

## **4. ESCENARIOS**

### **4.1. Introducción.**

Un escenario es una descripción parcial del comportamiento de la aplicación en un momento específico. La utilización de escenarios implica identificar distintas situaciones y describir la acción a llevar a cabo. Los mismos son de gran ayuda en el momento de especificar requerimientos; y su rol principal es el de permitir la comunicación entre expertos de software y del dominio, y analizar aspectos específicos de un sistema, describiéndolo en forma concreta. La ventaja de los escenarios sobre cualquier otro método de elicitación de requerimientos, es que los escenarios guardan una gran similitud a la forma en que los seres humanos entienden y describen los problemas.

Los escenarios describen actores, objetivos y episodios. Un actor no necesariamente es una persona o agente físico, un actor representa un rol dentro del sistema, por lo tanto, los actores son las entidades que hacen uso del sistema para satisfacer cierta necesidad, estas necesidades son los objetivos, que representan las condiciones a ser alcanzadas. Los objetivos están representados por episodios. Un episodio es un conjunto de acciones asignadas a determinados actores. Están formados por un conjunto de oraciones en concordancia a un lenguaje natural simple que hace posible la descripción operacional del comportamiento, las cuales involucran la actividad de alguna función del sistema.

Varios autores han estudiado esta técnica como por ejemplo [Potts], [Jacobson], [Carrol], [Zorman]. Esto lleva aparejado que el formato en el cual se generan es muy variado. Pueden ser hechos en lenguaje natural, como los uses cases en [Jacobson], pueden ser storyboards, una descripción gráfica en donde las acciones se identifican con distintos cuadros como si fuera una historieta, o pueden ser diagramas de interacción entre objetos.

Si bien cada escenario describe una situación particular, ninguno es totalmente independiente del resto. Cada uno de ellos tiene una relación semántica con otros escenarios [Booch].

Es importante considerar que el nivel de detalle con el que se describen los escenarios depende de dos factores:

- el grado de importancia que el cliente le otorgue a los hechos específicos del problema;
- la fase en la que se encuentra el proceso de desarrollo.

Los escenarios cumplen distintos objetivos, dependiendo de la fase en que se describen durante el ciclo de desarrollo del software. En la fase de producción de requerimientos, los objetivos de los escenarios son:

- capturar los requerimientos;
- proveer un medio de comunicación entre los clientes y los ingenieros de software;
- contar con un instrumento de traceability.

## **4.2. Modelos existentes.**

A continuación se analizan algunos modelos de escenarios propuestos por diferentes autores.

### **4.2.1 Uses Cases de Jacobson**

Para ver las similitudes entre escenarios y use cases, analicemos el planteo de Jacobson de los use cases, En el cual se observa que el modelo de use cases define el comportamiento del sistema (una de las dos componentes fundamentales de los escenarios). Se describe el entorno del sistema (situación como se llama en los escenarios) a través de los distintos usuarios que operan el sistema a través de los use cases.

El modelo de use cases es un grafo con dos tipos de nodos, nodos actores y nodos use cases, y con un nombre, el nombre del sistema. Cada nodo actor tiene un nombre y una clase. Los nombres de los nodos actores son únicos. Por su parte, cada nodo use case tiene también un nombre y una clase. Los nombres de los nodos use cases, también son únicos.

Un nodo actor tiene al menos un arco hacia un nodo use case, y un nodo use case tiene al menos un arco hacia un nodo actor. Estos arcos se denominan arcos de comunicación.

Una instancia de un actor puede crear instancias de use cases, y una instancia de use cases obedece a su clase. Un arco de comunicación entre un nodo actor y un nodo use case significa que un estímulo ha sido enviado entre una instancia de la clase actor y una de la clase use case o entre instancias de la clase use case.

Los actores son objetos que residen fuera del modelo del sistema. Representan todo lo que necesita intercambiar información con el sistema. Nada más fuera del sistema tiene algún impacto en él. Los actores pueden ser humanos u otro sistema.

Se hace una distinción entre actores y usuarios. Un usuario es un humano que usa el sistema, en cambio, un actor representa el rol específico que

un usuario puede jugar. Los actores son instancias de una clase, y los usuarios son algún tipo de recursos que implementan estas instancias. El mismo usuario puede así actuar como instancias de diferentes actores.

#### **4.2.2. OBA**

Se propone la descripción de escenarios a través de scripts. Un script es una descripción estructurada de un uso típico del sistema. Se forman realizando un contrato entre dos roles. El primer rol, iniciador, colabora con el segundo participante para realizar un paso de la tarea completa. El iniciador realiza una acción, responsabilidad y el participante responde con otra acción, el servicio correspondiente.

Cada script contiene: nombre, autor, versión, precondition (estado del sistema para que ese script suceda), postcondición (estado final del sistema al finalizar el script), trace (área de actividad del dominio a la que pertenece ese script).

Estos scripts surgen en la etapa de análisis de requerimientos para capturar la funcionalidad del sistema. En la etapa de diseño se utilizan los scripts para encontrar los objetos del sistema, sus responsabilidades y colaboraciones con el resto de los objetos. Finalmente se usan las pre y post condiciones para determinar el ciclo de vida completo de un objeto.

