista que contenga las diferencias, errores, u omisiones detectadas	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	

Analista de sistemas	Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios		
Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs Procesador de textos. 			
Método:			
<p>Verificar clasificación</p> <ul style="list-style-type: none"> Chequear que el símbolo corresponde con el tipo asignado <p>Verificar descripción</p> <ul style="list-style-type: none"> Chequear definición de símbolos Chequear completitud en la definición de símbolos Chequear que la descripción sea compatible con el tipo <p>Verificar identificación</p> <ul style="list-style-type: none"> Chequear si existen símbolos omitidos Chequear si existen sinónimos incorrectos Chequear la incorrecta inclusión de símbolos <p>Verificar referencias</p> <ul style="list-style-type: none"> Chequear falta de referencias a otros símbolos Chequera si existe mal uso de los símbolos 			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Lista de símbolos	Describir símbolos	Lista de símbolos completamente descripta según el estándar.	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
Lista de símbolos	Validar LEL	Lista de símbolos verificados	Work product
Lista DEO verificación	Recolectar y analizar información / Identificar símbolos / Clasificar símbolos / Describir símbolos	Si no esta vacía	Work product
Estándares:			
Se deben tener en cuenta todos los propuestos en 2.00			
Restricciones:			
Todas las establecidas en las actividades 2.01, 2.02, 2.03 y 2.04 Debe tomarse con sentido crítico y constructivo			
Variables a medir:			
Nombre de la variable			
Símbolos que deben ser modificados.			

Observaciones

Esta actividad debe desarrollarse con un gran sentido crítico, y no utilizar los resultados para buscar responsables de los errores. El responsable de asegurar esto es el Líder de proyectos.

Los analistas de sistemas que deben participar en esta actividad son los que participaron activamente en la actividad de Recolectar y analizar la información .

Código:		Nombre:	
2.06		Validar LEL	
Objetivos:			
Asegurarse que los símbolos modelados en el LEL son consistentes con lo esperado por el usuario y el equipo de desarrollo.			
Justificación:			
El desarrollar esta actividad nos permite asegurarnos que los símbolos modelados en el LEL, son consistentes con el léxico utilizado por el usuario / cliente.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Validar el LEL con el usuario	Analizar la consistencia entre los símbolos descriptos y el léxico del usuario	Líder de proyectos
2	Confeccionar lista DEO	Construir una lista que contenga las diferencias, errores, u omisiones detectadas	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas para realizar entrevistas	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas para realizar entrevistas	
Usuario Gerencial/Operativo		Conocimiento específico de la aplicación Conocimiento de cómo interpretar el LEL	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs • Procesador de textos. 			
Método:			
Entrevistas Validar el LEL <ul style="list-style-type: none"> • Validar descripción • Validar completitud 			

Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condición. Y/O	Tipo
Lista de símbolos	Verificar LEL	Lista de símbolos verificados	Work product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
Lista de símbolos del LEL	Construcción de Escenarios	Lista de símbolos validados por el usuario	Work product
Lista DEO de validación	Recolectar y analizar información / Identificar símbolos / Clasificar símbolos / Describir símbolos	Si no está vacía	Work product
Estándares:			
Restricciones:			
Realizar entrevistas.			
Variables a medir:			
Nombre de la variable			
Símbolos con errores			
Observaciones			

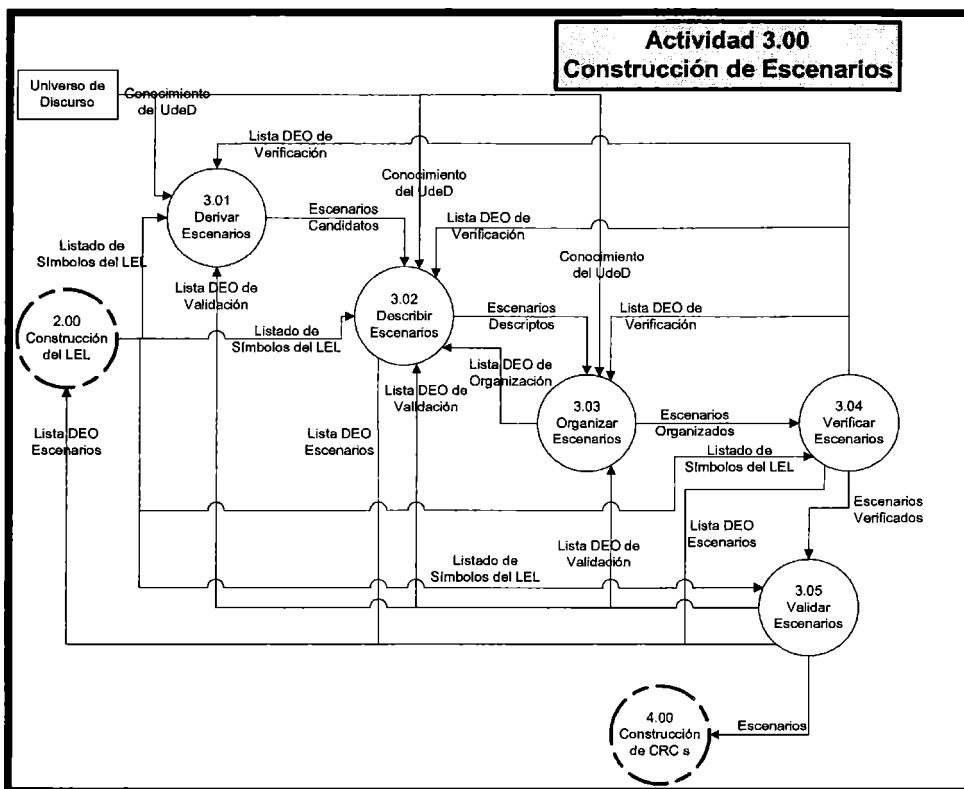
6.5- Construcción de Escenarios

Es la tercera actividad del proceso, por la complejidad que tiene, se desagrega en más actividades que su resultado final son los Escenarios que modelan el comportamiento esperado por el usuario para el sistema.

Las actividades que involucra son las siguientes:

- Derivar escenarios
- Describir escenarios
- Organizar escenarios
- Verificar escenarios
- Validar escenarios

En principio se describe la actividad Construcción de Escenarios, con un alto grado de abstracción para que permita tener una visión global de lo que esta actividad involucra, es por eso que dentro de esta actividad, la característica actividades menciona cada una de las actividades que involucra pero sin especificar el rol responsable de la ejecución. Luego se describe de forma detallada cada una de la actividades involucradas. En el DFD de la figura 6.3 se muestra la relación entre las actividades que involucra la creación del LEL.



Proceso LEC – Construcción de Escenarios
 Figura 6.3

Código:		Nombre:	
3.00		Construcción de Escenarios	
Objetivos:			
El objetivo de esta actividad es construir los escenarios para entender la aplicación.			
Justificación:			
Con esta actividad logramos capturar los requerimientos del usuario, utilizando su propio lenguaje modelado en el LEL. En esta actividad se modela el comportamiento esperado por el usuario para el sistema.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Derivar escenarios	Construir los escenarios que pueden obtenerse partiendo únicamente de los símbolos del LEL.	
2	Describir escenarios	Completar los escenarios candidatos, agregando información del dominio utilizando las heurísticas de descripción y tomando como base los símbolos del LEL.	
3	Organizar escenarios	Crear Escenarios Integradores, obtener una visión generalista de la aplicación.	
4	Verificar escenarios	Asegurar que las pautas establecidas para la construcción de Escenarios se desarrollaron según lo planificado.	
5	Validar escenarios	Asegurarse que los escenarios modelados son consistentes con lo esperado por el usuario y el equipo de desarrollo.	

Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas de elicitación	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas de elicitación	
Usuario Gerencial		Visión global de la empresa Conocimiento generales de la aplicación	
Usuario Operativo		Visión global de la empresa Conocimiento específicos de la aplicación	
Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs . 			
Método:			
Heurísticas para derivar escenarios Heurísticas para describir escenarios Heurísticas para organizar los escenarios Heurísticas para verificar escenarios Heurísticas para validar escenarios			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Conocimiento del dominio	Universo de discurso	Conocimiento sobre el sistema a modelar	Conocimiento
Síntesis de información	Recolección y análisis de información	Resumen del conocimiento recolectado en la actividad de construcción del LEL	Información
Lista DEO CRC	Construcción de CRC	Si no esta vacía	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
Escenarios	Construcción de CRC	Escenarios validados por el usuario	Work Product
Lista DEO Escenarios	Construcción del LEL	Si no esta vacía	Work Product

Estándares:

Escenario: descripción de una situación en el dominio de la aplicación.

Sintaxis: Título + Objetivo + Contexto + {Recursos} 1N + {Actores} 1N + {Episodios} 2N + {Excepciones}

Título: identificación del escenario. En el caso de un sub escenario, el título es el mismo que la sentencia

del episodio (ver en la definición de Episodios más abajo), sin las restricciones.

Sintaxis: Frase | ([Actor | Recurso] + Verbo + Predicado)

Objetivo: meta a ser alcanzada en el dominio de la aplicación. El escenario describe el cumplimiento del objetivo.

Sintaxis: [Actor | Recurso] + Verbo + Predicado

Contexto: compuesto al menos por uno de los siguientes sub componentes:

Ubicación Geográfica: ubicación física del escenario.

Ubicación Temporal: especificación de tiempo para el desarrollo del escenario.

Precondición: estado inicial del escenario.

Sintaxis: {Ubicación Geográfica} + {Ubicación Temporal} + {Precondición}

donde Ubicación Geográfica es:

Frase + {Restricción}

donde Ubicación Temporal es:

Frase + {Restricción}

donde Precondición es:

[Sujeto | Actor | Recuso] + Verbo + Predicado + {Restricción}

Recursos: elementos físicos relevantes o información que debe estar disponible en el escenario.

Sintaxis: Nombre + {Restricción}

Actores: personas, dispositivos u organización que tienen un rol en el escenario.

Sintaxis: Nombre

Episodios: conjunto de acciones que describen al escenario y proveen su comportamiento. Un episodio también puede ser descripto como un escenario.

Sintaxis (usando una representación tipo BNF):

<episodios> ::= <serie de grupos> | <serie de episodios>

<serie de grupos> ::= <grupo> <grupo> | <grupo no secuencial> | <serie de grupos> <grupo>

<grupo> ::= <grupo secuencial> | <grupo no secuencial>

<grupo secuencial> ::= <sentencia básica> | <grupo secuencial> <sentencia básica >

<grupo no secuencial> ::= # <serie de episodios> #

<serie de episodios > ::= <sentencia básica > <sentencia básica> |

<serie de episodios> <sentencia básica >

<sentencia básica> ::= <sentencia simple> | <sentencia condicional> | <sentencia opcional>

<sentencia simple> ::= <sentencia episodio> CR

<sentencia condicional> ::= **IF** <condición> **THEN** <sentencia episodio> CR

<sentencia opcional> ::= [<sentencia episodio >] CR

donde <sentencia episodio > es descripta como:

(([Actor | Recurso] + Verbo + Predicado) | ([Actor | Recurso] + [Verbo] + Título)) + {Restricción}

Excepciones: usualmente reflejan la falta o mal funcionamiento de un recurso necesario. Una excepción impide el cumplimiento del objetivo del escenario. El tratamiento de la excepción puede ser expresado por medio de otro escenario.

Sintaxis: Causa [(Solución)]

donde Causa es:

Frase | ([Sujeto | Actor | Recurso] + Verbo + Predicado)

donde Solución es:

Título

Restricción: alcance o requisito de calidad referido a una entidad dada. Es un atributo de Recursos, Episodios básicos o subcomponentes de Contexto.

Sintaxis: ([Sujeto Actor | Recurso] + [No] **Debe** + Verbo + Predicado) | Frase

+ *significa composición, {x} significa cero o más ocurrencias de x, () es usado para agrupamiento, | significa or, y [x] denota que x es opcional*

Restricciones:

Se debe partir de los símbolos modelados en el LEL

Variables a medir:

Nombre de la variable

Escenarios candidatos

Sub-escenarios

Escenarios modificados

Escenarios agregados

Escenarios eliminados

Escenarios integradores

Escenarios que deben ser modificados

Escenarios aceptados

Escenarios modificados

Escenarios eliminados

Escenarios agregados

Escenarios en la primer iteración

Escenarios definitivos

Observaciones

Código:		Nombre:	
3.01		Derivar Escenarios	
Objetivos:			
Construir los escenarios que pueden obtenerse partiendo únicamente de los símbolos del LEL.			
Justificación:			
El basarnos en los símbolos modelados en el LEL y aplicando las heurísticas de derivación, nos permite construir la lista de escenarios candidatos utilizando el propio lenguaje del usuario.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Identificar actores	Crear una lista con los actores primarios y secundarios	Analista de sistemas
2	Identificar escenarios	Crear una lista con los escenarios candidatos	Analista de sistemas
3	Crear escenarios	Construir escenarios basándose en los símbolos del LEL	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs . 			
Método:			
Identificar actores			
<ul style="list-style-type: none"> Identificar dentro del LEL los símbolos que representan actores del dominio de la aplicación . 			
Identificar escenarios			
<ul style="list-style-type: none"> Crear una lista de escenarios candidatos con cada impacto de los símbolos del LEL elegidos como actores, principales. Eliminar escenarios repetidos excepto uno. 			

Crear escenarios

- Analizar el contenido de cada impacto de los símbolo de tipo Sujeto que llevó a un escenario candidato, para encontrar símbolos del LEL del tipo Verbo.
 - **Si el impacto contiene un símbolo Verbo:**
 - Definir el objetivo según el título y la noción del símbolo Verbo.
 - Identificar los actores y recursos del escenario a partir de la información contenida en el símbolo Verbo y deben ser símbolos de tipo Sujeto y Objeto respectivamente.
 - Derivar los episodios a partir de cada uno de los impactos del símbolo Verbo.
 - **Si el impacto no contiene un símbolo Verbo:**
 - Definir el objetivo de acuerdo al título del escenario y el punto de vista de la aplicación.
 - Leer la definición completa de los símbolos, para seleccionar posibles actores y recursos.
 - Derivar actores de los símbolos de tipo Sujeto.
 - Derivar recursos de los símbolos de tipo Objeto.
- Extraer la ubicación geográfica y temporal del contexto puede ser extraída del impacto del símbolo sujeto del LEL que originó el escenario. La información relevante que debería ser registrada en las precondiciones del contexto podría estar disponible no sólo en ese impacto, sino también en la relación de secuencia con otros impactos del mismo símbolo.

Repetir el proceso para los actores secundarios

Input:

Nombre	Origen	Criterio de Entrada : Estado o Condición. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Lista DEO Verificación Validación	Verificar Escenarios Validar Escenarios	Si no está vacía	Work Product

Output:

Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
Escenarios candidatos	Describir escenarios	Escenarios candidatos sin episodios	Work Product

Estándares:

El propuesto en 3.00

Restricciones:

Solo se utilizan los símbolos del LEL para derivar los Escenarios candidatos.

Variables a medir:

Nombre de la variable	
Escenarios candidatos	

Observaciones

El Líder de proyectos si bien puede no participa directamente en las actividades que se realizan, es el responsable de controlar el cumplimiento de las actividades tal como fueron pensadas.