#### **4.2.3. ICM, una nueva propuesta [Potts 95]**

En el trabajo de Potts 95, se especifica en forma más detallada el concepto de escenarios, se identifican partes que los componen y se presentan estrategias para definirlos.

Un escenario es una descripción narrativa de un uso concreto del sistema. Describe la ejecución de una parte de la funcionalidad del mismo. Los escenarios tienen actores con objetivos (condiciones a ser alcanzadas). Estos objetivos pueden no alcanzarse por determinados obstáculos. Estos obstáculos pueden ser condiciones del sistema o porque se ha alcanzado algún otro objetivo (conflictivo). Los objetivos y obstáculos están representados por episodios. Un episodio es un conjunto de acciones asignadas a determinados actores. Un actor no necesariamente es una persona o agente físico, un actor representa un rol dentro del sistema.

El esquema propuesto de escenarios es el siguiente:

*Escenario: nombre del escenario*

*Settings:*

*Background: información, estado o situación del sistema que indica los objetivos de los actores.*

*Roles(actores): Cada uno de los actores intervinientes.*

*Narrativa: Conjunto de episodios: Cada episodio está descrito por: objetivos, obstáculos (opcional), acciones y logros.*

El hecho de considerar obstáculos, fuerza a los diseñadores a pensar en soluciones flexibles y robustas para situaciones del sistema no idealizadas (por ejemplo: errores cometidos por el usuario o usos del sistema en una forma no prevista durante el análisis y diseño).

#### **4.2.4. El modelo de Carroll**

Este autor proviene del área hci<sup>5</sup>, motivo por el cual considera un escenario como el uso (real o imaginario) que tiene el usuario de un sistema, analizando el comportamiento deseado o no deseado del individuo frente al mismo.

Un escenario describe en forma textual una situación particular de un usuario interactuando con el sistema. Esta información sirve como fuente para el diseño del sistema. A través del escenario se puede observar que es lo que hace el usuario con el sistema, como interactúa, como reacciona ante las respuestas y que problemas tiene. El conjunto de escenarios permite razonar sobre el comportamiento de los usuarios ante determinadas situaciones del sistema.

La creación de escenarios no puede basarse en la simple observación del sistema. A veces pueden necesitarse escenarios de sistemas que aún no existen o de los cuales no se conoce su uso. Esto se puede hacer por analogía con sistemas ya existentes (el nuevo sistema puede ser similar a otro ya existente o ser descendiente de otro sistema).

Si bien se menciona el uso de escenarios para la etapa de adquisición de requerimientos iniciales, el trabajo está enfocado en el uso de escenarios para diseño. El conjunto de escenarios muestra como actúan los usuarios ante determinadas situaciones, y sobre esta base se pueden discutir alternativas de diseño. Propone el diseño racional, a partir de un escenario analizar las relaciones causales del mismo: ante determinada situación se puede desencadenar una reacción favorable o no en el usuario. Cada situación del escenario debe ser analizada de la siguiente forma:

*En <situación> <una expresión> causa <consecuencias deseables> pero puede causar <consecuencias indeseables>*

A partir de aquí se pueden analizar escenarios alternativos en donde distintas condiciones del sistema intentan obviar las consecuencias no deseadas.

#### **4.2.5. La propuesta de Booch**

Según Booch, los escenarios cumplen tres principios fundamentales. En primer lugar los escenarios son una parte esencial para capturar los requerimientos. Los escenarios hablan el lenguaje del usuario final y del experto del dominio, por lo tanto proveen un medio para que ellos expliquen sus expectativas sobre el comportamiento del sistema. Segundo, los escenarios

---

<sup>5</sup>Human Computer Interfase

proveen un vehículo de comunicación. Llevan al usuario final y experto del dominio al nivel del problema, exigiendo al desarrollador a adquirir el dominio del problema, forzándolo a considerar una distribución inteligente de responsabilidades dentro del sistema. Y tercero, a medida que el proyecto avanza, los escenarios sirven como instrucciones tanto a los desarrolladores individuales, como al equipo de pruebas.

¿Qué es un escenario? Un escenario provee un esbozo del comportamiento del sistema. Los escenarios documentan decisiones de requerimientos o diseño, proveen un punto de comunicación sobre la semántica del sistema y pueden servir como punto de partida para la implementación detallada.

Luego, el comportamiento del sistema de software puede ser capturado a través de una red de escenarios de la misma forma que lo hace los storyboards con respecto a una película. Aunque, la analogía entre el proceso de desarrollo de soft y filmar una película se quiebra en el punto en que las películas tienen sólo un camino de acción y pueden ser representadas por un largo y continuo storyboard. En cambio, todas las aplicaciones de soft interesantes, raramente tienen un camino simple de comportamiento.

Booch utiliza distintos métodos para representar escenarios. En primer lugar usa tarjetas CRC. El probó que son una buena manera de abordar la construcción de escenarios. Su mayor atractivo como técnica de desarrollo es que son totalmente libres. Lamentablemente, las tarjetas CRC padecen de una gran limitación: no pueden considerar aspectos temporales de un escenario.