Código:		Nombre:	
3.02		Describir Escenarios	
Objetivos:			
<p>Completar los escenarios candidatos obtenidos en la etapa anterior, agregando información del dominio.</p>			
Justificación:			
<p>En la actividad anterior se obtuvo la lista de escenarios candidatos, en esta actividad se debe describir completamente los escenarios del sistema que estamos modelando. De la lista de escenario candidatos puede haber algunos que no sean modelados como escenarios en si mismos, sino como episodios o que directamente sean eliminados por no considerarse necesarios para el modelo.</p>			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Completar componentes de los escenarios	Describir los episodios de todos los escenarios	Analista de sistemas
2	Crear Sub-escenarios	Agrupar episodios que se repiten en distintos escenarios y crear uno nuevo	Analista de sistemas
3	Completar restricciones	Agregar las restricciones que corresponden para cada escenario	Analista de sistemas
4	Completar excepciones	Agregar las excepciones que corresponden para cada escenario	Analista de sistemas
5	Confeccionar lista DEO	Construir una lista que contenga las diferencias, errores, u omisiones detectadas	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de Escenarios Técnicas de elicitación	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de Escenarios Técnicas de elicitación	

Tecnológicos:			
Herramientas que soporten LEL & Escenarios & CRCs .			
Método:			
<p><u>Completar componentes</u> Realizar entrevistas estructuradas, observaciones y lectura de documentos. La información obtenida permite confirmar y mejorar el curso normal de eventos del escenario. Revisar descripciones iniciales de los escenarios parcialmente descriptos. Revisar episodios para confirmar orden secuencial o detectar paralelismos, y para encontrar episodios opcionales.</p> <p><u>Crear subescenarios</u> Se utilizan cuando: - se detecta comportamiento común en varios escenarios, - aparecen cursos de acción condicionales o alternativos complejos en un escenario, - se detecta en un escenario la necesidad de mejorar una situación con un objetivo concreto y preciso (Esta actividad se debe desarrollar pero puede ocurrir que no sea necesario crear escenarios de este tipo)</p> <p><u>Completar restricciones</u> Las restricciones se utilizan para caracterizar requisitos no funcionales aplicados a Contexto, Recursos y Episodios. Algunas pueden ser elicitadas desde el Dominio de la aplicación y otras pueden surgir examinando los episodios.</p> <p><u>Completar excepciones</u> Detectar los casos alternativos y excepciones. Algunas causas de excepción son elicitadas desde las fuentes de información mientras que otras pueden deducirse analizando los episodios y la no disponibilidad o malfuncionamiento de los recursos.</p>			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Escenarios candidatos	Derivar escenarios	Escenarios Derivados sin episodios	Work Product
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Conocimiento del dominio	Universo de discurso	Conocimiento sobre el sistema a modelar	Conocimiento
Síntesis de información	Recolectar y analizar información	Resumen del conocimiento recolectado en la actividad de construcción del LEL	Información

Lista DEO Verificación Validación Organización CRC	Verificar Escenarios Validar Escenarios Organizar Escenarios Construcción de CRC	Si no esta vacía	Work Product
--	---	------------------	--------------

Output:

Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
Escenarios	Organizar Escenarios	Escenarios completamente descriptos	Work Product
Lista DEO Escenarios	Construcción del LEL	Si no esta vacía	Work Product

Estándares:

Episodios: conjunto de acciones que describen al escenario y proveen su comportamiento. Un episodio también puede ser descripto como un escenario.
 Sintaxis (usando una representación tipo BNF):
 <episodios> ::= <serie de grupos> | <serie de episodios>
 <serie de grupos> ::= <grupo> <grupo> | <grupo no secuencial> | <serie de grupos>
 <grupo>
 <grupo> ::= <grupo secuencial> | <grupo no secuencial>
 <grupo secuencial> ::= <sentencia básica> | <grupo secuencial> <sentencia básica >
 <grupo no secuencial> ::= # <serie de episodios> #
 <serie de episodios > ::= <sentencia básica > <sentencia básica> |
 <serie de episodios> <sentencia básica >
 <sentencia básica> ::= <sentencia simple> | <sentencia condicional> | <sentencia opcional>
 <sentencia simple> ::= <sentencia episodio> CR
 <sentencia condicional> ::= **IF** <condición> **THEN** <sentencia episodio> CR
 <sentencia opcional> ::= [<sentencia episodio >] CR
 donde <sentencia episodio > es descripta como:
 (([Actor | Recurso] + Verbo + Predicado) | ([Actor | Recurso] + [Verbo] + Título)) + {Restricción}

Excepciones: usualmente reflejan la falta o mal funcionamiento de un recurso necesario. Una excepción impide el cumplimiento del objetivo del escenario. El tratamiento de la excepción puede ser expresado por medio de otro escenario.

Sintaxis: Causa [(Solución)]

donde Causa es:

Frase | ([Sujeto | Actor | Recurso] + Verbo + Predicado)

donde Solución es:

Título

Restricción: alcance o requisito de calidad referido a una entidad dada. Es un atributo de Recursos, Episodios básicos o subcomponentes de Contexto.

Sintaxis: ([Sujeto Actor | Recurso] + [No] **Debe** + Verbo + Predicado) | Frase

Restricciones:

Variables a medir:	
Nombre de la variable	
Sub-escenarios	
Escenarios modificados	
Escenarios agregados	
Escenarios eliminados	
Observaciones	

Código:		Nombre:	
3.03		Organizar Escenarios	
Objetivos:			
Crear Escenarios integradores que nos permitan tener una visión general de la aplicación.			
Justificación:			
<p>Esta actividad debe desarrollarse dado que las aplicaciones que se desarrollan, por lo general, tienen una alta complejidad y el número de escenarios puede ser muy difícil de manejar y los ingenieros de requerimientos puede analizar detalles muy puntuales, perdiendo la visión general de la aplicación. Con el fin de solucionar este problema, se propone la construcción de escenarios integradores, que dan una visión global de la relación existente entre varias situaciones, dado que cada episodio de un escenario integrador corresponde a un título de un escenario.</p> <p>Esta etapa no debería producir muchas modificaciones, durante su desarrollo, si en las anteriores se cumplió con tener un alto nivel de abstracción y amplio conocimiento del dominio de la aplicación.</p>			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Reorganizar escenarios	Aplicar operaciones de composición / descomposición	Analista de sistemas
2	Definir relaciones entre escenarios	Analizar las relaciones existentes entre los escenarios	Analista de sistemas
3	Integrar escenarios	Construir escenarios integradores	Analista de sistemas
4	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de Escenarios	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de Escenarios	
Tecnológicos:			
Herramientas que soporten LEL & Escenarios & CRC s.			

Método:**Reorganizar**

La reorganización consiste en juntar dos o más escenarios en uno, o dividir un escenario en uno o más.

Operaciones posibles de reorganizar

Empotrar subescenarios:

Si detectan subescenarios no relevantes con pocas ocurrencias en otros escenarios.

Agregar subescenarios:

Si se detecta un conjunto de episodios muy relevantes o un conjunto de episodios con diferente nivel de detalle en relación con el resto.

Consolidar escenarios:

Si dos o más escenarios sus objetivos y contextos son similares y tienen muchas coincidencias en los episodios.

Dividir escenarios:

Si existen varios episodios condicionales con la misma condición.

Fusionar escenarios:

Si existen escenarios que presentan un orden de precedencia contigua o secuencial, objetivos complementarios y contextos acoplados.

Partir escenarios:

Si hay una brecha temporal entre los episodios o cuando se detecta un contexto temporal muy largo.

Definir relaciones entre escenarios

Relación de superposición: Esta relación se observa principalmente cuando varios episodios comunes se presentan en diferentes escenarios.

Relación de orden: Se establece entre dos jerarquías de escenarios cuando una precede a otra.

Relación a través de excepciones: Se establece entre un escenario y aquellos escenarios que tratan sus excepciones.

Integrar

Detectar orden parcial entre jerarquías: Es realizado en base a comparaciones entre precondiciones o recursos o restricciones de la jerarquía contra los títulos.

Construir secuencia de jerarquías: Comienza con la revisión de la relación de la clasificación parcial descubierta entre las jerarquías. Esto se hace para clarificar cual es el set de jerarquías que preceden a uno dado

Construir el esqueleto de la integración: Este paso construye sólo los episodios componentes de los escenarios de integración.

Proponer Título, Objetivo y Contexto para los escenarios integradores: Es el paso dónde la nueva información semántica necesita ser agregada. Esto se hace mirando los escenarios involucrados en el escenario integración.

Input:

Nombre	Origen	Criterio de Entrada : Estado o Condición. Y/O	Tipo
Escenarios	Describir escenarios	Escenarios completamente descriptos	Work Product
Conocimiento del dominio	Universo de Discurso	Conocimiento sobre el sistema a modelar	Conocimiento
Lista DEO Verificación Validación CRC	Verificar Escenarios Validar Escenarios Construcción de CRC	Si no esta vacía	Work Product

Output:

Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
Escenarios	Verificar escenarios	Escenarios y escenarios integradores	Work Product
Lista DEO Organización	Describir Escenarios	Si no esta vacía	Work Product

Estándares:

El propuesto en 3.00.

Restricciones:

Utilizar solo los escenarios completamente descriptos.

La descomposición no debe ser aplicada cuando el objetivo no puede ser desacoplado

Variables a medir:

Nombre de la variable

Escenarios integradores

Observaciones

Las actividades de **reorganización** sirven como guía para mejorar la comprensión y administración de los escenarios. Esto debe verse como una buena práctica y no como una actividad obligatoria.

Cuando los objetivos son diferentes y sólo pueden ser expresados juntos usando conjunciones, las operaciones de Intercalar y fusionar no son recomendadas.

Cuando las ubicaciones geográficas y temporales podrían ser fragmentadas, se recomienda la operación Partir Escenarios.

Código:		Nombre:	
3.04		Verificar Escenarios	
Objetivos:			
Asegurar que las pautas establecidas para la construcción de Escenarios se desarrollaron según el proceso definido.			
Justificación:			
En esta actividad se debe asegurar que fueron respetadas todas las pautas establecidas para la construcción de escenarios antes de realizar la validación con el usuario.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Verificar consistencia interna	Controlar la sintaxis, semántica y relación de los componentes	Líder de proyectos
2	Verificar consistencia entre escenarios	Controlar subescenarios, contexto y equivalencias	Líder de proyectos
3	Verificar con el LEL	Chequear los símbolos del LEL y su relación con los componente de los escenarios.	Líder de proyectos
4	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios	
Tecnológicos:			
Herramientas que soporten LEL & Escenarios & CRCs . Procesador de textos.			

Método:**Verificar consistencia interna**Verificar sintaxis:

- Chequear la completitud de cada componente
- Chequear la existencia de más de un episodio por escenario
- Chequear la sintaxis de cada componente

Verificar relaciones entre componentes:

- Chequear que cada actor participe en al menos un episodio
- Chequear que cada actor mencionado en los episodios esté incluido en el componente actores
- Chequear que cada recurso sea usado en al menos un episodio
- Chequear que cada recurso mencionado en los episodios esté incluido en el componente recursos

Verificar semántica:

- Chequear coherencia entre el título y el objetivo
- Chequear que el conjunto de episodios satisface el objetivo y está dentro del contexto
- Chequear que las acciones presentes en las precondiciones ya han sido realizadas
- Chequear que los episodios sólo contienen acciones a ser realizadas
- Chequear el mismo nivel de detalle dentro del escenario

Verificar consistencia entre escenariosVerificar existencia de subescenarios:

- Chequear que cada episodio identificado como un subescenario existe dentro del conjunto de escenarios
- Chequear que el conjunto de episodios de cada subescenario no haya sido incluido en otro escenarios
- Chequear que cada excepción sea tratada por un escenario

Verificar contexto:

- Chequear que cada precondición es un hecho incontrolable o es satisfecha por otro escenarios
- Chequear la coherencia entre las precondiciones de los subescenarios y las precondiciones de los escenarios
- Chequear que las ubicaciones geográfica y temporal de los subescenarios son iguales o más restringidas que las de los escenarios

Verificar equivalencia en el conjunto de escenarios:

- Chequear que las coincidencias de objetivo sólo se den en situaciones diferentes
- Chequear que las coincidencias de episodios sólo se den en situaciones diferentes
- Chequear que las coincidencias de contexto sólo se den en situaciones diferentes

Verificar con el LEL

- Chequear que cada símbolo del léxico esté identificado
- Chequear el uso correcto de los símbolos del léxico
- Chequear que los actores sean preferentemente símbolos del tipo Sujeto
- Chequear que los recursos sean preferentemente símbolos del tipo Objeto
- Chequear que los impactos de los símbolos del tipo Sujeto estén cubiertos por los escenarios

Input:

Nombre	Origen	Criterio de Entrada : Estado o Condic. Y/O	Tipo
Escenarios	Organizar escenarios	Escenarios y escenarios integradores	Work Product
Lista de símbolos del LEL	Validar el LEL	Lista de símbolos validados por el usuario	Work Product

Output:

Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
Escenarios	Validar Escenarios	Escenarios verificados	Work Product
Lista DEO verificación	Derivar Escenarios Describir Escenarios Organizar Escenarios	Si no esta vacía	Work Product
Lista DEO Escenarios	Construcción del LEL	Si no esta vacía	Work Product

Estándares:

El propuesto en 3.00

Restricciones:

Todas las establecidas en 3.01, 3.02, y 3.03

Variables a medir:

Nombre de la variable	
Escenarios que deben ser modificados	

Observaciones

Esta actividad debe desarrollarse con un gran sentido crítico, y no utilizar los resultados para buscar responsables de los errores. El responsable de asegurar esto es el Líder de proyectos.

Los analistas de sistemas que deben participar en esta actividad son los que participaron activamente en la actividad de Recolectar y analizar la información .

Código:		Nombre:	
3.05		Validar Escenarios	
Objetivos:			
Asegurarse que los escenarios modelados son consistentes con lo esperado por el usuario y el equipo de desarrollo.			
Justificación:			
Esta actividad busca la aceptación del cliente/usuario de las especificaciones realizadas sobre el dominio de la aplicación que son las bases para el diseño de software. En el caso que existan diferencias crear la lista DEO que permita realizar las correcciones necesarias. Esta etapa tiene un beneficio adicional que es la elicitación de nueva información que surge de la interacción con el usuario.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Validar los escenarios con el usuario	Analizar la consistencia entre los escenarios y lo esperado por el usuario.	Líder de proyectos
2	Confeccionar lista DEO	Construir una lista que contenga las diferencias, errores, u omisiones detectadas	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas para realizar entrevistas	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL & Escenarios Técnicas para realizar entrevistas	
Usuario Gerencial/Operativo		Conocimiento general y específico respectivamente de la aplicación Conocimiento de cómo interpretar el LEL & Escenarios	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs • Procesador de textos. 			

Método:			
Entrevistas			
Registrar todas las discrepancias, errores y omisiones			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condición. Y/O	Tipo
Escenarios	Verificar Escenarios	Escenarios verificados	Work Product
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
Escenarios	Construcción de CRC	Escenarios validados por el usuario	Work Product
Lista DEO Validación	Derivar Escenarios / Describir Escenarios / Organizar Escenarios	Si no está vacía	Work Product
Lista DEO Escenarios	Construcción del LEL	Si no está vacía	Work Product
Estándares:			
Restricciones:			
Realizar entrevistas estructuradas			
Variables a medir:			
Nombre de la variable			
Escenarios aceptados			
Escenarios modificados			
Escenarios eliminados			
Escenarios agregados			
Escenarios en la primera iteración			
Escenarios definitivos			
Observaciones			

Código:		Nombre:	
4.00		Construcción de CRC	
Objetivos:			
En esta actividad se crean las tarjetas CRC que serán la base para realizar el diseño orientado a objetos.			
Justificación:			
Partiendo de la información obtenida en las etapas anteriores, podemos derivar las clases, responsabilidades y colaboraciones que surgen, para poder realizar una primera aproximación al diseño del software con metodología orientada a objetos.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Identificar clases	Determinar las clases primarias, secundarias y responsabilidades utilizando el LEL	
2	Refinar responsabilidades y colaboraciones	Clarificar las responsabilidades y colaboraciones utilizando los escenarios	
3	Verificar CRC	Determinar si las CRC s obtenidas modelan el comportamiento del sistema modelado en los escenarios y determinar si las responsabilidades y colaboraciones son las adecuadas.	
4	Validar CRC	Determinar junto con el usuario si las CRC s obtenidas modelan el comportamiento del sistema	
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objeto	

Analista de sistemas	Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objetos		
Usuario Operativo	Conocimiento específico respectivamente de la aplicación Conocimiento de cómo interpretar el LEL & Escenarios		
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			
Método:			
Heurísticas de creación de CRC s Principios de OO. Entrevistas.			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condic. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Escenarios	Construcción de Escenarios	Escenarios validados	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
CRC s		CRC s validadas por el usuario	Work Product
Lista DEO CRC	Construcción del LEL / Construcción de Escenarios	Si no esta vacía	Work Product
Estándares:			
Nombre de la Clase: Como se identifica a la clase			
Responsabilidades: Se especifica que es lo que debe resolver			
Colaboraciones: Se especifica con que otras clases que interactúa para cumplir con las responsabilidades			
Restricciones:			
Variables a medir:			
Nombre de la variable			
Clases derivadas			

Clases definitivas	
Clases que deben ser modificadas (Verificación)	
Clases que deben ser modificadas (Validación)	
Observaciones	

Código:		Nombre:	
4.01		Identificar clases	
Objetivos:			
Esta actividad se ocupa de determinar las clases primarias, secundarias, responsabilidades y colaboraciones utilizando el LEL.			
Justificación:			
Partiendo del LEL podemos derivar las clases, responsabilidades y colaboraciones que surgen, para poder realizar una primera aproximación al diseño del software con metodología orientada a objetos.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Identificar clases primarias	Determinar las clases primarias sus responsabilidades y colaboraciones utilizando el LEL	Analista de Sistemas
2	Identificar clases secundarias	Determinara las clases secundarias sus responsabilidades y colaboraciones utilizando el LEL	Analista de Sistemas
3	Definir clases	Analizar si las definiciones de las clases y responsabilidades es la adecuada	Analista de Sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Conocimientos básicos de Objeto	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Conocimientos básicos de Objetos	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			

Método:

Identificar clases primarias v sus responsabilidades

- Modelar los sujetos del LEL como clases candidatas
- Determinar las responsabilidades a partir de los impactos

Identificación de clases secundarias v sus responsabilidades

- Analizar los términos del LEL que corresponden a Objetos, Estados y Frases verbales.
 - Determinar si un Objeto del LEL es modelado como atributo o clase
 - Determinar si una Frases verbal del LEL se modela como clases o responsabilidad
 - Determinar si un Estado del LEL se modela como clases o atributos

Definir las responsabilidades de las clases secundarias utilizando el LEL

Determinar las responsabilidades de los términos correspondientes a:
 Objetos
 Frases Verbales
 Estados

Definición de las clases

- Elegir el nombre más corto y significativo para definir la clase
- Definir los nombres de las responsabilidades con verbos en infinitivo
- Analizar si el impacto no representa más de una acción
- Analizar responsabilidades redundantes

Input:

Nombre	Origen	Criterio de Entrada : Estado o Condic. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Lista DEO Verificación Validación	Verificar CRC Validar CRC	Si no esta vacía	Work Product

Output:

Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
Clases primaria y secundarias	Refinar responsabilidades y colaboraciones	Clases primarias y secundarias derivadas del LEL	Work Product

Estándares:	
Se debe tener en cuenta el propuesto en 4.00	
Restricciones:	
Solo derivar clases de los símbolos del LEL	
Variables a medir:	
Nombre de la variable	
Clases derivadas	
Observaciones	

Código:		Nombre:	
4.02		Refinar responsabilidades y colaboraciones	
Objetivos:			
En esta actividad se refinan las responsabilidades y colaboraciones utilizando los escenarios			
Justificación:			
Al definir las responsabilidades de las clases primarias y secundarias a partir de los impactos del LEL, aparecen naturalmente las clases que colaboran ya que son los términos del LEL que están en los impactos analizados que se han convertido en clases. En esta etapa se refinan las colaboraciones y responsabilidades, utilizando los escenarios			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Refinar responsabilidades y colaboraciones analizando el modelo de escenarios.	Clarificar las responsabilidades y colaboraciones utilizando los escenarios	Analista de Sistemas
2	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Conocimientos básicos de Objeto	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Conocimientos básicos de Objetos	
Tecnológicos:			
<ul style="list-style-type: none"> • Herramientas que soporten LEL & Escenarios & CRCs . • Procesador de textos. 			

Método:			
Refinar responsabilidades y colaboraciones analizando el modelo de escenarios			
<ul style="list-style-type: none"> • Analizar las estructuras de los escenarios que indiquen cursos alternativos de acción • Agregar responsabilidades básicas • Definir clases que no pertenecen al LEL • Analizar los verbos con semántica de registra • Analizar si la clase representa a un individuo o una colección de individuos 			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condic. Y/O	Tipo
Clases primaria y secundarias	Identificar clases	Clases primarias y secundarias derivadas del LEL	Work Product
Escenarios	Construcción de Escenarios	Escenarios validados	Work Product
Lista DEO Verificación Validación	Verificar CRC Validar CRC	Si no esta vacía	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condic. Y/O	Tipo
CRC s	Verificar CRC	CRC s con responsabilidades y colaboraciones refinadas	Work Product
Lista DEO CRC	Construcción del LEL / Construcción de Escenarios	Si no esta vacía	Work Product
Estándares:			
Nombre de la Clase: Como se identifica a la clase			
Responsabilidades: Se especifica que es lo que debe resolver			
Colaboraciones: Se especifica con que otras clases que interactúa para cumplir con las responsabilidades			
Restricciones:			
Utilizar el modelo de escenarios para refinar responsabilidades y colaboraciones			
Variables a medir:			
Nombre de la variable			
Clases definitivas			
Observaciones			

Código:		Nombre:	
4.03		Verificar CRC	
Objetivos:			
En esta actividad se ocupa de determinar si las CRC s obtenidas modelan el comportamiento del sistema modelado en los escenarios y determinar si las responsabilidades y colaboraciones son las adecuadas.			
Justificación:			
Con las CRC s obtenidas se debe analizar si modelan el comportamiento esperado, y además nos permite detectar DEO en las CRC s que pueden causar modificaciones en los modelos de LEL y Escenarios.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Ejecutar los escenarios sobre la base de las CRC s	Determinar si las clases y sus responsabilidades llevan a cabo las acciones de los episodios.	Analista de Sistemas
2	Aplicar principios de Orientación a Objetos	Determinar si la cantidad de responsabilidades y colaboraciones es adecuada.	Analista de Sistemas
3	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objeto	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objetos	
Usuario Operativo		Conocimiento específico respectivamente de la aplicación Conocimiento de cómo interpretar el LEL & Escenarios	

Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs . Procesador de textos. 			
Método:			
Principios de OO			
Entrevistas			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Cond. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Escenarios	Construcción de Escenarios	Escenarios validados	Work Product
CRC s	Verificar CRC	CRC s con responsabilidades y colaboraciones refinadas	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Cond. Y/O	Tipo
CRC s	Validar CRC	CRC s verificadas	Work Product
Lista DEO CRC	Construcción del LEL / Construcción de Escenarios	Si no esta vacía	Work Product
Lista DEO Verificación	Identificar Clases / Refinar responsabilidades y colaboraciones	Si no esta vacía	Work Product
Estándares:			
Restricciones:			
Variables a medir:			
Nombre de la variable			
Clases que deben ser modificadas			
Observaciones			
Esta actividad debe desarrollarse con un gran sentido crítico, y no utilizar los resultados para buscar responsables de los errores. El responsable de asegurar esto es el Líder de proyectos.			

Código:		Nombre:	
4.04		Validar CRC	
Objetivos:			
En esta actividad se ocupa de determinar, junto con el usuario, si las CRC s obtenidas modelan el comportamiento del sistema.			
Justificación:			
Con las CRC s obtenidas se debe analizar si modelan el comportamiento esperado, y además nos permite detectar DEO en las CRC s que pueden causar modificaciones en los modelos de LEL y Escenarios.			
Actividades:			
Paso	Nombre Actividad	Objetivo	Rol responsable de la ejecución
1	Validar las CRC s con los Usuarios	Determinar si los términos y comportamiento claves fueron bien interpretados en las CRC	Lider de proyectos / Analista de sistemas / Usuario Operativo
2	Confeccionar lista DEO	Registrar todos los DEO que se detectaron en la actividad de verificación	Analista de sistemas
Recursos:			
Humanos:			
Rol:		Capacitación requerida:	
Líder de proyectos		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objeto	
Analista de sistemas		Alto nivel de abstracción Conocimiento del dominio de la aplicación Técnicas de construcción de LEL, Escenarios & CRC Técnicas para realizar entrevistas Conocimientos básicos de Objetos	
Usuario Operativo		Conocimiento específico respectivamente de la aplicación Conocimiento de cómo interpretar el LEL & Escenarios	

Tecnológicos:			
<ul style="list-style-type: none"> Herramientas que soporten LEL & Escenarios & CRCs . Procesador de textos. 			
Método:			
Entrevistas Paráfrasis de CRC			
Input:			
Nombre	Origen	Criterio de Entrada : Estado o Condición. Y/O	Tipo
Lista de símbolos del LEL	Construcción del LEL	Lista de símbolos validados por el usuario	Work Product
Escenarios	Construcción de Escenarios	Escenarios validados	Work Product
CRC s	Verificar CRC	CRC s verificadas	Work Product
Output:			
Nombre	Destino	Criterio de Salida : Estado o Condición. Y/O	Tipo
CRC s		CRC s validadas por el usuario	Work Product
Lista DEO CRC	Construcción del LEL / Construcción de Escenarios	Si no está vacía	Work Product
Lista DEO Validación	Identificar clases / Refinar responsabilidades y colaboraciones	Si no está vacía	Work Product
Estándares:			
Restricciones:			
Variables a medir:			
Nombre de la variable			
Clases que deben ser modificadas			
Observaciones			

Cuadro de Work Products

El siguiente cuadro Figura 6.4 muestra los Work products obtenidos en cada una de las actividades del proceso y como se relaciona con el restos.

Código	Actividad	Input	Orig	Output	Dest
1.00	Definición de fuentes de información	Información del dominio a modelar	UdeD	Listado de stakeholders	2.01
				Listado de fuentes de información adicional	2.01
2.00	Construcción del LEL				
2.01	Recolectar y analizar información	Listado de stakeholders	1.00	Listado de palabras y frases	2.02
		Listado de fuentes de información adicional	1.00		
		Lista DEO Descripción	2.04		
		Lista DEO Verificación	2.05		
		Lista DEO Validación	2.06		
2.02	Identificar Símbolos	Lista de palabras y frases	2.01	Lista de símbolos	2.03
		Lista DEO Descripción	2.04		
		Lista DEO Verificación	2.05		
		Lista DEO Validación	2.06		
		Lista DEO Escenarios	3.00		
		Lista DEO CRC	4.00		
2.03	Clasificar Símbolos	Lista de símbolos	2.02	Lista de símbolos clasificados	2.04
		Lista DEO Descripción	2.04		
		Lista DEO Verificación	2.05		
		Lista DEO Validación	2.06		
		Lista DEO Escenarios	3.00		
		Lista DEO CRC	4.00		
2.04	Describir Símbolos	Lista de símbolos clasificados	2.03	Lista de símbolos Descriptos	2.05
		Lista DEO Verificación	2.05	Lista DEO Descripción	2.01 2.02 2.03
		Lista DEO Validación	2.06		
		Lista DEO Escenarios	3.00		
		Lista DEO CRC	4.00		
2.05	Verificar LEL	Lista de símbolos Descriptos	2.04	Lista de símbolos Verificados	2.06
				Lista DEO Verificación	2.01 2.02 2.03 2.04
2.06	Validar LEL	Lista de símbolos Verificados	2.05	Lista de símbolos del LEL Validados	3.00 4.00
				Lista DEO de Validación	2.01 2.02 2.03 2.04
3.00	Construcción de Escenarios				
3.01	Derivar escenarios	Lista de símbolos del LEL Validados	2.00	Escenarios Candidatos	3.02
		Lista DEO Verificación	3.04		
		Lista DEO Validación	3.05		
3.02	Describir escenarios	Escenarios Candidatos	3.01	Escenarios Descriptos	3.03
		Lista de símbolos del LEL Validados	2.00	Listado DEO Escenarios	2.00
		Lista DEO Verificación	3.04		
		Lista DEO Validación	3.05		
		Lista DEO Organización	3.03		
		Lista DEO CRC	4.00		
3.03	Organizar escenarios	Escenarios Descriptos	3.02	Escenarios incluyendo Integradores	3.04
		Lista DEO Verificación	3.04	Lista DEO Organización	3.02
		Lista DEO Validación	3.05		
		Lista DEO CRC	4.00		
3.04	Verificar escenarios	Escenarios incluyendo Integradores	3.03	Escenarios Verificados	3.05
		Lista de símbolos del LEL Validados	2.00	Lista DEO Verificación	3.01 3.02 3.03
				Lista DEO Escenarios	2.00
3.05	Validar escenarios	Escenarios Verificados	3.04	Escenarios Validados	4.00
		Lista de símbolos del LEL Validados	2.00	Lista DEO Validación	3.01 3.02 3.03
				Lista DEO Escenarios	2.00
4.00	Construcción de CRC s				
4.01	Identificar clases	Lista de símbolos del LEL Validados	2.00	Clases primarias y secundarias	4.02
		Lista DEO Verificación	4.03		
		Lista DEO Validación	4.04		
4.02	Refinar responsabilidades y colaboraciones	Clases primarias y secundarias	4.01	CRC con responsabilidades y colaboraciones refinadas	4.03
		Escenarios Validados	3.00	Lista DEO CRC	2.00 3.00
		Lista DEO Verificación	4.03		
		Lista DEO Validación	4.04		
4.03	Verificar CRC	Lista de símbolos del LEL Validados	2.00	CRC s Verificadas	4.04
		Escenarios Validados	3.00	Lista DEO Verificación	4.01 4.02
		CRC con responsabilidades y colaboraciones refinadas	4.02	Lista DEO CRC	2.00 3.00
4.04	Validar CRC	CRC s Verificadas	4.03	CRC s Validadas	

Figura 6.4

Caso de estudio

Objetivos del Capítulo

El objetivo de este capítulo es describir el caso de estudio sobre el cual se aplicó el proceso descrito y los resultados obtenidos.

Contenido

- 7.1- Introducción.
- 7.2- Síntesis del caso de estudio.
- 7.3- Resultados de cada una de las etapas.
- 7.4- Conclusión del caso.

7.1- Introducción

En la ingeniería de software, la actividad más dificultosa es la de determinar en forma precisa qué es lo que el usuario espera [Brooks 87].

Partir de especificaciones incorrectas, nos lleva a un producto de software que no cumple con las necesidades del usuario. Contar con una buena definición de requerimientos nos ayuda a lograr un producto que satisfaga los requerimientos del cliente.

Para probar el proceso propuesto utilizando el framework [Capítulo 5 y 6], se aplicó a un caso real.

7.2- Síntesis del caso de estudio

El caso de estudio es un sistema que permite a una universidad administrar todas las consultas realizadas por los alumnos a la secretaría de Asuntos Estudiantiles.

Cotidianamente se presentan diferentes casos problemáticos de distinta índole, algunos son aislados y otros comunes a una franja determinada de alumnos.

Un área importante de contacto es la Secretaría de Asuntos Estudiantiles(SEAS), que depende directamente del Rectorado.

Dentro de esta Secretaria existe un área de Atención y Orientación al estudiante, que es quien solicita el sistema, y de la que podemos tomar sus informes mensuales. En ellos hacen una síntesis de las consultas recibidas, como fuente de información.

Ante esta Secretaria se plantean problemas y sugerencias, esto exige comunicación directa con los responsables de cada sector para la resolución temprana de problemas detectados, también colabora con la imagen institucional mediante la explicación particular y personal en cada caso. Desde la Secretaria se facilita el monitoreo de la percepción de las actividades que tiene la comunidad estudiantil.

Actualmente el tratamiento de estas consultas se realiza de manera manual y requiere mucho esfuerzo para obtener estadísticas y realizar el seguimiento de las consultas. Para poder realizar una mejor administración de estas consultas, se desarrolló el Sistema de Consultas .

De la implementación del Sistema de Consultas la Secretaria espera obtener los siguientes impactos:

- Detección de problemas planteado por los alumnos
- Solución o planificación de cronograma de resolución de los problemas planteados por los alumnos
- Mejora de la calidad de servicios ofrecidos
- Incremento de participación y compromiso institucional del alumnado
- Monitoreo del clima organizacional por parte de los alumnos
- Refuerzo del sentimiento de pertenencia de los alumnos
- Representación del alumnado en instancias jerárquicas superiores

Recursos Humanos participantes en el proceso.

Equipo de desarrollo

El equipo de desarrollo estuvo constituido por cinco personas: Un Licenciado en Informática y cuatro Analistas de Sistemas.

El Licenciado en Informática asumió el rol de Líder de proyecto.

En la etapa de requerimientos participaron sólo dos analistas y en las etapas siguientes solo los otros dos.

La comunicación entre los analistas que participaron en la etapa de requerimientos y los que desarrollaron las etapas siguientes, se realizó por medio de la documentación generada en el proceso.

Usuarios Participantes

Los usuarios que participaron fueron los siguientes:

- Responsable del área de Atención y Orientación al Estudiante perteneciente a la Secretaría de Asuntos Estudiantiles (SEAS), con formación universitaria en bibliotecología y documentación, y especialización en gestión académica. Es quien solicita el sistema.
- Un Director de Departamento, con formación universitaria en sistemas, quien tiene a cargo la gestión académica de la Licenciatura e Ingeniería en Informática.
- Un Jefe de Área Administrativa, con formación universitaria en psicopedagogía.

Fuentes de información utilizadas

Para obtener los requerimientos se utilizaron las siguientes fuentes:

- Responsable del área de Atención y Orientación al estudiante.
- Director de Departamento y el Jefe de Área Académica.
- Documentación
 - E-mails enviados por los alumnos con sus consultas y las respuestas por los sectores a los que se derivan.
 - Consultas registradas manualmente
 - Documento de funciones del área de Atención y Orientación al Estudiante perteneciente a la Secretaría de Asuntos Estudiantiles.