#### **4.2.5. EL enfoque de Escenarios de Leite.**

La metodología de Leite [Leite 95] es a la que se adhiere este trabajo de tesis, la misma propone trabajar con documentos en lenguaje natural altamente vinculados entre sí, centrados en una estructura orientada al cliente llamada Requirements Baseline, siendo está un metamodelo que contiene descripciones sobre el dominio del problema y el artefacto de software que ha de ser construido dentro de ese dominio. Entre los documentos que incorpora esta metodología, se encuentran los escenarios, cuyo principal objetivo durante la etapa de elicitación de requisitos, es comprender el problema en su totalidad. En las siguientes etapas acompaña el proceso de desarrollo del software describiendo aspectos de diseño, codificación y la representación de modificaciones a requisitos existentes y la representación de requisitos nuevos.

Los escenarios tienen una estructura compuesta por el Título que lo identifica, el Objetivo a lograr en el macrosistema, el Contexto que describe la ubicación geográfica y temporal del escenario, así como un estado inicial o precondition, también se especifican los Recursos necesarios que estén disponible, los Actores que tienen un rol en el escenario y por ultimo los Episodios que son una serie ordenada de sentencias escritas en lenguaje natural.

El Objetivo, el Contexto, los Recursos y los Actores son sentencias declarativas, mientras que los Episodios son un conjunto de sentencias con un lenguaje muy simple que hace posible la descripción operativa de comportamientos.

Un episodio puede concebirse como un escenario en sí mismo, esto posibilita la descomposición de un escenario en subescenarios.

En estudios anteriores [Hadad 97] se han propuesto heurísticas para derivar los escenarios desde el LEL. Estas heurísticas permiten generar una primera versión de los escenarios. Se detectan los actores en el LEL y se utilizan sus impactos y sus vínculos para describir parcialmente los escenarios. Luego, se completan los escenarios validándolos con el usuario y ampliando la información faltante o confusa.

#### **4.2.5. Comparación de los distintos modelos.**

Como los enfoques basados en escenarios atraen un interés creciente entre los ingenieros de requerimientos, la literatura sobre métodos, modelo y notaciones de escenarios prolifera. En función de las definiciones vistas y teniendo en cuenta la metodología a la que adhiere esta tesis (ver cuadro 5), podemos considerar que una de las más coincidentes es el modelo ICM. En él, podemos ver que se tienen en cuenta los actores, sus objetivos, las acciones que realizan para alcanzarlos y los posibles obstáculos que impiden alcanzar esos objetivos. Un elemento que aparece en casi todos los trabajos, son las pre y post condiciones. Las mismas, dan una idea de estado inicial y final de un escenario como se hace en el modelo de OBA. Los distintos trabajos, si bien lo denominan de otra forma, también tienen presente esta idea.

Con respecto a los actores, si bien en todas las representaciones se mencionan, en el único trabajo en que se los identifican como entidades externas del sistema es en el modelo de use cases. Esto significa que es necesario conocer los límites del sistema, los que en realidad se están tratando de definir. No siempre es fácil identificar en un primer momento, qué entidades son externas y cuáles internas al sistema.

La propuesta de Booch y los use cases destacan los beneficios de crear un sistema de descripción concreta orientada al uso, antes de modelar la función, los datos y el comportamiento. Las extensiones y alternativas propuestas incluyen el agregado de estructura a los use cases; el tratamiento formal de escenarios y el uso de escenarios durante la documentación, discusión y evolución de los requerimientos.

Los escenarios también son populares en otros campos, como en la interacción hombre-computadora [Carroll] y planeamiento estratégico.

	Objetivo	Contexto	Recursos	Actores	Episodios
Use Case	Sí	No	No	Sí	No
OBA	Sí	Sí	No	Sí	No
ICM	Sí	Sí	No	Sí	Sí
Carroll	Sí	No	No	Sí	Sí
Booch	Sí	No	No	No	Sí

*Cuadro 6. Comparación de distintos modelos de Escenarios.*

#### 4.2.6. Uso de Escenarios en la Practica

Mientras los escenarios permiten un aprendizaje interdisciplinario en ingeniería de requerimientos, ellos también sirven como medio para divisiones de trabajo, con consecuencias significantes para el proyecto de administración e integración del artefacto de software. De la investigación que se informa en el artículo Scenarios in System Development: Current Practice publicado en [IEEE 98], se puede resumir algunas de las observaciones a las que llegaron luego de evaluar 15 proyectos desarrollados en 4 países europeos.

- ◆ **Uso de escenarios cuando falla el modelado abstracto**

Cuando algunos proyectos fracasaron al desarrollar los modelos abstractos conceptuales como los modelos de clase, debido a la complejidad del dominio del problema, los desarrolladores utilizaron escenarios para elicitación y documentar los requerimientos del cliente, satisfactoriamente en ambos casos.

- ◆ **Los Escenarios requieren la coexistencia de prototipos.**

En las dos terceras partes de los proyectos, la generación y uso de escenarios se vinculó con una rápida prototipación o aún construcción de las primeras versiones del nuevo sistema.