Técnicas utilizadas

Las técnicas de elicitación utilizadas fueron las siguientes:

- Entrevistas individuales.
- Análisis de documentación e información.

Herramientas utilizadas

Para dar soporte al desarrollo del proceso utilizamos las siguientes herramientas:

- BMW [Antonelli 99]
- Procesador de textos

Estrategia general utilizada en el caso

La estrategia general utilizada en el caso fue seguir el proceso propuesto en el capítulo 6 de este trabajo.

En los puntos siguientes se detallan los resultados de cada una de las actividades y las decisiones tomadas.

Los nombres de las actividades propuestas en el proceso se utilizan como título para explicar su aplicación en el caso.

7.2.1-Determinación de fuentes de información

Dentro de *Determinación de fuentes de información*, tal como lo plantea el proceso, debemos realizar lo siguiente:

1. Seleccionar stakeholders del dominio a modelar.
2. Crear lista de stakeholders
3. Seleccionar otras fuentes de información para elicitar.
4. Crear lista de otras fuentes de información.
5. Validar con el usuario la lista de stakeholders y de fuentes de información.

Para *Seleccionar stakeholders del sistema a modelar* el Líder de proyecto realizó una entrevista con el responsable de Atención y Orientación al estudiante perteneciente al SEAS, que es quien solicitó el sistema.

En esta entrevista tuvimos que obtener, en primer lugar, una visión general del sistema a modelar para luego poder determinar los usuarios afectados.

Sin contar con esta visión general del sistema a modelar hubiera sido imposible determinar quienes son los usuarios involucrados.

El la entrevista el usuario del SEAS, que solicita el sistema, nos proporciono la siguiente documentación:

- Documento con funciones del área de Atención y Orientación al estudiante,
- e-mail de alumnos con consultas,
- consultas registradas manualmente,
- respuesta generadas por las distintas áreas a las consultas de los alumnos.

Analizando la información obtenida se detectaron como usuarios del sistema los doce Directores de departamento de las tres facultades y los diez Jefes de Áreas Administrativas que tienen relación con la atención de alumnos.

Para determinar los usuarios que participaron en la actividad de recolectar y analizar información, se tuvo en cuenta lo siguiente:

- cuales son las áreas que reciben más cantidad de consultas, y

- comentarios realizados por la persona entrevistada sobre quienes tienen un mayor compromiso con la resolución de las consultas,

Con el resultado de este análisis se determinó el *Listado de stakeholders* Figura 7.1.

Listado de stakeholders

Stakeholders	Responsabilidad
Responsable de Atención y Orientación al estudiante – SEAS (Solicita el sistema)	Recepción de consultas y dar respuestas a la que corresponden a su área
Director de Dpto. Tecnología Informática	Dar respuestas a las consultas que se relacionan con su área
Responsable de Equivalencias	Dar respuestas a las consultas que se relacionan con su área

Figura 7.1

Analizando los documentos ya mencionados se creó el *Listado de fuentes de información adicional* Figura 7.2.

Listado de fuentes de información adicional

Fuente Seleccionada	Información que debe aportar
Documento con funciones del área de Atención y Orientación al estudiante.	Para permitir determinar cual es el alcance que tienen sus funciones y como se relaciona con los demás sectores
e-mail enviados por alumnos	Analizar el tipo de consultas y las respuestas dadas
Consultas registradas manualmente por la secretaría	Analizar el tipo de consultas y las respuestas dadas
Respuestas generadas por las distintas áreas a las consultas de los alumnos	Analizar el tipo de respuesta dada.

Figura 7.2

Una vez que tuvimos estos dos listados completos se validaron con el usuario del SEAS por medio de una entrevista que realizó el Líder de proyecto.

El usuario consideró adecuada la elección de los usuarios que se van a relevar ya que tienen un fuerte compromiso con la resolución de las consultas.

Con respecto a los valores de las variables a medir que propone el proceso se pueden ver en la Figura 7.3

Variables medidas en Determinación de fuentes de información

Variables medidas:	
Nombre de la variable	
Usuarios detectados por jerarquía	
Usuarios seleccionados por jerarquía	
Total de usuarios por área	1
Usuarios seleccionados por área	1
Total de usuarios por función:	
Responder consultas	22
Generar consultas	1
Usuarios seleccionados por función	3

Figura 7.3

En *Determinación de fuentes de información*, cuando analizamos los valores de las variables, detectamos que no se justifica obtener cantidad de usuarios por jerarquía. Desde el punto de vista del *Sistema de Consultas* no existe una jerarquía entre sus distintos usuarios.

Cuando analizamos las variables *total de usuarios por áreas* y *usuarios seleccionados por área*, detectamos que para poder comprender lo que realmente queríamos transmitir se necesitaban aclaraciones.

En el caso de *total de usuarios por área* su valor es uno, pero no tenemos el total de áreas sobre las cuales estamos seleccionando, ni cuantas áreas están involucradas. La misma situación se plantea con la cantidad de *usuarios seleccionados por área*, donde de alguna manera debíamos aclarar que seleccionábamos un usuario por cada una de las tres áreas seleccionadas.

Del análisis de los valores de las dos variables, nos hace ver que debemos agregar dos variables a medir que son, *Total de áreas involucradas* y *Áreas seleccionadas*. Como así también modificar el nombre de las dos variables propuestas, *total de usuarios por área* a *Total de usuarios por áreas seleccionadas* y *Usuarios seleccionados por área* a *Usuarios seleccionados por área seleccionada*.

Esto nos hace ver que debemos modificar las variables a medir propuestas en el proceso.

Las modificaciones propuestas de variables, junto con los valores obtenidos en el proceso se pueden ver en la Figura 7.4.

Variables medidas en Determinación de fuentes de información(Modificado)

Variables medidas:	
Nombre de la variable	
Usuarios detectados por jerarquía	
Usuarios seleccionados por jerarquía	
Total de áreas involucradas	22
Áreas seleccionadas	3
Total de usuarios por área seleccionada	1
Usuarios seleccionados por área seleccionada	1
Total de usuarios por función :	
Responder consultas	22
Generar consultas	1
Usuarios seleccionados por función	3

Figura 7.4

7.2.2-Construcción del LEL

Dentro de *Construcción del LEL*, tal como lo plantea el proceso, debemos realizar lo siguiente:

1. Recolectar y analizar la información
2. Identificar símbolos
3. Clasificar símbolos
4. Describir símbolos
5. Verificar LEL
6. Validar LEL

El análisis del caso se realizó desde el punto de vista del SEAS quien recibe las consultas y desde el punto de vista de quienes responden estas consultas, ya sean de tipo administrativa o académica.

Recolectar y analizar información

El proceso plantea realizar lo siguiente para esta actividad:

1. Analizar fuentes de información
2. Preparar entrevistas
3. Aplicar técnicas de elicitación

Partiendo del *listado de stakeholders y fuentes de información adicional*, se procedió a *analizar la información* de las fuentes adicionales.

Una vez que se analizaron las fuentes en que nos basamos para elicitación, se prepararon las entrevistas.

La estrategia planteada para las entrevistas fue la siguiente:

- Preparar preguntas muy generales y elaborar preguntas más específicas durante el transcurso de las entrevistas.
- De los tres participantes del equipo de desarrollo en la etapa de requerimientos se decidió que:

El Líder de proyectos y un Analista realicen las entrevistas. El Líder de proyecto realizaba las preguntas y el Analista registraba, las palabras y frases propias del dominio y comentarios relevantes realizados por el entrevistado.

El tercer integrante del equipo se limitó a analizar la documentación proporcionada por el usuario y los resultados de las entrevistas.

- Realizar la primera entrevista a la responsable del área de Atención y Orientación al estudiante – SEAS. Esta elección fue realizada por considerar que, al ser este sector el que solicita el sistema, la información que nos puede proporcionar es muy relevante.
Esto nos permitió obtener mayor conocimiento del dominio, antes de realizar las entrevistas a los demás usuarios.
- Realizar las entrevistas al Director de Dpto. de Tecnología Informática y posteriormente al responsable de Equivalencias.

Como paso siguiente se procedió a *realizar las entrevistas* según el proceso planteado. Durante el desarrollo de las entrevistas se generó la *Lista de palabras y frases* que repetían con más frecuencia o enfatizaban los entrevistados.

Cabe aclarar que si bien la actividad *Identificar Símbolos* en el proceso propuesto, se describe como una etapa posterior, en realidad el analista la realiza a medida que transcurre la entrevista, registrando las frases y palabras que detecta con un significado importante dentro del dominio de la aplicación para generar después la lista de símbolos correspondientes al LEL.

Con respecto a los resultados para este caso de las variables a medir que propone el proceso se pueden ver en la Figura 7.5

Variables medidas en Recolectar y Analizar información

Variables medidas:	
Nombre de la variable	
Usuarios entrevistados	3
Entrevistas realizadas por usuario	2
Documentos consultados por tipo :	
e-mail	10
Consultas realizadas manuales	5
Documento de funciones	1
Reuniones grupales	0

Figura 7.5

En este caso se dio que con todos los usuarios se realizaron dos entrevistas, de no haber sido así se debería aclarar, de la misma forma que se hizo con los documentos, por cada usuario cuantas entrevistas se realizaron.

Identificar Símbolos

En esta actividad el proceso indica realizar lo siguiente:

1. Analizar lista de palabras y frases.
2. Analizar fuentes adicionales de información

En esta actividad la estrategia planteada fue la siguiente, construir la *lista de símbolos del LEL* entre los tres participantes del equipo de desarrollo en la etapa de requerimientos.

Antes de incorporar un símbolo a la lista se discutía entre los tres participantes.

Esto nos permitió crear una versión de la lista de símbolos en común desde las etapas más tempranas.

El contar en esta actividad, *Identificar Símbolos*, con un miembro del equipo de desarrollo que no participó de las entrevistas, nos permitió controlar la comprensión del significado de los símbolo planteados para el caso, antes de validarlo con el usuario.

Los valores obtenidos para las variables a medir que propone el proceso se pueden ver en la Figura 7.6

Variables medidas en Identificar Símbolos

Variables medidas:	
Nombre de la variable	
Cantidad de símbolos identificados	27

Figura 7.6

Lista de símbolos del LEL definitiva

Luego de dos iteraciones en el proceso, se llegó a la lista definitiva de símbolos del LEL.

Figura 7.7 - Primera columna

Clasificar Símbolos

El proceso en esta actividad propone *Generar lista de símbolos con clasificación*

La estrategia adoptada para generar la lista de símbolos clasificados, al igual que en la actividad identificar símbolos, fue:

- Que participen los tres integrantes del equipo de desarrollo involucrados en la etapa de requerimientos.
- Discutir la clasificación de los símbolos entre los tres integrantes del equipo antes de incorporarlo a la lista de símbolos clasificados.

Cuando se desarrolló la actividad Clasificar Símbolos, se debió elegir la clasificación que se utilizará para los símbolos.

Las alternativas fueron: elegir una clasificación para este caso en particular, o utilizar la clasificación estándar (Sujeto, Verbo, Objeto, Estado).

Se eligió esta última clasificación por considerar que si se planteaba una clasificación particular, de todas formas se debía marcar la relación con la estándar. Otro motivo fue estandarizar más el proceso, para facilitar su aplicación en trabajos futuros.

Con la *lista de símbolos* obtenida en la actividad *Identificar Símbolos* y las síntesis de información obtenida en la actividad de *recolectar y analizar la información* se procedió a clasificar los símbolos teniendo en cuenta la clasificación elegida y reflejarlo en la *Lista de Símbolos Clasificados*, Figura 7.3.

Lista de Símbolos Clasificados

<u>Símbolo</u>	<u>Clasificación</u>
alumno	Sujeto
asignar gravedad	Verbo
asignar respuesta	Verbo
asignar tiempo estimado	Verbo
asignar tiempo para la implementación	Verbo
carga fecha de respuesta	Verbo
consulta	Objeto
consulta abierta	Estado
consulta adeuda respuesta	Estado
consulta cerrada	Estado
consulta cerrada no resuelta	Estado
consulta con respuesta asignada	Estado
evaluar respuesta	Verbo
fecha de contacto	Objeto
fecha de derivación	Objeto
fecha de respuesta	Objeto
genera mail de derivación	Verbo
gravedad	Objeto
mail de derivación	Objeto
número de legajo	Objeto
referencia	Objeto
registra consulta	Verbo
respuesta	Objeto
secretaría	Sujeto
tiempo de respuesta estimado	Objeto
tiempo de respuesta real	Objeto
tiempo de solución establecido	Objeto
verificar estado de consulta	Verbo
área de derivación	Sujeto

Figura 7.7

En la Figura 7.8 se pueden ver los valores obtenidos para las variables que propone medir el proceso en *Clasificar Símbolos*.

Variables medidas en Clasificar Símbolos

Variables medidas:	
Nombre de la variable	
Símbolos del LEL por categoría:	
Sujeto	3
Verbo	9
Objeto	12
Estado	5
Símbolos reclasificados	0

Figura 7.8

Describir Símbolos

En esta actividad, según lo propone el proceso, se debe realizar lo siguiente:

1. Completar noción e impacto de los símbolos
2. Confeccionar lista DEO

Al igual que en las dos actividades anteriores, se decidió que:

- Participen los tres integrantes del equipo de desarrollo que realizan la etapa de requerimientos.
- Discutir entre los tres participantes la descripción de los símbolos antes de incorporarlos a la lista de símbolos completamente descriptos.

Con la *lista de símbolos clasificados*, la síntesis de información obtenida en la actividad *Recolectar y Analizar la información* y conocimiento obtenido del UdeD se procedió a *Describirlos símbolos* teniendo en cuenta la clasificación elegida.

En este punto se observó que al describir los símbolos respetando los principios de circularidad y vocabulario mínimo, la redacción de las nociones y los impactos en algunos casos no era clara. Por ejemplo donde debíamos poner un símbolo en plural lo teníamos reflejado en singular y viceversa, por ejemplo: consulta o consultas.

Para solucionar este inconveniente se nos ocurrieron alternativas, la primer alternativa es agregar cada una de estas palabras como sinónimos del símbolo, pero fue descartada por considerar que en realidad ampliaba la lista de sinónimos y el beneficio obtenido no lo justificaba.

La segunda alternativa, que es la que se adoptó para este caso, fue aceptar estas diferencias en la redacción, que no influyen en la comunicación.

Teniendo en cuenta estas decisiones se procedió a describir completamente los símbolos para obtener el *Listado de símbolos completamente descriptos*. El listado completo de símbolos descriptos se encuentra en el Anexo 1 de este trabajo.

En este caso la Lista DEO resultó vacía.

En Describir Símbolos el proceso propone medir las variables que se encuentran en la Figura 7.8

Variables medidas en Describir Símbolos

Variables medidas:	
Nombre de la variable	
Símbolos en la lista definitiva del LEL	29
Símbolos corregidos	2
Símbolos eliminados	2
Símbolos agregados	2

Figura 7.8

Los símbolos corregidos, agregados y eliminados se obtuvieron como resultado de la iteración en el proceso producido por la Lista DEO de Validación.

Verificar LEL

En actividad Verificar LEL el proceso propone hacer lo siguiente:

1. Verificar clasificación
2. Verificar descripción
3. Verificar identificación
4. Verificar referencias
5. Construir lista DEO

En la verificación del LEL no se detectaron errores.

En este punto consideramos que esto fue la consecuencia de realizar la identificación, clasificación y descripción de los símbolos de forma conjunta entre los tres participantes de la elicitación.

La verificación fue realizada en paralelo con las actividades anteriores, cabe aclarar que esto fue posible por ser un equipo de personas muy reducido, en caso que se deba trabajar en equipos más numerosos podría no darse los mismos resultados.

En este punto el proceso propone medir las variables que se pueden ver en la Figura 7.9 Si el valor de esta variable hubiese sido distinto de cero, deberíamos haber vuelto atrás en el proceso para modificar los errores detectados.

Variables medidas en Verificar LEL

Variables medidas:	
Nombre de la variable	
Cantidad de símbolos que deben ser modificados.	0

Figura 7.9

Validar LEL

En esta actividad, como lo indica el proceso se debe:

1. Verificar el LEL con el usuario
2. Construir la lista DEO

Para validar el LEL, se realizaron entrevistas con los usuarios.

La estrategia planteada en estas entrevistas fue la siguiente:

- Utilizar como soporte los documentos HTML generados por la herramienta BMW
El objetivo de utilizar esta técnica es que se quería evaluar la facilidad de comunicación que tiene el LEL con el usuario real.
- De la entrevista participaron el Líder de proyecto y el mismo analista que participó en las entrevistas anteriores.
- Realizar la primer entrevista al usuario del SEAS, que solicitó el sistema. Este usuario es el que tiene una visión integral del dominio, dado que es quien genera las consultas, y si le corresponde a su área, genera la respuesta correspondiente.
- Realizar entrevistas a los otros dos usuarios del sistema.

Al inicio de cada una de las entrevistas, en aproximadamente 10', se explicó al usuario el significados de los símbolos del LEL.

Luego, con la guía del Líder de proyecto, se revisaron uno por uno los símbolos modelados.

El Analista se limitó a registrar los problemas detectados, para luego confeccionar la lista DEO y hacer las modificaciones correspondientes.

Para revisar cada uno de los símbolos planteados en el LEL, se utilizaron los HTML en soporte magnético. Los símbolos se analizaron en orden alfabético.

En algunos caso se utilizó el vinculo que establece el LEL entre los símbolos para aclarar algunas dudas que surgían.

En las entrevistas se llegó a la conclusión que algunos símbolos reflejados en el LEL, como por ejemplo, “Consulta general cerrada / consulta parcial cerrada” no representaban la realidad, ya que se genera una consulta para cada pregunta que realice un alumno y no una general con muchas consultas parciales.

De esta validación surgió una *lista DEO* indicando que teníamos que modificar 4 símbolos y agregar 2 nuevos.

Con la información resultante de la validación se volvió a la actividad de *Describir símbolos* y se repitieron las etapas siguientes para obtener una nueva versión del LEL. Se generó en la segunda iteración una lista final de 29 símbolos que fue aceptada por el usuario. (Ver Lista definitiva LEL - Anexo1)

Merece destacarse que el usuario que solicitó el sistema de software, responsable del SEAS, al final de la entrevista de validación mencionó que: “.es muy sencillo comprender estos documentos y explican bien la situación...”.

Las variables que propone medir el proceso en *Validar LEL*, se pueden ver junto con los valores que toman en este caso, en la figura 7.10

Variables medidas en Validar LEL

Variables medidas:	
Nombre de la variable	
Símbolos con errores	4

Figura 7.10

Esto nos hace notar que debemos volver atrás en el proceso para solucionar los errores detectados. Cuando el resultado indique que no tenemos errores podemos comenzar con *Construcción de Escenarios*.

7.3.3-Construcción de Escenarios

En la Construcción de Escenarios, tal como lo plantea el proceso se debe hacer lo siguiente:

1. Derivar escenarios
2. Describir escenarios
3. Organizar escenarios
4. Verificar escenarios
5. Validar escenarios

Principalmente las decisiones tomadas para la implementación de *Construcción de Escenarios* fueron:

- Respetar en la redacción de los episodios los símbolos del LEL, aunque en algunos casos la redacción no quede clara. Este criterio fue tomado para obtener una mayor posibilidad de navegar en los HTML generados por la herramienta utilizada.
- Las actividades de Derivar, Describir y Organizar Escenarios, aprovechando las lecciones aprendidas en la construcción del LEL, fueron realizadas de manera conjunta por el Líder de proyecto y el Analista que realizaron las entrevistas y el Analista que solo se limitó a leer la documentación proporcionada por el usuario.

Derivar Escenarios

En esta actividad el proceso indica realizar lo siguiente:

1. Identificar actores
2. Identificar escenarios
3. Crear escenarios

Partiendo de la *Lista de símbolos del LEL* se analizaron, entre los tres participantes en la etapa de requerimientos, cuales símbolos correspondientes a la clasificación sujeto eran actores primarios y cuales actores secundarios.

El resultado de este análisis fueron dos actores primarios, la *secretaría* y *área de derivación* y un actor secundario, *alumno*.

Como paso siguiente se creo una lista con los impactos de estos actores, que produjo una lista de diez escenarios de los cuales dos fueron eliminados por estar repetidos. Por ejemplo “Responder consulta” que es una función que puede ser desarrollada por cualquiera de los dos actores primarios, obteniendo como resultado final una lista de ocho Escenarios.

Con estos resultados y completando el resto de los componentes, como lo establece el proceso, se procedió a generar la lista de *Escenarios Candidatos* Figura 7.11

Escenarios Candidatos

Alumno realiza la consulta
Alumno recibe la respuesta
Registrar consulta alumno
Asignar respuesta a consulta alumno
Emitir estadísticas de consultas
Evaluar respuesta dada a la consulta
Hacer seguimiento de consultas
Asignar tiempo para la implementación de la solución

Figura 7.11

En este punto el proceso propone medir las variables que se pueden ver en la Figura 7.12

Variables medidas en Derivar Escenarios

Variables medidas:	
Nombre de la variable	
Escenarios candidatos	8

Figura 7.12

Describir Escenarios

En esta actividad, según el proceso, se debe:

1. Completar componentes de los escenarios
2. Crear Sub-escenarios
3. Completar restricciones
4. Completar excepciones
5. Crear lista DEO

Con la lista de *Escenarios Candidatos* se procedió a describirlos utilizando los símbolos del LEL y la información recolectada en las actividades anteriores.

En el momento de describir los escenarios encontramos que 3 de ellos debían ser eliminados dado que no representaban escenarios en sí mismos.

Por ejemplo “El alumno realiza consulta” que fue tomado como información para describir el escenario de REGISTRA CONSULTA DE ALUMNO

Como resultado de esta actividad obtuvimos 5 *Escenarios descriptos*, Anexo 1 – Escenarios Descriptos, y no hubo elementos en la lista DEO.

En Describir Escenarios las variables que es proceso propone medir se pueden ver en la Figura 7.13, junto con los valores que toman en este caso.

VARIABLES MEDIDAS EN DESCRIBIR ESCENARIOS

VARIABLES MEDIDAS:	
Nombre de la variable	
Sub-escenarios	0
Escenarios modificados	0
Escenarios agregados	0
Escenarios eliminados	3

Figura 7.13

Organizar Escenarios

En la actividad Organizar escenarios, el proceso indica realizar lo siguiente:

1. Reorganizar escenarios
2. Definir relaciones entre escenarios
3. Integrar escenarios
4. Confeccionar lista DEO

Una vez que se obtuvieron los *escenarios descriptos* se analizaron las relaciones que existían entre ellos, como se propone en el proceso, y como resultado se obtuvo un escenario integrador que nos proporcionó una visión general de la aplicación.

En este caso la visión general de la aplicación la teníamos reconocida, pero no se había representado en ningún escenario. Pero creemos que en una aplicación de mayor envergadura y complejidad su aplicación aportará mayores beneficios. El escenario integrador de este caso se puede ver en la Figura 7.14.

Ejemplo de Escenario Integrador

Scenario: Gestionar *consulta*

Goal: Realizar la gestión de las *consulta* realizadas por los *alumno*

Context: el *alumno* necesita información sobre algún tema

Resources:

Actors:

Episodes: **Registrar consulta alumno**

Asignar respuesta a consulta realizada

Evaluar respuesta dada a la consulta

Emitir estadísticas de consulta

Hacer seguimiento de consulta

Links def usr:

Figura 7.14

En Organizar Escenarios se propone medir las variables que se pueden observar en la Figura 7.15

Variables medidas en Organizar Escenarios

Variables medidas:	
Nombre de la variable	
Escenarios integradores	1

Figura 7.15

Verificar Escenarios

En esta actividad el proceso propone realizar lo siguiente:

1. Verificar consistencia interna
2. Verificar consistencia entre escenarios
3. Verificar con el LEL
4. Confeccionar la lista DEO

En la verificación de los Escenarios no se detectaron errores.

Aprovechando las lecciones aprendidas en la actividad de *Describir símbolos*, se decidió aplicar la misma técnica para realizar la construcción de los escenarios de forma conjunta entre los tres participantes de la etapa de requerimientos.

La verificación fue realizada en paralelo con las actividades anteriores, cabe aclarar que esto se pudo dar por ser un equipo de personas muy reducido, en caso que se deba trabajar en equipos más numerosos podrían no darse los mismos resultados.

En este caso, como se puede ver en la Figura 7.16, los resultados de la variable que el proceso propone medir indica que no hay escenarios que deban modificarse.

Variables medidas en Verificar Escenarios

Variables medidas:	
Nombre de la variable	
Escenarios que deben ser modificados	0

Figura 7.16

Validar Escenarios

En la actividad Validar escenarios el proceso propone realizar lo siguiente:

1. Validar los escenarios con los usuarios
2. Confeccionar lista DEO

Para validar escenarios, se realizaron entrevistas con los usuarios.

La estrategia planteada en estas entrevistas fue la siguiente:

- Utilizar como soporte los documentos HTML generados por la herramienta BMW, LEL y Escenarios.
El objetivo de utilizar esta técnica es que se quería evaluar la facilidad de comunicación que tienen los escenarios con el usuario real.
- De la entrevista participaron el Líder de proyecto y el mismo analista que participó en las entrevistas anteriores.
- Realizar la primer entrevista al usuario que solicitó el sistema.
Este usuario es el que tiene una visión integral del dominio, dado que es quien genera las consultas, y si le corresponde a su área, genera la respuesta correspondiente.
- Realizar entrevistas a los otros dos usuarios del sistema.

Al inicio de cada una de las entrevistas, en aproximadamente 15', se explicó al usuario el significados de los componentes del modelo de escenarios.

Luego, con la guía del Líder de proyecto, se revisaron uno por uno los escenarios y su relación con los demás escenarios.

El Analista se limitó a registrar los problemas detectados, para luego confeccionar la lista DEO y hacer las modificaciones correspondientes.

Para validar los escenarios se utilizaron los HTML en soporte magnético, *Partiendo del Escenario Integrador* y recorriendo todos los escenarios. Es importante destacar que el mismo usuario navegaba por los HTML.

En ningún caso se tuvo la necesidad de volver sobre la definición de símbolos del LEL

En estas entrevistas se determinó que los escenarios modelaban el comportamiento del sistema tal como lo hacia el usuario, con lo cual fueron aceptados en la primera iteración.

El no haber encontrado modificaciones para realizar en este punto creemos que se debe a que se tuvo una buena definición del LEL y un sistema que no tiene una complejidad excesiva en su comportamiento.

Con respecto a los Escenarios, la opinión transmitida por los usuarios fue nuevamente muy favorable. Esto nos sirvió para confirmar que tanto el LEL como los Escenarios son una buena herramienta para la comunicación con el usuario.

En la Figura 7.17 se pueden ver los valores de la variables que el proceso propone medir en *Validar Escenarios*.

Variables medidas en Valida Escenarios

Variables medidas:	
Nombre de la variable	
Escenarios aceptados	6
Escenarios modificados	0
Escenarios eliminados	0
Escenarios agregados	0
Escenarios en la primer iteración	6
Escenarios definitivos	6

Figura 7.17

7.3.4- Construcción de CRC s

En *Construcción de CRC s* el proceso propone realizar lo siguiente:

1. Identificar clases
2. Refinar responsabilidades y colaboraciones
3. Verificar CRC
4. Validar CRC

En esta actividad, aprovechando las lecciones aprendidas en las anteriores, se ratificó que participaran nuevamente los tres participantes de etapa de requerimientos.

Identificar clases

En esta actividad el proceso propone hacer lo siguiente:

1. Identificar clases primarias
2. Identificar clases secundarias
3. Definir clases

Teniendo en cuenta que en este nivel la idea es modelar el macrosistema, se procedió a aplicar los métodos tal como se plantea en el proceso.

Partiendo de la *Lista de símbolos del LEL* se aplicaron las heurísticas para derivar las clases primarias, secundarias y sus responsabilidades y obtuvimos 8 clases.

Las variables que el proceso propone medir, junto con los valores que toma para este caso, se pueden ver en la Figura 7.18

Variables medidas en Identificar Clases

Variables medidas:	
Nombre de la variable	
Clases derivadas	8

Figura 7.18

Refinar responsabilidades y colaboraciones

En esta actividad el proceso indica realizar lo siguiente:

1. Refinar responsabilidades y colaboraciones analizando el modelo de escenarios.
2. Confeccionar lista DEO

En este punto se aplicaron las heurísticas correspondientes y nos surgieron algunas dudas.

A modo de ejemplo, nos surgió la duda si construíamos los distintos estados de la *Consulta* como objetos o como atributos. La decisión fue dejarlos en esta instancia como atributos pensando en que es más fácil para la interpretación del usuario.

Otra situación problemática planteada fue la clase *Secretaría* que se derivó del LEL, se elimina posteriormente en el diseño por ser una instancia de *Area de derivación*, pero para la comunicación con el usuario decidimos conservarla para facilitar la interpretación del usuario.

El proceso propone medir la variable *Clases definitivas*, que se pueden ver en la Figura 7.19.

Variables medidas en Refinar responsabilidades y colaboraciones

Variables medidas:	
Nombre de la variable	
Clases definitivas	5

Figura 7.19

Verificar CRC

En esta actividad el proceso indica realizar los siguiente:

- Ejecutar los escenarios sobre las bases de las CRC s
- Aplicar principios de OO
- Confeccionar lista DEO

En la *ejecución de los escenarios sobre la base de las CRC s* no se detectaron problemas.

No sucedió lo mismo en la *aplicación de los principios de orientación a objetos* por las decisiones tomadas en las dos actividades anteriores, *identificar clases y refinar responsabilidades y colaboraciones*. Por ejemplo, modelar secretaría como clase, cuando en realidad es una instancia de *área de derivación*.

En este caso los resultados fueron que no debía modificarse ninguna clase, como se puede ver en las variables a medir que propone el proceso Figura 7.20.

Variables medidas en Verificar CRC

Variables medidas:	
Nombre de la variable	
Clases que deben ser modificadas	0

Figura 7.20

Validar CRC

En esta actividad, Validación de CRC, el proceso indica realizar lo siguiente:

- Validar las CRC s con el usuario
- Confeccionar lista DEO

La estrategia planteada en esta actividad para realizar la entrevista fue la siguiente:

- De la entrevista participaron el Líder de proyecto y el mismo analista que participó en las entrevistas anteriores.
- Se utilizó paráfrasis de CRC s.

De las entrevistas realizadas con el usuario que solicitó el sistema para validar las CRC, siguiendo el método propuesto en el proceso no surgieron modificaciones para realizar. Consideramos que dentro de la actividad Construcción de CRC, no es tan importante y productivo realizar la validación con el usuario, dado que esto requiere un conocimiento técnico para poder realizar la actividad con resultados beneficiosos para el modelo que estamos realizando.

El proceso propone medir la cantidad de clases que deben ser modificadas, que en este caso fue cero como puede verse en la figura 7.21

Variables medidas en Validar CRC

Variables medidas:	
Nombre de la variable	
Cantidad de clases que deben ser modificadas	0

Figura 7.21

Las CRC s obtenidas como resultado de la aplicación del proceso se encuentran en el Anexo 1 de esta tesis.

7.4 Conclusión del Caso

Algunos aspectos importantes sobre el ambiente en el que se desarrolló el caso fueron:

- El usuario que solicitó el producto, tiene el suficiente respaldo político para lograr que los demás se involucren en el proceso.
- Contar con usuarios que tienen formación universitaria, facilitó la interpretación de las herramientas utilizadas y la comunicación.
- El sistema era relativamente sencillo.

Al momento de cerrar el informe de este trabajo, el sistema se encuentra en etapa de prueba y de obtener los resultados esperados se implementará de manera definitiva.

Conclusiones y Trabajos futuros

Objetivos del Capítulo

El objetivo de este capítulo es transmitir las conclusiones obtenidas de la aplicación del proceso.

Contenido

8.1-Conclusiones.

8.2-Aporte realizados por este trabajo

8.3-Trabajos futuros

8.1- Conclusiones.

En este trabajo se propuso un framework para dar soporte al proceso de elicitación requerimientos basado en lenguaje natural, utilizando LEL, Escenarios y CRC.

Los objetivos perseguidos fueron,

- obtener un medio de capacitación para quienes no conozcan las técnicas y herramientas utilizadas
- servir de guía para quienes apliquen la metodología en un proceso real.