Los escenarios iniciales sirvieron para validar los prototipos e, indirectamente, la especificación de requerimientos. La evaluación de los prototipos condujo a la detección de malentendidos entre los expertos de dominio y los desarrolladores. Resolver tales malentendidos se torna más fácil con escenarios como una base común para la comunicación.

- ◆ **Los Escenarios facilitan el acuerdo parcial y la consistencia**

Los involucrados en un sistema bajo desarrollo tienen diferentes objetivos, aún sus percepciones de la realidad varían significativamente. Juntar a todos los involucrados y alcanzar un acuerdo general en los aspectos de un sistema consume demasiado tiempo y a veces es imposible. Similarmente, asegurar que el sistema a construirse está conforme con todos los aspectos de los sistemas existentes en la organización es impracticable. No obstante, casi todos los proyectos usaron escenarios para manejar el proceso de acuerdo y establecer consistencia parcial entre los sistemas existentes.

◆ **Escenarios de enlace y glosarios**

En varios proyectos, el vincular escenarios con un glosario amplio del proyecto estableció un entendimiento común de los términos usados entre diferentes grupos: desarrolladores, expertos de dominio y administradores.

Se notó una relación bidireccional interesante entre escenarios y glosarios. Los desarrolladores establecieron una infraestructura de hipertexto en una intranet de proyecto amplio que enlaza las partes correspondientes al escenario con los ítems del glosario.

Estos enlaces ayudaron a los desarrolladores y expertos del dominio a ajustar sus interpretaciones a los términos claves usados y así alcanzar un entendimiento común del proyecto. Más aún, estas relaciones ayudaron a nuevos miembros del proyecto a familiarizarse con la terminología del proyecto.

Podemos apreciar que estas conclusiones tienen una gran similitud con las características del uso del Léxico Extendido del Lenguaje.

◆ **Evolución de Escenarios**

En todos los proyectos, la definición de escenarios no fué una actividad corta, los escenarios evolucionaron a través del tiempo.

◆ **Derivando casos de prueba de los escenarios**

Casi todos los desarrolladores que se entrevistaron en este estudio, mencionaron la necesidad de basar las pruebas de sistema en los escenarios definidos con el cliente durante la etapa de Ingeniería de requerimiento. Esto significa apoyar al desarrollador para que pueda probar al cliente, que el sistema implementado alcanza los requerimientos.

◆ **Traceability**

Muchos desarrolladores también mencionaron la necesidad de un mayor apoyo de traceability, con traceability visto como un prerequisite para establecer el amplio ciclo de vida del uso de escenarios definidos.

Los escenarios desactualizados fueron inconsistentes con el diseño actual de versiones de prototipo y además no pudieron ser usados como base para casos de prueba. La Traceability posibilita la integración de cambio, ayudando a usuarios a mantener escenarios actualizados. Los desarrolladores deben establecer traceability entre niveles de abstracción de los escenarios, versiones de escenarios, escenarios y prototipos, escenarios y la especificación, y escenarios y casos de prueba. Establecer traceability requiere entender la relación entre los artefactos producidos durante el desarrollo del proyecto y los escenarios.

Por ultimo, podemos decir que del trabajo de investigación también se desprende que los usuarios necesitan mejores herramientas para manejar los escenarios y sus relaciones. Observaron una frecuente falta de apoyo apropiado de herramientas y la falta de capacidad para asegurar manualmente la consistencia entre escenarios o entre escenarios y otros artefactos. Difícilmente



dos proyectos cualquiera usen las mismas herramientas para el manejo de escenarios, excepto por el procesador de texto. Esto indica que no existe ninguna herramienta aceptada generalmente.

### **4.3. Esquema de descripción de Escenarios**

En esta sección se describe la estructura que se adopta en este trabajo para representar los escenarios.

La vista del modelo de escenarios aplicada es una estructura compuesta por el **nombre**, el **objetivo**, el **contexto**, los **recursos**, los **actores** y los **episodios**. El objetivo, el contexto, los recursos y los actores son sentencias declarativas, mientras que los episodios son un conjunto de sentencias con un lenguaje muy simple que hace posible la descripción operativa de comportamientos.

Para la descripción de escenarios se utilizó el template descrito en el Cuadro 7 [Leite]:

**Nombre:** título del escenario. En el caso de un sub-escenario, el título es el mismo que la sentencia episodio (ver abajo la definición Episodio), sin las restricciones y/o excepciones.

Sintaxis:

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

**Objetivo:** finalidad a ser alcanzada en el contexto del problema. El escenario describe el logro del objetivo.

Sintaxis:

[Sujeto] + Verbo + Predicado

**Contexto:** ubicación geográfica y temporal del escenario, y/o estado inicial del mismo.

Sintaxis:

Ubicación + Estado

donde Ubicación es:

Nombre

donde Estado es:

[Actor | Recurso] + Verbo + Predicado + {Restricciones}

**Recursos:** medios de soporte, dispositivos u otros elementos pasivos necesarios para estar disponibles en el escenario.