En la aplicación en el caso de estudio de esta tesis los beneficios obtenidos fueron los siguientes:

- Contar con el framework facilitó y simplificó considerablemente mantener un criterio común sobre las actividades y objetivos perseguidos en cada etapa. Esto permitió enfocar los esfuerzos en los aspectos que eran de interés para cada etapa en particular.
- La formación de los ingenieros de requerimientos, tiene una influencia directa en los resultados de la etapa, en este caso se puso como pre-condición manejar un alto grado de abstracción. Esto nos permitió obtener de situaciones particulares transmitidas por los usuarios una representación abstracta para modelar símbolos del LEL y posteriormente modelar el comportamiento en los escenarios.
- Disponer de las definiciones de las variables que debíamos medir, ayudó a evaluar los resultados de la aplicación del proceso.
- Elegir la clasificación “estándar” de símbolos facilitó la determinación de los símbolos del LEL debido a que esa clasificación tiene un entendimiento común, independientemente del dominio de la aplicación y puede ayudar al reuso de símbolos.
- Contar con una herramienta como BMW, facilitó considerablemente la documentación y la comunicación entre los miembros del grupo y de estos con el usuario. Si bien no se dispone de los datos precisos, la comparación con las experiencias de construcción manual de LEL y Escenarios otorga una considerable ventaja al uso de una herramienta CASE.

Como síntesis final concluimos que:

- En el caso del proceso LEC el framework sirvió para encontrar algunas falencias, precisar detalles y profundizar actividades.
- Contar con un framework definido para un proceso nos permite controlar el desarrollo.
- El proceso LEC es factible aplicarlo en el desarrollo de un caso real.
- Los métodos basados en lenguaje natural para modelar requerimientos son un buen medio de comunicación con el usuario final.
- La utilización de una herramienta CASE en el proceso de elicitación posee ventajas en cuanto a calidad del producto, productividad de los analistas y comunicación con el usuario.

8.2- Aportes realizados por este trabajo

- Definición de un framework y el template para definir procesos.

De analizar las propuestas de Basili y Olson, ninguna de las dos nos pareció adecuada para definir el proceso.

Con lo cual teniendo como base estas dos propuestas se creó un nuevo framework y template para definir procesos.

- El framework y el template propuesto permiten definir procesos del estilo que requiere CMM y que pueden ser implementados.

El CMM requiere procesos definidos, documentados y estandarizados. El proceso LEC es un proceso de elicitación definido, documentado y estandarizado.

En este caso utilizamos las Key Process Questions [Olson 93], para poder determinar si el framework y el template propuesto cumplen con los elementos necesarios y si es capaz de ser implementado.

- Verificación de la aplicabilidad del framework definido a través de la especificación del proceso LEC basado en lenguaje natural utilizando el framework propuesto.

Utilizando el framework y el template propuesto en el capítulo 5, se definió el proceso LEC, basado en LEL, Escenarios y CRC.

- Definir con mayor rigurosidad las actividades que deben desarrollarse, para aplicar LEL, Escenarios y CRC s.

Utilizar el framework y el template propuesto, nos hizo notar que debíamos definir con mayor precisión las actividades, recursos y productos con sus criterios de entrada y salida.

- Definición de una etapa exclusivamente destinada a determinar los stakeholders que deben participar de la elicitación.

Esta etapa se planteó en el proceso con el mismo nivel de jerarquía que, por ejemplo, *Construcción del LEL*.

No seleccionar los participantes adecuados, más allá de aplicar de manera adecuada el proceso de elicitación, nos conduce a crear un sistema que no satisfaga las necesidades del usuario.

- Verificación que el proceso LEC definido puede ser aplicado en un caso real.

El proceso LEC fue aplicado al desarrollo de un sistema de un caso real. Esto nos permitió probar su factibilidad de implementación y llegar a un software que cumple con los requerimientos del usuario.

Este sistema se encuentra en etapa de prueba y de brindar los beneficios esperados, se implementará de forma definitiva.

- Demostrar que aplicando el proceso LEC que se definió se construye un software que funciona.

En las etapas posteriores a la de requerimientos, solamente se utilizaron como medio de comunicación la documentación resultante de aplicar el proceso LEC.

- Comprobar las ventajas de utilizar una Herramienta CASE como soporte al proceso de elicitación.

Para dar soporte a todo el proceso LEC se utilizó la herramienta BMW. Esto nos permitió tener una fuente única de documentación.

Otro beneficio obtenido fue la opción generar HTML, que fueron utilizados para la validación con el usuario.

- Se pudo evaluar la participación del usuario real en el proceso de elicitación.

El desarrollo del caso de estudio se realizó con la participación de los usuarios reales del sistema.

- Evaluar el impacto en la comunicación con el usuario real que tiene utilizar LEL y Escenarios.

Para realizar la validación con el usuario, se utilizaron el LEL y los Escenarios producidos.

En ambos caso con los HTML generados por la herramienta BMW.

- Utilizar HTML s en soporte magnético para validar con el usuario el conocimiento modelado en el LEL y los Escenarios.

Para obtener los beneficios que brindan los HTML se utilizaron en soporte magnético. Esto nos permitió realizar la validación con el usuario de una forma mas amigable.

- Se detectó la ventaja de contar con una visión general del sistema, para poder definir los usuarios que participarán en el proceso de elicitación.

Para poder realizar una buena elección de los participantes en la elicitación, debemos tener en principio una visión global del sistema que tenemos que modelar.

Esto nos permite ver qué áreas y quiénes están involucrados, y qué relación tienen con el sistema.

Analizar la cantidad de usuarios y funciones para luego poder determinar quienes participarán en la etapa de elicitación.

- Se detectaron ventajas al dividir funciones dentro del equipo en cuanto a la interacción con el usuario.

Realizar las actividades del proceso, como por ejemplo *Describir símbolos del LEL*, con la participación de un integrante del equipo de desarrollo que no

participó de las entrevistas ayuda a mejorar la calidad del LEL, Escenarios y CRC obtenidos.

Este integrante utiliza como fuentes de información únicamente la documentación proporcionada por el usuario y los resultados obtenidos en las entrevistas.

8.3-Trabajos Futuros.

- Ampliar el proceso de verificación del proceso a través de aplicarlo en otro proyecto real de mayor envergadura y que posea mayor “representatividad”.
- Crear los documentos que dan soporte al proceso propuesto.
- Ampliar las métricas que ayuden a mejorar la calidad del proceso en su implementación.
- Desarrollar una herramienta CASE que brinde un soporte integral a la elicitación en cuanto a los productos a elaborar y los procesos utilizados.

Documentos del Caso de Estudio

En este anexo se encuentran todos los Work products producidos en la aplicación del proceso al caso:

Work Products

Listado de stakeholders

Stakeholders	Responsabilidad
Responsable de Atención y Orientación al estudiante – SEAS (Solicita el sistema)	Recepción de consultas y dar respuestas a la que corresponden a su área
Director de Dpto. Tecnología Informática	Dar respuestas a las consultas que se relacionan con su área
Responsable de Equivalencias	Dar respuestas a las consultas que se relacionan con su área

Listado de fuentes de información adicional

Fuente Seleccionada	Información que debe aportar
Documento con funciones del área de Atención y Orientación al estudiante.	Para permitir determinar cual es el alcance que tienen sus funciones y como se relaciona con los demás sectores
e-mail enviados por alumnos	Analizar el tipo de consultas y las respuestas dadas
Consultas registradas manualmente por la secretaria	Analizar el tipo de consultas y las respuestas dadas
Respuestas generadas por las distintas áreas a las consultas de los alumnos	Analizar el tipo de respuesta dada.

Lista de símbolos del LEL

alumno

asignar gravedad

asignar respuesta

asignar tiempo estimado

asignar tiempo para la implementación

carga fecha de respuesta

consulta

consulta abierta

consulta adeuda respuesta

consulta cerrada

consulta cerrada no resuelta

consulta con respuesta asignada

Evaluar respuesta

fecha de contacto

fecha de derivación

fecha de respuesta

genera mail de derivación
gravedad
mail de derivación
número de legajo
referencia
registra consulta
respuesta
secretaría
tiempo de respuesta estimado
tiempo de respuesta real
tiempo de solución establecido
verificar estado de consulta
área de derivación

Lista de Símbolos Clasificados

<u>Símbolo</u>	<u>Clasificación</u>
alumno	Sujeto
asignar gravedad	Verbo
asignar respuesta	Verbo
asignar tiempo estimado	Verbo
asignar tiempo para la implementación	Verbo
carga fecha de respuesta	Verbo
consulta	Objeto
consulta abierta	Estado
consulta adeuda respuesta	Estado
consulta cerrada	Estado
consulta cerrada no resuelta	Estado
consulta con respuesta asignada	Estado
evaluar respuesta	Verbo
fecha de contacto	Objeto
fecha de derivación	Objeto
fecha de respuesta	Objeto
genera mail de derivación	Verbo
gravedad	Objeto
mail de derivación	Objeto
número de legajo	Objeto
referencia	Objeto
registra consulta	Verbo
respuesta	Objeto
secretaría	Sujeto
tiempo de respuesta estimado	Objeto
tiempo de respuesta real	Objeto
tiempo de solución establecido	Objeto
verificar estado de consulta	Verbo
área de derivación	Sujeto

Lista Símbolos descriptos**Tipo Sujeto**

Lel entry: alumno

Notion: Cualquier persona que se encuentra inscrita en la universidad para cursar una materia o un año de una carrera

Behavioral responses: Realiza la consulta
Recibe la respuesta

Lel entry: secretaria

Notion: área de derivación administrativa de la universidad que se ocupa de: Servicios de atención y orientación al estudiante.
Becas, Servicio de empleos, Actividades culturales, solidarias y deportivas

Behavioral responses: Registra consulta realizada por el alumno
Hace el seguimiento de la consulta
Evaluar respuesta
Genera respuesta
asignar tiempo para la implementación de la solución
realiza estadísticas de las consulta registradas

Lel entry: área de derivación

Notion: Sector de la universidad académico o administrativo a la que se deriva la consulta realizada por el alumno para su resolución.

Behavioral responses: Genera respuesta
asignar tiempo para la implementación de la solución

Tipo Verbo

Lel entry: asignar gravedad

Notion: cuando se recibe una consulta se evalúa la importancia y en base a esto se determina el nivel de gravedad de la consulta

Behavioral responses: Se registra el nivel de gravedad asignado a la consulta

Lel entry: asignar respuesta

Notion: El área de derivación que recibe la consulta , la evalúa y propone una solución

Behavioral responses: Se debe registrar la respuesta

Se debe asignar estado de consulta con respuesta asignada

Se debe cargar fecha de respuesta

Se debe calcular tiempo de respuesta real

Se debe genera mail de derivación para la secretaría

Lel entry: asignar tiempo estimado

Notion: La secretaría según el tipo de problema, asigna un plazo para el cual se debe tener la respuesta a la consulta

Behavioral responses: Se registra el tiempo estimado para la consulta

Lel entry: asignar tiempo para la implementación

Notion: El área de derivación, cuando proporciona una respuesta a la consulta establece el tiempo en el cual tendrá implementada la solución dada

Behavioral responses: Se registra el tiempo establecido para la consulta

Lel entry: carga fecha de respuesta

Notion: Cuando se ingresa la respuesta al sistema, se debe actualizar la fecha de respuesta con la fecha del día

Behavioral responses: Se registra la fecha de respuesta para la consulta

Lel entry: Evaluar respuesta

Notion: Cuando se recibe la respuesta del área de derivación se verifica si es una respuesta aceptable o no

Behavioral responses: Hace que se cambie el estado de la consulta a consulta cerrada

Envía un mail de derivación para solicitar otra respuesta.

Lel entry: genera mail de derivación

Notion: La secretaría emite un e-mail con la consulta para el área de derivación, para notificar la consulta a la que debe responder

El área de derivación emite un e-mail para notificar a la secretaría de la respuesta dada a la consulta

Behavioral responses: Es el que dispara el proceso que debe realizar el área de derivación para dar la solución

Lel entry: registra consulta

Notion: La secretaría, cada vez que un alumno se acerca con una inquietud, la registra para ser derivada al área de derivación correspondiente para que de una respuesta

Behavioral responses: Produce que se deba asignar un área de derivación.

Se debe asignar tiempo estimado de respuesta

Se debe generar mail de derivación para el área de derivación asignada

Se debe registrar la fecha de contacto

Se debe registrar la fecha de derivación

Se debe asignar estado de consulta general abierta

Se debe registrar estado de consulta parcial abierta

Lel entry: verificar estado de consulta

Notion: Esta acción se realiza de manera automática todos los días comparando la fecha de respuesta asignada, con la fecha del día

Behavioral responses: Se debe mostrar todas las consulta que se encuentran en este estado de respuesta y estado consulta abierta

Tipo Objeto

Lel entry: consulta

Notion: Define cual o cuales son los motivos por los cuales el alumno se acerca a plantear una inquietud a la secretaría

Contiene los datos del alumno: Numero de legajo, Apellido y Nombre, Carrera y cantidad de consultas.

Behavioral responses: Se lo puede derivar la consulta a un área de derivación para su resolución

Se le asigna un estado de consulta

Se le asigna una respuesta

El alumno lo evalúa

Se le asigna tiempo estimado de respuesta

Se le carga fecha de respuesta

Se le asignar tiempo para la implementación de la solución

Lel entry: fecha de contacto

Notion: Es el día, mes y año que el alumno realiza la consulta a la secretaría

Behavioral responses: Se le asigna a la consulta realizada por el alumno
Participa en la emisión de estadísticas
Participa en el calculo de tiempo de respuesta

Lel entry: fecha de derivación

Notion: Es el día, mes y año que la consulta es derivada a un área de derivación para su resolución

Behavioral responses: Se le asigna a la consulta realizada cuando se deriva a un sector
Participa en la emisión de estadísticas
Participa en el calculo de tiempo de respuesta

Lel entry: fecha de respuesta

Notion: Es la solución dada o la acción a tomar para resolver la consulta realizada

Behavioral responses: Se le asigna a la consulta realizada cuando el área de derivación envía la respuesta a la consulta

Lel entry: gravedad

Notion: es el grado de importancia que tiene la consulta para su tratamiento

Behavioral responses: Se le asigna a la consulta
Participa de las estadísticas de consulta

Lel entry: mail de derivación

Notion: Es una notificación que se hace de la consulta realizada al área de derivación

Behavioral responses: Es generado por la secretaría y enviado al área de derivación

Lel entry: número de legajo

Notion: Es un número correlativo, individual y único que se asigna a cada alumno que ingresa por primera vez y que haya completado su trámite de inscripción

Behavioral responses: Se asigna a la consulta para identificar quien la realiza

Lel entry: respuesta

Notion: Es la solución dada o la acción a tomar para resolver la consulta realizada por el alumno

Behavioral responses: Se le asigna a la consulta realizada una vez que se da la solución

Lel entry: tiempo de respuesta estimado

Notion: Es el tiempo que se asigna según el tipo de respuesta, es asignado por la secretaría

Behavioral responses: Se le asigna a la consulta en el momento que se registra y antes de ser derivada

Lel entry: tiempo de respuesta real / tiempo de respuesta real

Notion: Es el tiempo que se insumió en dar la respuesta por parte del área de derivación.

Se obtiene del lapso que existe entre la fecha de derivación y la fecha de respuesta

Behavioral responses: Se registra en la consulta en el momento de recibir la respuesta

Lel entry: tiempo de solución establecido

Notion: Se registra en la consulta en el momento de recibir la respuesta

Behavioral responses: Se asigna a la consulta para poder hacer un seguimiento de la situación

Lel entry: referencia

Notion: Es el tema por el cual se identifica a la consulta realizada por el alumno

Behavioral responses: Se le asigna a la consulta

Tipo Estado

Lel entry: consulta abierta

Notion: Este estado indica que la consulta fue ingresada y que no tiene una respuesta asignada, pero se encuentra dentro del plazo para asignar respuesta

Behavioral responses: Participa en el informe de consulta abiertas

Lel entry: consulta adeuda respuesta

Notion: Este estado indica que la consulta supera ya el tiempo asignado para la respuesta y que están atrasados con la elaboración de la respuesta

Behavioral responses: Participa en el informe de respuestas adeudadas

Lel entry: consulta cerrada

Notion: Este estado indica que la consulta fue resuelta

Behavioral responses: Se calcula el tiempo de respuesta real de la consulta cuando pasa a este estado

Lel entry: consulta cerrada no resuelta

Notion: Este estado se produce cuando un alumno hace una consulta que se da una respuesta pero que no soluciona el problema

Behavioral responses: Participa de la estadística de consultas cerradas que no tienen una respuesta implementada

Lel entry: consulta con respuesta asignada

Notion: Este estado indica que la consulta tiene una respuesta dada por el área de derivación y debe ser seguida para verificar la solución al problema

Behavioral responses: Participa en el informe de consulta con respuestas asignadas

Escenarios**Escenario Candidatos**

Alumno realiza la consulta
Alumno recibe la respuesta
Registrar consulta alumno
Asignar respuesta a consulta alumno
Emitir estadísticas de consultas
Evaluar respuesta dada a la consulta
Hacer seguimiento de consultas
Asignar tiempo para la implementación de la solución

Escenario Integrador

Scenario: Gestionar *consulta*

Goal: Realizar la gestión de las *consulta* realizadas por los *alumno*

Context: el *alumno* necesita información sobre algún tema

Resources:

Actors:

Episodes: **Registrar consulta alumno**

Asignar respuesta a consulta realizada

Evaluar respuesta dada a la consulta

Emitir estadísticas de consulta

Hacer seguimiento de consulta

Links def usr:

Escenarios

Scenario: Registrar consulta alumno

Goal: registrar la inquietud planteada por el alumno para poder dar una respuesta

Context: El alumno debe tener una consulta para realizar a la secretaría

Resources: PC para el ingreso de la consulta, número de legajo alumno

Actors: alumno

secretaría

área de derivación

Episodes: Se registra la consulta realizada por el alumno asignándole un número único.