Sintaxis:

Nombre + {Restricciones}

**Actores:** personas o estructuras organizacionales que tienen un rol en el escenario.

Sintaxis:

Nombre

**Episodios:** conjunto de acciones que detallan el escenario y proveen su comportamiento.

Sintaxis:

<episodios> ::= <series>

<series> ::= <sentencia> | <series>

<sentencia> ::= <sentencia secuencial> | <sentencia no secuencial> |

<sentencia condicional> | <sentencia optativa>

<sentencia secuencial> ::= <sentencia episodio>

<sentencia condicional> ::= **Si** <condición> **entonces** <sentencia episodio>

<sentencia no secuencial> ::= # <series> #

<sentencia optativa> ::= [ <series> ]

donde <sentencia episodio> se describe:

[Actor | Recurso] + Verbo + Predicado + {Restricciones} + {Excepciones}

Cuadro 7 - Esquema para la descripción de escenarios

En la Figura 3, se describe el modelo de escenarios usando el diagrama de entidad-relación. Se observa en dicho diagrama que un episodio puede

concebirse como un escenario en sí mismo, esto posibilita la descomposición de un escenario en sub-escenarios.

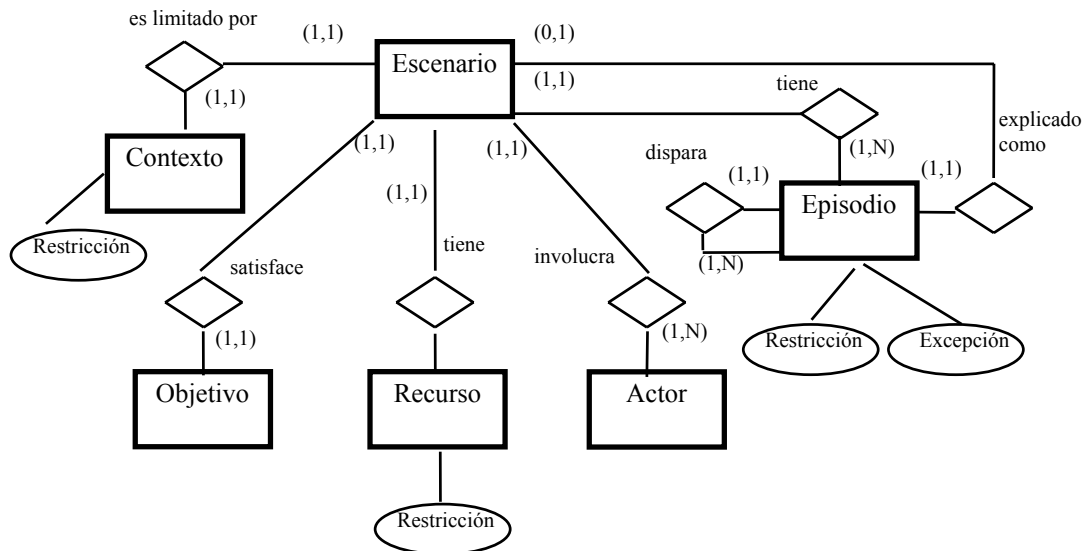


Figura.3: Diagrama de Entidad-Relación para el Modelo de Escenarios.

#### 4.4. Proceso de construcción de Escenarios

Algunos trabajos de investigación han profundizado en el estudio del LEL y Escenarios aplicándolos a varios casos de estudio [Hadad96] y [Hadad97], lo que ha permitido desarrollar heurísticas para la construcción de escenarios, las cuales serán implementadas en la herramienta que presenta esta tesis.

La construcción de los escenarios se basa exclusivamente en la información contenida en el LEL, al que se le aplican una serie de heurísticas, que se detallan más adelante en esta sección, para obtener una primera lista de escenarios candidatos.

Posteriormente se realizan nuevas entrevistas con los clientes, con el fin de ampliar información sobre aquellos escenarios cuya descripción resultara incompleta y/o confusa a partir de la información obtenida del LEL.

Esto implica que la derivación de escenarios que se detalla en el punto siguiente es el comienzo de la descripción de los escenarios, siendo luego necesario retornar al Universo de Discurso para concluirla.

Los escenarios se describen utilizando el esquema informado en la sección 4.3. y se emplea el **punto de vista** de la aplicación.

El proceso de construcción de los escenarios consta de las siguientes etapas:

1. Identificación de los actores de la aplicación;
2. Generación de la lista de escenarios candidatos, a partir de los actores principales;
3. Descripción de los escenarios candidatos, provenientes de los actores principales;
4. Ampliación de la lista de escenarios candidatos, a partir de los actores secundarios;
5. Descripción de los escenarios candidatos, provenientes de actores secundarios;
6. Revisión de los escenarios;
7. Validación de escenarios.

En la figura 4 se presentan gráficamente las etapas mencionadas y a continuación se detalla cada una de estas etapas.