Se le asigna referencia

Se le asignar gravedad

Se le asignar tiempo estimado de respuesta

Se registra la fecha de contacto.

Se asigna el estado de consulta abierta

Se genera el mail de derivación para que el área de derivación conozca la consulta y emita una respuesta.

Se registra la fecha de derivación

Links def usr:

Scenario: Asignar respuesta a consulta realizada

Goal: Registrar en el sistema la respuesta dada a la consulta del alumno

Context: El área de derivación analizó el problema y elaboró una respuesta

Resources: Pc para el ingreso de la respuesta

Solución dada a la consulta

Actors: área de derivación

Episodes: Se le asignar respuesta a la consulta recibida.

Se envía un mail de derivación a la secretaría para que haga el cierre de la consulta si considera que la respuesta dada es satisfactoria.

Se actualiza el estado de la consulta a consulta con respuesta asignada

Se carga fecha de respuesta

Se asignar tiempo para la implementación

Links def usr:

Scenario: Emitir estadísticas de consulta

Goal: Emitir listados estadísticos que permitan analizar la gestión de consulta de alumno

Context: Se deben tener las consulta registradas

Resources: PC
información de estado de consulta

Actors: secretaría

Episodes: Se cuentan cantidad de consulta por área de derivación en estado consulta abierta, consulta adeuda respuesta, consulta cerrada, consulta con respuesta asignada.

Links def usr:

Scenario: Evaluar respuesta dada a la consulta

Goal: Determinar si la respuesta es considerada satisfactoria

Context: El área de derivación envió el mail de derivación avisando que generó una respuesta.

Resources: Pc para evaluar la respuesta dada
Respuesta de consulta

Actors: secretaría
área de derivación

Episodes: Verifica la respuesta y si la respuesta es considerada satisfactoria actualiza el estado de la consulta a consulta cerrada y se calcula el tiempo de respuesta real y se establece el tiempo de solución establecido

Links def usr:

Scenario: Hacer seguimiento de consulta

Goal: Verificar cual es el estado de las consulta realizadas por los alumno

Context: Tener consulta ingresadas

Resources: Pc con el software de consulta
Respuesta de consulta

Actors: secretaría
área de derivación

Episodes: Se analiza el estado de la consulta y en el caso de estar demorada su respuesta se envía un mail de derivación al área de derivación reclamando una respuesta

Links def usr:

CRC s derivadas

A continuación se muestran las clases y las colaboraciones obtenidas.

CRC Card: alumno

Collaborations:
consulta

Responsabilities: Realizar la consulta

Recibir la respuesta

Actualizar Carrera.

Mostrar Carrera.

Actualizar Apellido y Nombre.

Mostrar Apellido y Nombre.

Actualizar Domicilio

Mostrar Domicilio

Actualizar Localidad

Mostrar Localidad

Actualizar Código Postal

Mostrar Código Postal

Actualizar Teléfono

Mostrar Teléfono

Validad Alumno

CRC Card: consulta

Collaborations: alumno

secretaría

área de derivación

respuesta

Responsabilities: derivar la consulta a un área de derivación para su resolución
asignar estado de consulta

asignar una respuesta

asignar tiempo estimado de respuesta

Actualizar código

Mostrar código

Actualizar Gravedad

Mostrar Gravedad

Mostrar tiempo estimado de respuesta

Mostrar Estado

Mostrar área de derivación de consulta

Actualizar fecha de derivación

Mostrar fecha de derivación

Mostrar respuestas

Actualizar Alumno

Mostrar Alumno

Actualizar Texto

Mostrar Texto

Mostrar Consulta

Crear Consulta

Actualizar Referencia

Mostrar Referencia

CRC Card: respuesta

Collaborations:

consulta

Responsabilities: asignar respuesta a la consulta realizada una vez que se da la solución

Actualizar Texto

Mostrar Texto

Actualizar Fecha de Creación

Mostrar Fecha de Creación

Actualizar Fecha de Envío

Mostrar Fecha de Envío

Actualizar Área de Derivación

Mostrar Área de Derivación

Mostrar respuesta

Actualizar Tiempo para la Implementación de la Solución

Mostrar Tiempo para la Implementación de la Solución

CRC Card: secretaría

Collaborations: consulta

Responsabilities: Registra consulta realizada por el alumno

Hace el seguimiento de la consulta

Evaluar respuesta

Genera respuesta

asignar tiempo para la implementación de la solución

realiza estadísticas de las consulta registradas

CRC Card: área de derivación

Collaborations:

consulta

Responsabilities: Generar respuesta

asignar tiempo para la implementación de la solución

Actualizar Nombre Área

Mostrar Nombre Área

Actualizar e-mail

Mostrar e-mail

Escenarios de la aplicación

Como etapa final se desarrollaron los escenarios de la implementación para poder observar la relación que existe entre los escenarios planteados para modelar el dominio. Esta actividad no es planteada por el proceso y no es necesario incluirla.

Se explican a continuación los criterios adoptados para representar estos escenarios.

Título : Se colocó el mismo nombre que en los escenarios del dominio con el agregado de implementación para distinguirlos.

Objetivos: Fueron planteados con lo que el sistema debe hacer para ese escenario.

Contexto: Qué situación se tiene que dar en el ambiente para que pueda desarrollarse el objetivo del escenario.

Recursos: Se plantearon los objetos que deben interactuar en el sistema para obtener los resultados. El caso de solo mostrar datos de los objetos relacionados, no fue incluido en este punto.

Por ejemplo en el escenario “Derivar consulta”, se ponen como recursos Consulta, Área de derivación y Estado de Consulta. Los demás datos que se muestran se obtienen por las colaboraciones entre Consulta y los otros objetos (Ej: Alumno)

Actores: En este punto mencionan quien interactúa con el sistema y quien recibe información del sistema.

Episodios: Se describen las situaciones que producen los actores y las acciones que debe realizar el sistema en consecuencia.

Las acciones que debe realizar el sistema están escritas con los verbos en infinitivo.

El resto son las acciones que realiza el actor (usuario del sistema)

Se puede observar que en los escenarios que modelan el dominio, se tiene un mayor grado de abstracción. Esto cumple con el objetivo de entender el dominio de la aplicación y poder utilizarse como medio de comunicación entre las diferentes etapas del proceso de desarrollo.

Cuando realizamos los escenarios de la implementación el nivel de detalle es mayor y se observan las actividades que debe desarrollar el sistema para cumplir con los requerimientos planteados por el usuario.

Esto se puede observar en el escenario planteado como “Registrar consulta alumno” y “Registrar consulta alumno implementación”

En el escenario “Registrar consulta alumno” se pueden ver las actividades que se realizan actualmente para registrar las consultas realizadas y en el mismo escenario planteado para la implementación se observa paso por paso lo que debe hacer el sistema para cumplir con ese objetivo.

En algunos es conveniente desdoblar los escenarios en dos, debido a que al ser automatizados existen diferencias en el tiempo en el que se realiza la actividad y no permitiría registrar una consulta hasta que se tenga el área de derivación definida.

En la misma situación planteada en el sistema actual, no se registraba inconveniente por realizarse de manera manual, con lo cual es razonable que no se haya reflejado en los escenarios planteados.

Scenario: Registrar consulta alumno Implementación

Goal: Registrar la consulta del alumno en el sistema para poder derivarla a un área de derivación

Context: El alumno realizó una consulta a la secretaría

Resources: alumno
área de derivación
gravedad de consulta
estado de consulta

Actors: secretaría

Episodes: Solicitar LU

Validar alumno

Si existe Mostrar Alumno

No existe Mostrar mensaje de Error

Solicitar ingreso del texto de la consulta

Mostrar posibilidades de gravedad

Mostrar tiempo de respuesta establecido para la gravedad seleccionada

Registrar consulta

Derivar consulta

Links def usr:

Scenario: Derivar consulta implementación

Goal: Registrar en el sistema el área de derivación a la que se asigna la consulta para que responda

Context: Se evaluó quien debe responder por esa consulta

Resources: Consulta
área de derivación

Actors: Secretaría
Área de derivación

Episodes: Se selecciona listado de consulta sin asignar

Se selecciona consulta para derivar

Mostrar consulta

Solicitar se asigne área de derivación

Actualizar estado de consulta a consulta asignada

Actualizar área de derivación

Enviar mail de derivación

Links def usr:

Scenario: Asignar respuesta a consulta realizada implementación

Goal: Registrar en el sistema la consulta dada a la consulta del alumno

Context: El área de derivación analizó el problema y elaboró una respuesta

Resources: Consulta

Respuesta

Actors: Área de derivación

Episodes: Selecciona consulta por código

Buscar consulta

si no encuentra Mostrar mensaje de error

si encuentra Mostrar consulta

Selecciona responder

Mostrar ventana respuesta

Mostrar texto consulta

Ingresar respuesta

Actualizar tiempo de solución establecido

Selecciona responder

Enviar mail de derivación

Actualizar estado de consulta a respuesta asignada

Actualizar fecha de respuesta con el día de la fecha

Links def usr:

Scenario: Evaluar Respuesta dada a la consulta implementación

Goal: Registrar si la respuesta es considerada satisfactoria por la secretaría

Context: La secretaría recibió el mail de derivación avisando que se generó una respuesta

Resources: mail de derivación

Consulta

Respuesta

Actors: secretaría

Episodes: Selecciona consulta por código de consulta

Mostrar consulta

Selecciona respuesta

Mostrar respuesta

Verifica si la respuesta es satisfactoria

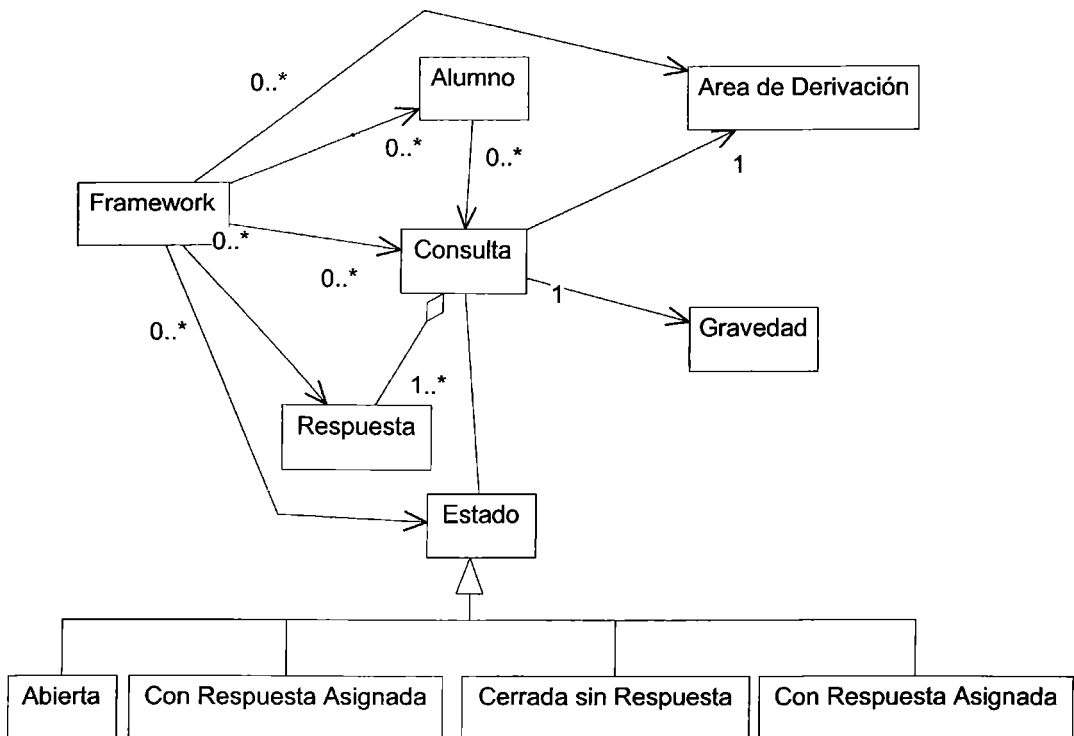
Si es satisfactoria Actualizar estado de consulta a consulta cerrada

calcular tiempo de respuesta real

registrar tiempo de solución establecido

Links def usr:

Diagrama de Clases de la aplicación



Interfaces del Sistema

Se exponen a modo de ejemplo algunas de las interfaces del sistema del sistema desarrollado a partir de los requerimientos establecidos. Para mostrar la relación entre los escenarios y el producto se insertaron las interfaces en el desarrollo del escenarios.

Scenario: Registrar *consulta alumno* Implementación

Goal: Registrar la *consulta* del *alumno* en el sistema para poder derivarla a un *área de derivación*

Context: El *alumno* realizó una *consulta* a la *secretaría*

Resources: *alumno*
área de derivación
gravedad de *consulta*
estado de consulta

Actors: *secretaría*

Episodes: Solicitar *LU*

Consultas

Consulta : SinCodigo

Alumno

L.U.

Nombre

Apellido

E-Mail

Domicilio

Localidad CP

Telefono

Gravedad

Gravedad

Descripcion

Tiempo estimado Dias

Derivacion

Area

Fecha Cierre

Historia

Estado

Respuestas

Area	Creacion	Envio
------	----------	-------

Validar *alumno*

Si existe *Mostrar Alumno*

Consultas

Consulta : SinCodigo

Alumno

L.U.

Nombre

Apellido

E-Mail

Domicilio

Localidad CP

Telefono

Gravedad

Gravedad

Descripcion

Tiempo estimado Dias

Derivacion

Area

Fecha Cierre

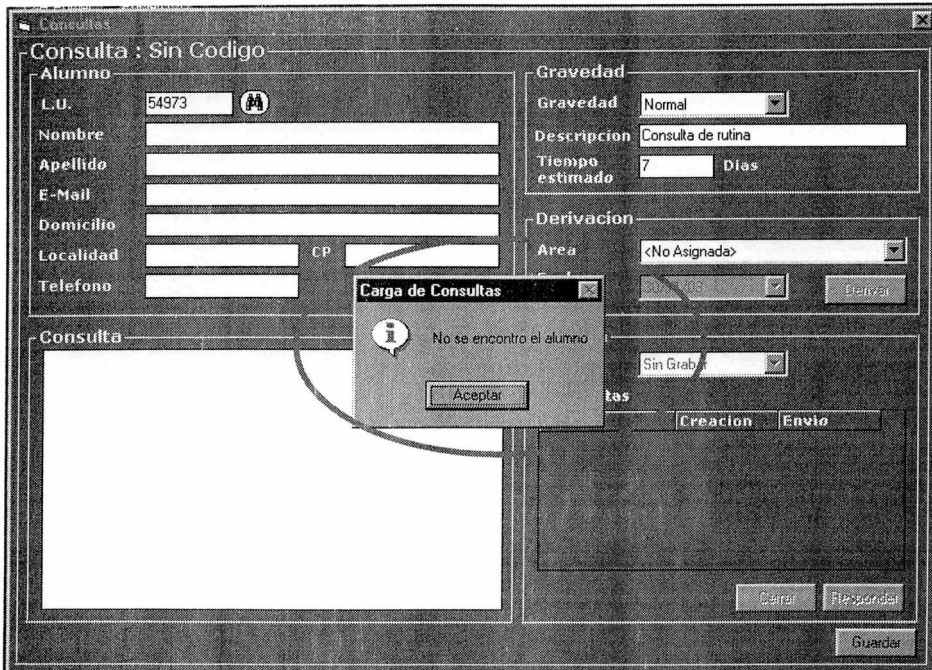
Historia

Estado

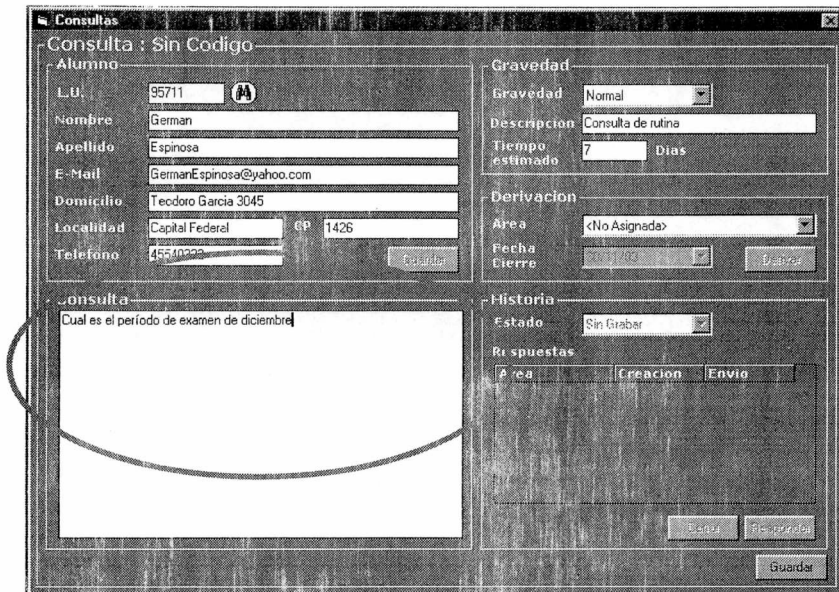
Respuestas

Area	Creacion	Envio
------	----------	-------

No existe Mostrar mensaje de Error



Solicitar ingreso del texto de la consulta



Mostrar posibilidades de *gravedad*

The screenshot shows a window titled 'Consultas' with a sub-header 'Consulta : SinCodigo'. It contains several sections:

- Alumno:** Fields for L.U. (95711), Nombre (German), Apellido (Espinosa), E-Mail (GermanEspinosa@yahoo.com), Domicilio (Teodoro Garcia 3045), Localidad (Capital Federal), C.P. (1426), and Telefono (45540322). A 'Guardar' button is present.
- Gravedad:** A dropdown menu is open, showing options: 'Normal' (selected), 'Muy Alta', 'Alta', 'Normal', 'Baja', and 'Muy Baja'.
- Derivacion:** Fields for Area (<No Asignada>), Fecha Cierre (30/11/09), and a 'Derivar' button.
- Consulta:** A text area containing the question 'Cual es el periodo de examen de diciembre'.
- Historia:** A dropdown menu for 'Estado' set to 'Sin Grabar'.
- Respuestas:** A table with columns 'Area', 'Creacion', and 'Envio'. Below it are 'Cerrar' and 'Responder' buttons.
- A 'Guardar' button is located at the bottom right of the window.