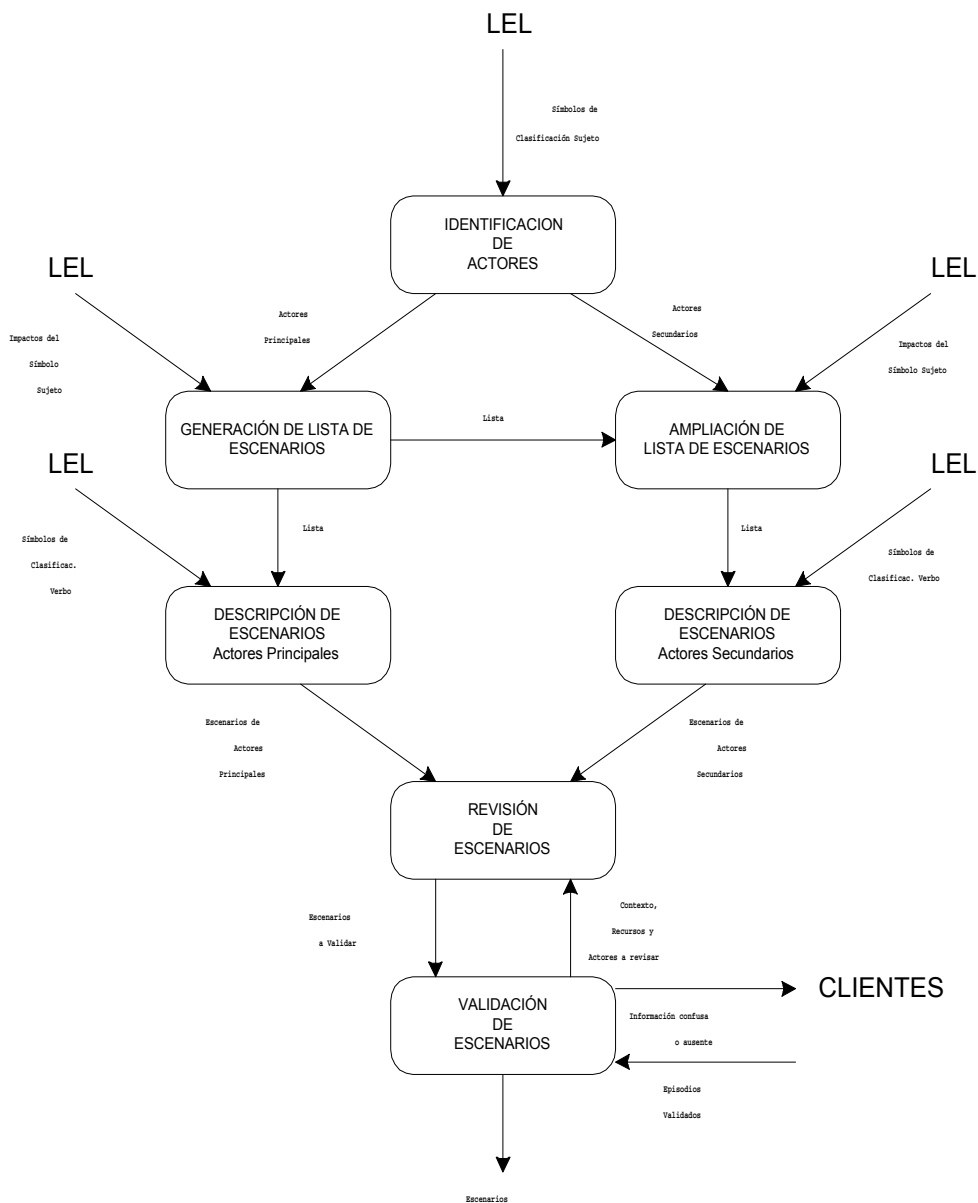


Figura. 4:- Etapas para la construcción de escenarios

#### 4.4.1. Identificación de los actores de la aplicación.

A partir del LEL, se identifican los símbolos que representan a los **actores** del Universo de Discurso. Estos símbolos pertenecen a la **clasificación Sujeto**, que agrupa a los individuos que interactúan en el Universo de Discurso.

Se identifica cuáles son los **actores principales** y cuáles los **actores secundarios**. Para ello, se considera que los actores principales son aquellos que realizan acciones directas sobre la aplicación, mientras que los actores

secundarios son los sujetos que reciben y/o brindan información, pero no ejecutan acciones directas sobre la aplicación.

#### **4.4.2 Generación de la lista de Escenarios candidatos, a partir de los actores principales.**

Del símbolo del LEL correspondiente a cada actor principal se obtienen sus **impactos**. Cada impacto representa un posible escenario, que se incorpora a la lista de escenarios candidatos. El **nombre del escenario** se conforma con la acción (verbo) contenida en el impacto en tiempo infinitivo.

De esta lista, se eliminan los escenarios repetidos, en el caso de que distintos actores ejecuten la misma acción.

#### **4.4.3. Descripción de los Escenarios candidatos, provenientes de los actores principales.**

A partir de la lista candidata, se describe cada uno de los escenarios, utilizando el esquema descrito en el Cuadro 5. Esta descripción se realiza aplicando las reglas que se detallan a continuación:

Para cada impacto que representa un posible escenario, se verifica si contiene un símbolo del LEL que pertenezca a la **clasificación Verbo**.

##### **4.4.3.1 En caso afirmativo:**

Se busca dicho símbolo en el LEL.

Se define el **objetivo** del escenario basándose en el nombre del escenario, el punto de vista de la aplicación y, como ayuda adicional, la **noción** de este símbolo que representa una actividad.

Se identifican los **actores** y los **recursos** que surgen de la información de este símbolo, y que pertenecen, a su vez, a la **clasificación Sujeto** y a la **clasificación Objeto** respectivamente.

Se definen los **episodios** del escenario a partir de cada **impacto** del símbolo.

Para la definición del **contexto**, se observa si en el símbolo del cual se obtuvo este escenario, el impacto que lo originó tenía alguna secuencia con respecto a los restantes impactos. En caso afirmativo, se describe en el contexto como precondición la ejecución previa de dichos impactos que corresponden a escenarios candidatos. También se define en el contexto la ubicación física o temporal en la realización del escenario.

Se completa el ítem **dudas** con toda aquella información que parezca confusa o ausente.

##### **4.4.3.2 Si el impacto no contiene un símbolo del LEL que pertenezca a la clasificación Verbo:**

Se identifican **símbolos** dentro del **impacto**.

Se buscan dichos símbolos en el LEL para obtener mayor información sobre el posible escenario.

Se define el **objetivo** en función del nombre del escenario y el punto de vista de la aplicación.

Se identifican posibles **actores** y **recursos** del escenario, a través de los símbolos asociados, pertenecientes a la **clasificación Sujeto y Objeto** respectivamente.

No se describen episodios partiendo del LEL, se deja su definición para un paso posterior.

Para la definición del **contexto** se observa si el impacto del símbolo que lo originó tenía alguna secuencia con respecto a los restantes impactos, en caso afirmativo éstos se incorporan como precondition.

Se completa el ítem **dudas** con información confusa o ausente.

A medida que se describen los episodios pueden surgir las **excepciones**.

#### **4.4.4. Ampliación de la lista de Escenarios candidatos, a partir de los actores secundarios.**

Se aplica a los actores secundarios el procedimiento descrito en el punto 4.4.2, para obtener otros escenarios candidatos e incorporarlos a la lista.

#### **4.4.5. Descripción de los Escenarios candidatos, provenientes de los actores secundarios.**

Se aplica el procedimiento descrito en el punto 4.4.3 a los nuevos escenarios incorporados a la lista en el punto anterior.

#### **4.4.6. Revisión de los Escenarios.**

Descritos todos los escenarios candidatos, éstos se revisan de la siguiente manera:

##### **4.4.6.1 Detección** de escenarios candidatos como episodios simples dentro de otros escenarios.

Estos escenarios se originan en impactos de los símbolos de actores que no contienen otro símbolo perteneciente a la **clasificación Verbo**. Por lo tanto, son escenarios cuyos episodios no se han definido, ver punto 4.4.3.2.

Se entrevista al cliente para determinar si este escenario involucra más de un episodio.

En caso afirmativo, se mantiene el escenario en la lista y, en los escenarios que figura como una acción simple, se reemplaza el episodio por el nombre del escenario. Es decir, ocurre una detección de **sub-escenarios**.

En caso contrario, se elimina de la lista el escenario candidato.

##### **4.4.6.2 Detección** de escenarios candidatos como un conjunto de episodios dentro de otros escenarios.

En escenarios provenientes de actores principales puede ocurrir que en la descripción de episodios, un conjunto de éstos correspondan a un escenario proveniente de un actor secundario.

Se reemplaza el conjunto de episodios por el escenario proveniente del actor secundario. Nuevamente ocurre la detección de **sub-escenarios**.

#### 4.4.6.3 **Unificación** de escenarios.

Si dos o más escenarios presentan episodios comunes o poseen el mismo objetivo y el mismo contexto, estos escenarios se agrupan en un sólo escenario. De requerirse, se utiliza la forma condicional para describir episodios diferentes.

#### 4.4.6.4 Definición de **restricciones**.

Durante la revisión de los escenarios, se definen las restricciones a los episodios. Estas provienen del contexto del problema y también, en función del contexto del escenario.

#### 4.4.6.5 Detección de escenarios que corresponden a **excepciones**.

Cuando un escenario queda “aislado” en cuanto a que su contexto no identifica claramente qué escenarios se realizan previamente, o el escenario no mantiene ninguna relación con otro escenario, se verifica si este escenario puede o no corresponder a una excepción de un escenario o de un episodio dentro de un escenario.

Una vez finalizado este punto se obtiene la lista de escenarios a validar con los clientes.

### 4.4.7. Validación de Escenarios.

#### 4.4.7.1 Se **valida** con los **clientes** cada uno de los escenarios incluidos en la lista definitiva, prestando especial atención al ítem dudas que existiese en el escenario.

La validación de los escenarios permite detectar errores u omisiones o ampliar información en los episodios. En el caso de ampliación de información de un episodio, se puede detectar que episodios considerados simples involucran un conjunto de acciones.