Mostrar tiempo de *respuesta* establecido para la *gravedad* seleccionada

This screenshot shows the same 'Consultas' window, but with the 'Gravedad' dropdown menu closed and the 'Tiempo estimado' field updated to '7 Dias'.

- Alumno:** Same data as the previous screenshot.
- Gravedad:** The dropdown is closed, showing 'Normal'.
- Derivacion:** Same data as the previous screenshot.
- Consulta:** Same text area.
- Historia:** Same dropdown menu.
- Respuestas:** Same table and buttons.
- The 'Tiempo estimado' field now displays '7 Dias'.

Registrar consulta

Consultas : Sin Código

Alumno

L.U.: 95711 (M)

Nombre: German

Apellido: Espinosa

E-Mail: GermanEspinosa@yahoo.com

Domicilio: Teodoro Garcia 3045

Localidad: Capital Federal CP: 1426

Telefono: 45540322

Gravedad

Gravedad: Normal

Descripcion: Consulta de rutina

Tiempo estimado: 7 Dias

Derivación

Area: <No Asignada>

Fecha Cierre: [Fecha]

Historia

Estado: Sin Grabar

Respuestas

Area	Creacion	Envio

Guardar

Derivar consulta implementación

Links def usr:

Scenario: Derivar consulta implementaciónGoal: Registrar en el sistema el área de derivación a la que se asigna la consulta para que respondaContext: Se evaluó quien debe responder por esa consultaResources: Consulta
área de derivaciónActors: Secretaría
Área de derivaciónEpisodes: Se selecciona listado de consulta sin asignar

Solicitar se asigne área de derivación

Consultas

Consulta : 20031123KTBI

Alumno

L.U. 95711

Nombre German

Apellido Espinosa

E-Mail GermanEspinosa@yahoo.com

Domicilio Teodoro Garcia 3045

Localidad Capital Federal CP 1426

Telefono 45540322

Gravedad

Gravedad Normal

Descripcion Consulta de rutina

Tiempo estimado 7 Dias

Derivacion

Area ALUMNOS

Fecha Cierre <No Asignada>
BIBLIOTECA
ALUMNOS
SECRETARIA

Historia

Estado Asignada

Respuestas

Area	Creacion	Envio
------	----------	-------

Consulta

Cual es el período de examen de diciembre

Consultas

Consulta : 20031123KTBI

Alumno

L.U. 95711

Nombre German

Apellido Espinosa

E-Mail GermanEspinosa@yahoo.com

Domicilio Teodoro Garcia 3045

Localidad Capital Federal CP 1426

Telefono 45540322

Gravedad

Gravedad Normal

Descripcion Consulta de rutina

Tiempo estimado 7 Dias

Derivacion

Area ALUMNOS

Fecha Cierre 2003/03

Historia

Estado Asignada

Respuestas

Area	Creacion	Envio
------	----------	-------

Consulta

Cual es el período de examen de diciembre

Actualizar estado de consulta a consulta asignada

Actualizar área de derivación

Enviar mail de derivación

Links def usr:

<h2>Referencias</h2>	
[Antonelli 99]	Antonelli, L. Oliveros, A Rossi G., "Baseline Mentor, An application that Derives CRC Cards from Lexicon and Scenario" XXVIII JAIO, II Workshop Iberoamericano en Ingeniería de Requerimientos, WER'99, Buenos Aires, Argentina, Septiembre 9 y 10, 1999
[Basili]	http://www.cs.umd.edu/users/mvz/mswe609/book/chapter1.pdf
[Beck 00]	Beck K "Extreme programming explained: embrace change" Addison Wesley 2000.
[Beck 89]	Beck K., Cunningham W. "A Laboratory for teaching Object-Oriented thinking" From the OOPSLA '89 Conference Proceedings October 1-6 1989. pp 1-6
[Booch91]	Booch, G., " <i>Object Oriented Design with Applications</i> ". The Benjamin Cumming Publishing Company, Inc., Redwood City, 1991.
[Booch94]	Booch, G., "Scenarios", <i>Report on Object Analysis and Design</i> , Vol.1, N° 3, 1994, pp. 3-6.
[Brooks 87]	by Frederick P. Brooks, Jr., "No Silver Bullet - Essence and Accidents of Software Engineering" Computer Magazine; April 1987 University of North Carolina at Chapel Hill
[Carroll195]	Carroll, J., " <i>Scenario-Based Design: Envisioning Work and Technology in System Development</i> ". Introduction: The Scenario Perspective on System Development , J. Carroll, ed., John Wiley & Sons, New York, 1995.
[Cockburn 99]	Cockburn Alistar "Responsibility - based Modeling" Technical Memo Hat TR-99.02 http://members.aol.com/humansandt/techniques/responsibilities.htm
[Cockburn95]	Cockburn, A., "Structuring Use Cases with Goals", Technical Report, Humans and Technology, 7691 Dell Rd, Salt Lake City, UT 84121, HaT.tr.95.01, http://members.aol.com/acockburn/papers/usecases.htm , 1995.
[Constantine98]	Constantine, L., "Joint Essential Modeling. User Requirements Modeling for Usability", International Conference on Requirements Engineering, Colorado Springs, Constantine & Lockwood, Ltd., 1998.
[Chen76]	Chen, P., "The entity-relationship model: Towards a unified view of data", <i>ACM Transactions on Database Systems</i> , 1(1), 1976.
[Chidamber 94]	Chidamber, Shyam. Kemerer, Chris. "A Metrics Suite for Object Oriented Design", IEEE Trans. On Software Engineering June 1994 pp476-493.
[Dano97]	Dano, B., Briand, H., Barbier, F., "An Approach Based on the Concept of Use Case to Produce Dynamic Object-Oriented Specification", Proceedings of the Third IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, Los Alamitos, California, 1997, pp.54-64.
[Davis 93]	Alam, M Davis, "Software Requirements: Objects, Functions and

	States, Second Edition”, Prentice Hall PTR, Marzo 1993
[Diaz 97]	Diaz, O. Iturrioz, J. Piattine, M. “Promoting business policies in object oriented methods” Sesión trabajos ya publicados: publicado en The Journal of Systems and Software, 1998. Actas II Jornadas de Ingeniería de Software JIS97, Dpto de Informática, Universidad del país Vasco, San Sebastián España. 1997, pp. 384-400.
[Doom98]	Doom, J., Kaplan, G., Hadad, G., Leite, J.C.S.P., “Inspección de Escenarios”, Proceedings of WER’98, Workshop en Engenharia do Requisitos, Maringá, Brazil, 1998, pp. 57-69.
[Firesmith94]	Firesmith, D. G, “Modeling the Dynamic Behavior of Systems, Mechanisms, and Classes with Scenarios”, <i>Report on Object Analysis and Design</i> , Vol.1, N° 2, 1994, pp. 32-36.
[Gamma 94]	Gamma E., Helm R., Johnson R., Vissides J. “Design Patterns. Elements of Reusable Object Oriented Software” Addison Wesley 1994.
[Goguen92].	Goguen, J.A., Linde, Ch., “Techniques for Requirements Elicitation”, Proceedings of the International Symposium on Requirements Engineering, IEEE Computer Society, 1992, pp. 152-164.
[Gough95]	Gough, P. A., Fodemski, F. T., Higgins, S. A., Ray, S. J., “Scenarios - An Industrial Case Study and Hypermedia Enhancements”, RE95: Proceedings of the International Symposium on Requirements Engineering, IEEE Computer Society Press, Los Alamitos, California, 1995, pp. 10-17.
[Hadad98]	Hadad, G., Kaplan, G., Leite, J.C.S.P., “Léxico extendido del lenguaje y escenarios del meeting scheduler”. Technical Report # 13, Dto. Investigación, U. Belgrano, Bs. As., 1998.
[Hadad99]	Hadad, G., Doorn, J., Kaplan, G., Leite, J.C.S.P., “Enfoque Middle-Out en la Construcción e Integración de Escenarios”. Proceedings de WER’99, Buenos Aires, Argentina, 1999, pp. 79-94.
[Hoydalsvik 93]	Geir Magne Hoydalsvik and Gttom Sindre “On the purpose of Object-Oriented Analysis” OOPSLA’93 Proceedings. Conference on Object-Oriented Information system (OOIS’98), 9-11 september 1998 at the University of La Sorbonne, Paris, France.
[Hsia94]	Hsia, P., Samuel, J., Gao, J., Kung, D., Toyosima, I. and Chen, C., “Formal Approach to Scenario Analysis”, <i>IEEE Software</i> , 1994, V.11, N° 2, pp. 33-41.
[Jacobson 95]	Jacobson I. Ericsson M(contributor). “The Object Advantage: Business Process Reengineering With Object Technology” Addison-Wesley Object Technology Series. 1995.
[Jacobson92]	Jacobson, I., Christerson, M., Jonsson, P., Overgaard, G., “Object-Oriented Software Engineering - A Use Case Driven Approach”, Reading, MA: Addison Wesley, New York: ACM Press, 1992.

[Jacobson94a]	Jacobson, I., "Basic Use-Case Modeling", <i>Report on Object Analysis and Design</i> , Vol.1, N° 2, 1994, pp. 15-19.
[Jacobson94b]	Jacobson, I., "Basic Use-Case Modeling (Continued)", <i>Report on Object Analysis and Design</i> , Vol.1, N° 3, 1994, pp. 7-9.
[Kendall 97]	Kenneth E. Kendall & Julie E. Kendall – Análisis y diseño de sistemas – 3ra edición. México 1997
[Leite00]	Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H., Kaplan, G.N., "A Scenario Construction Process", <i>Requirements Engineering Journal</i> , Vol.5, N° 1, 2000, pp. 38-61.
[Leite93]	Leite, J.C.S.P., Franco, A.P.M., "A Strategy for Conceptual Model Acquisition", IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, 1993, pp. 243-246.
[Leite96]	Leite, J.C.S.P., Oliveros, A., Rossi, G., Balaguer, F., Hadad, G., Kaplan, G., Maiorana, V., "Léxico extendido del lenguaje y escenarios del sistema nacional para la obtención de pasaportes", Technical Report #7, Departamento de Investigación, Universidad de Belgrano, Buenos Aires, 1996.
[Leite97a]	Leite, J.C.S.P., "Ingeniería de Requisitos". Notas de Cátedra, 1997.
[Leite97b]	Leite, J.C.S.P., Rossi, G., Balaguer, F., Maiorana, V., Kaplan, G., Hadad, G., Oliveros, A., "Enhancing a Requirements Baseline with Scenarios", <i>Requirements Engineering Journal</i> , Vol.2, No. 4, 1997, pp. 184-198.
[Lonardi 01]	Lonardi, Maria del Carmen. "Una estrategia de modelado conceptual de Objetos basadas en Modelos de Requisitos en Lenguaje Natural" UNLP – Noviembre 2001
[Lorenz 94]	Lorenz, Mark. Kidd, Jeff. "Object-Oriented Software Metrics: A Practical Guide", Prentice Hall 1994.
[Loucopoulos 95]	Loucopoulos, P., Karakostas, V., "System Requirements Engineering", McGraw-Hill, London, 1995.
[Martin 00]	Martin R. "Design Principles and Design Patterns" http://www.objectmentor.com/publications
[Meyer'97]	Meyer Bertrand "Object Oriented Construction" Prentice Hall Object oriented Series. 1997
[Ober98]	Oberg, R., Probasco, L, Ericsson, M., "Applying Requirements Management with Use Cases". Rational Software Corporation, 1998.
[Oliveros 01]	Oliveros, A. Apuntes de Cátedra. Maestría en ingeniería de software UNLP - 2001
[Olson 93]	Olson, T. Parker Gates, L. Mullaney, J. Over, J. Reizer, N. Kellner, M. Phillips, R. DiGennaro, S. Special Report CMU/SEI-93-SR-007 Carnegie Mellon – Software Engineering Institute. Junio 1993
[Pfleeger 02]	Shari Lawrence Pfleeger – Ingeniería de Software (Teoría y Práctica)

	Argentina 2002
[Potts94]	Potts, C., Takahashi, K., Antón, A. I., "Inquiry-Based Requirements Analysis", <i>IEEE Software</i> , Vol. 11, N° 2, 1994, pp. 21-32.
[Ridao 00]	Ridao, M., Doorn, J., Leite, J.C.S.P., "Uso de Patrones en la Construcción de Escenarios", Anais de WER 2000, Río de Janeiro, Brazil, 2000, pp. 140-157.
[Robertson95]	Robertson, S.P., "Generating Object-Oriented Design Representations via Scenario Queries, <i>Scenario-Based Design: Envisioning Work and Technology in System Development</i> ". J. Carroll, ed., John Wiley & Sons, New York, 1995.
[Schneider98]	Schneider, G., Winters, J., "Applying Use Cases, <i>A Practical Guide</i> ", Addison-Wesley, 1998.
[Somerville 02]	Sommerville, I., Ingeniería de software, 6ta Edición Mexico 2002
[Standish 94]	The Standish Group 1994. The CAHOS report . Dennis, Ma.
[Standish 95]	The Standish Group 1995, The scope of software development Project Failures. Dennis, Ma.
[Stucliffe 97b]	Sutcliffe, A., "A Technique Combination Approach to Requirements Engineering", Proceedings of the Third IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, Los Alamitos, Colorado, 1997, pp. 65-74.
[Weidenhaupt98]	Weidenhaupt, K., Pohl, K., Jarke, M., Haumer, P., "Scenarios in System Development: Current Practice", <i>IEEE Software</i> , 1998, pp. 34-45.
[Wirfs-Brock 95] [Wirfs 95]	Wirfs-Brock, R., "Designing Objects and Their Interactions: A Brief Look at Responsibility-Driven Design", <i>Scenario-Based Design: Envisioning Work and Technology in System Development</i> , J. Carroll, ed., John Wiley & Sons, New York, 1995, pp. 337-359.
[Wirfs-Brock R 90]	Wirfs-Brock Rebecca B, Wilkerson, L. Wiener "Designing Object Oriented Software", Prentice Hall, 1990
[Zorman95]	Zorman, L., "Requirements Envisaging by Utilizing Scenarios (Pebus)", Ph.D. Dissertation, University of Southern California, 1995.