#### 4.4.7.2 Se realizan las **correcciones** necesarias en cada ítem del escenario.

En el caso de ampliación de información para un episodio, según su complejidad, se puede:

- abrir el episodio en varios episodios dentro del mismo escenario, o
- reemplazar el episodio por un nuevo escenario que involucre el conjunto de acciones detectado e incorporar el nuevo escenario a la lista.

#### 4.4.7.3 Se realiza una **revisión** del contexto, los actores y los recursos de cada escenario, en función de las correcciones introducidas en los episodios en los puntos precedentes.



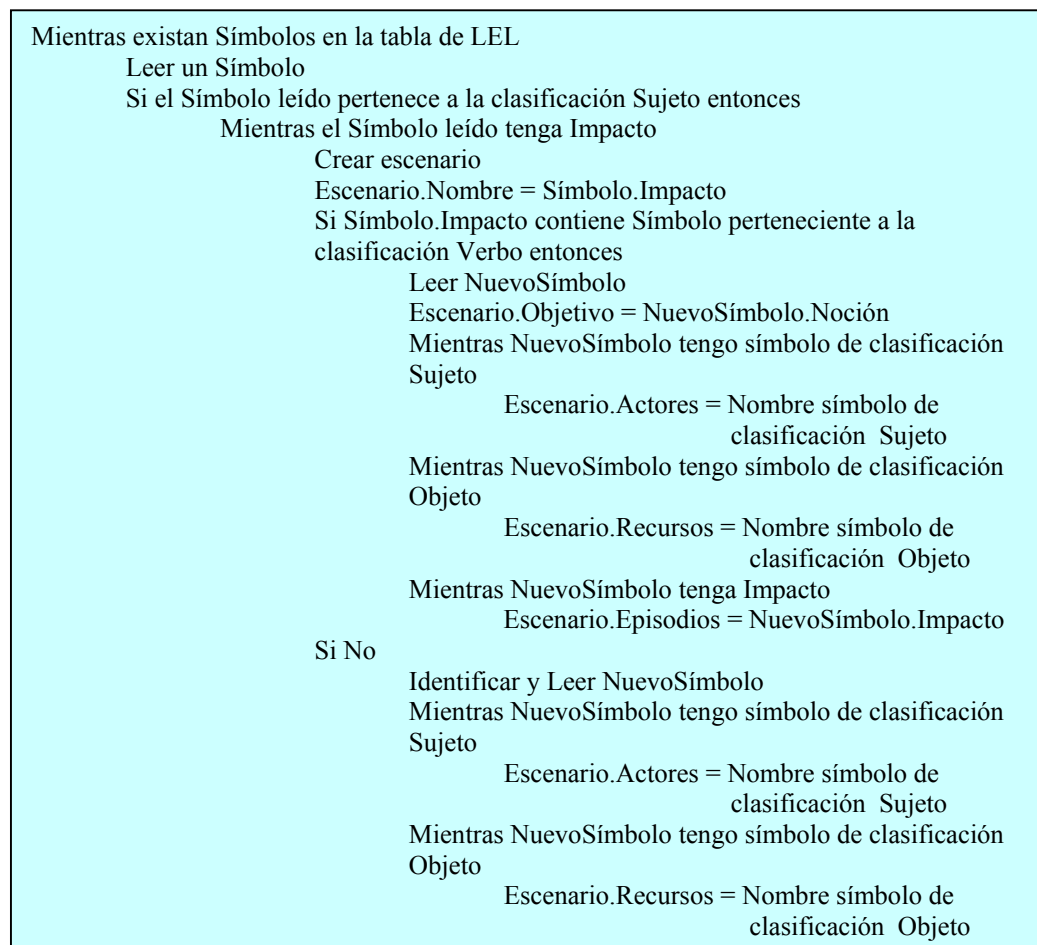
A partir de este punto, se obtiene la lista definitiva de escenarios y sus descripciones, correspondientes a la fase de elicitación de requerimientos.

#### **4.5. Algoritmo para generar Escenarios**

Desde el punto de vista informático, podemos pensar que la herramienta contara con un Repositorio en el que se almacenara toda la información relacionada con cada proyecto que el usuario esté desarrollando. Entre los componentes de este repositorio habrá una Base de Datos formada por una tabla de LEL, una tabla de Escenarios y un documento que contiene el Universo del Discurso obtenido.

La herramienta, a partir de la base de datos del LEL, identificara los símbolos que representan a los actores del Universo de Discurso. Estos símbolos pertenecen a la clasificación Sujeto y cada impacto representa un posible escenario, que se incorpora a la lista de escenarios candidatos.

La herramienta, a medida que genera la lista de escenarios candidatos, simultáneamente, describirá cada uno de los escenarios, aplicando las reglas que se detallaron en los puntos 4.4.2 y 4.4.5 de este informe. TILS verifica si el impacto que genera un escenario candidato contiene un símbolo del LEL que pertenezca a la clasificación Verbo, si ocurre esto entonces es posible generar la descripción de dicho escenario. A continuación en el cuadro 8 se expone el algoritmo que representa lo expuesto:



*Cuadro 8: Algoritmo para generar escenarios candidatos.*