

**Universidad de las Ciencias Informáticas**  
**Facultad 3**



**Título:** “Evaluación de la usabilidad en el diseño arquitectónico de los proyectos del CEIGE, basado en una herramienta para el manejo de patrones.”

**Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas**

**Autores:** Orlienis Rodríguez López  
Jordan Sandino Martínez

**Tutora:** MsC. Iliannis Pupo Leyva

**Co-tutor:** Ing. David Martínez Alarcón

**La Habana. Junio, 2012**

# Declaración de Autoría

---

---

## DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste, firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Autor 1

**Orlienis Rodríguez López**

---

Autor 2

**Jordan Sandino Martínez**

---

Tutora

**MsC. Iliannis Pupo Leyva**

---

Co-tutor

**Ing. David Martínez Alarcón**

## **RESUMEN**

El presente trabajo se realizó con el propósito de evaluar la usabilidad en la arquitectura del software en los proyectos desarrollados en el CEIGE, de modo que contribuya a elevar el soporte de usabilidad de sus productos. Para ello, se elaboró, como propuesta de solución, un procedimiento que está compuesto por tres fases, donde se especifican actividades, roles y artefactos en cada una de ellas. En dicho procedimiento se integran buenas prácticas de diferentes tendencias estudiadas durante el desarrollo del trabajo y se propone el uso de la herramienta USPA, la cual fue desarrollada por la necesidad de informatizar el marco SAU, utilizado por el método de evaluación SALUTA para la evaluación de la usabilidad en la arquitectura del software. La propuesta fue validada en los subsistemas Inventario y Costo y procesos del CEIGE, logrando identificar propiedades y patrones de usabilidad a incorporar en el diseño arquitectónico de dichos subsistemas. Mediante la correcta aplicación de la propuesta, se garantiza un mejor soporte de usabilidad en el desarrollo de productos en el CEIGE.

## **Palabras Claves**

Arquitectura de Software, patrones de usabilidad, usabilidad.

# Tabla de Contenido

---

---

<b>DECLARACIÓN DE AUTORÍA</b> .....	II
<b>RESUMEN</b> .....	III
<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	6
<b>1.1 Introducción</b> .....	6
<b>1.2 Definición de Usabilidad</b> .....	6
1.2.1 Importancia de la Usabilidad.....	7
1.2.2 Atributos de Usabilidad.....	8
1.2.3 Propiedades de Usabilidad .....	9
1.2.4 Patrones de Usabilidad.....	9
1.2.5 Beneficios de la evaluación de Usabilidad .....	9
<b>1.3 Arquitectura de Software</b> .....	10
1.3.1 Patrón Arquitectónico.....	11
<b>1.4 Impacto de la Usabilidad en la Arquitectura de Software</b> .....	11
<b>1.5 Investigaciones realizadas por CALISOFT</b> .....	12
<b>1.6 Tendencias relacionadas con la Usabilidad</b> .....	12
1.6.1 Proyecto STATUS .....	12
1.6.2 Método de evaluación SALUTA.....	15
1.6.3 Ingeniería de Usabilidad .....	17
1.6.4 ISO 13407 .....	19
1.6.5 Comparación entre tendencias internacionales estudiadas .....	20
<b>1.7 Conclusiones del Capítulo</b> .....	23
<b>CAPÍTULO 2: SOLUCIÓN PROPUESTA</b> .....	24
<b>2.1 Introducción</b> .....	24
<b>2.2 Solución propuesta</b> .....	24
2.2.1 Nombre.....	24
2.2.2 Objetivos.....	24
2.2.3 Alcance.....	24
2.2.4 Responsables.....	24
2.2.5 Términos.....	25
2.2.6 Referencias .....	25
<b>2.3 Procedimiento para la evaluación de la usabilidad en el diseño arquitectónico de los proyectos del CEIGE</b> .....	26
2.3.1 Fase: Requisitos.....	26
2.3.2 Fase: Diseño .....	28
2.3.3 Fase: Pruebas de aceptación .....	31
<b>2.4 Herramienta USPA</b> .....	32
2.4.1 Características y requerimientos técnicos.....	33
2.4.2 Uso de la herramienta USPA.....	33
<b>2.5 Conclusiones del Capítulo</b> .....	41

# Tabla de Contenido

---

---

<b>CAPÍTULO 3: VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN</b> .....	42
<b>3.1 Introducción al Capítulo</b> .....	42
<b>3.2 Descripción del subsistema Inventario</b> .....	42
<b>3.3 Descripción del subsistema Costo y procesos</b> .....	42
<b>3.4 Aplicación del procedimiento en los subsistemas Inventario y Costo y procesos.</b> .....	43
3.4.1 Fase: Requisitos.....	43
3.4.2 Fase: Diseño.....	46
<b>3.5 Determinar beneficios financieros económicos de la propuesta</b> .....	54
<b>3.6 Validación de la herramienta USPA mediante las pruebas de caja negra</b> .....	55
<b>3.7 Conclusiones del Capítulo</b> .....	56
<b>CONCLUSIONES GENERALES</b> .....	57
<b>RECOMENDACIONES</b> .....	58
<b>BIBLIOGRAFÍA</b> .....	59
Figura 1 Marco de definición de usabilidad de acuerdo a Nielsen.....	7
Figura 2 Método de diseño para usabilidad propuesto en STATUS.....	13
Figura 3 Fases del procedimiento.....	26
Figura 4 Fase: Requisitos.....	28
Figura 5 Fase: Diseño.....	31
Figura 6 Fase: Pruebas de aceptación.....	32
Figura 7 Pantalla de bienvenida de la herramienta USPA.....	34
Figura 8 Interfaz principal de USPA.....	35
Figura 9 Interfaz Crear Escenario.....	35
Figura 10 Interfaz Listado de Escenarios.....	36
Figura 11 Editar Escenario.....	37
Figura 12 Interfaz Listado de Atributos del Escenario.....	37
Figura 13 Propiedades del Atributo Priorizado.....	38
Figura 14 Patrones de usabilidad de la propiedad seleccionada.....	38
Figura 15 Descripción de un patrón de usabilidad.....	39
Figura 16 Listado de Patrones de cada escenario de uso.....	40
Figura 17 Descripción de la usabilidad en la Arquitectura.....	48
Figura 18 Listado de patrones de usabilidad a incorporar en la arquitectura.....	49

# Tabla de Contenido

---

---

Figura 19 Número de propiedades de usabilidad por cada escenario.....	52
Figura 20 Número de patrones de usabilidad por cada escenario.....	53
Figura 21 Soporte de la usabilidad.....	54
Tabla 1 Listado de Propiedades y Patrones de Usabilidad. ....	14
Tabla 2 Relaciones entre Atributos, Propiedades y Patrones.....	17
Tabla 3 Comparación entre tendencias internacionales estudiadas.....	20
Tabla 4 Fase: Requisitos. ....	26
Tabla 5 Fase: Diseño.....	28
Tabla 6 Fase: Pruebas de aceptación.....	31
Tabla 7 Listado de priorización de requisitos de usabilidad del producto .....	43
Tabla 8 Escenario de Uso: Insertar Producto.....	46
Tabla 9 Escenarios seleccionados.....	47
Tabla 10 Matriz Escenarios /patrones y propiedades de usabilidad. ....	49
Tabla 11 Soporte de usabilidad para cada escenario.....	51
Tabla 12 Soporte de usabilidad para cada escenario.....	53

## INTRODUCCIÓN

Hoy en día los desarrolladores de software tienen como una de sus metas fundamentales que los sistemas informáticos sean lo más usable posible. Se busca que la interacción del usuario con la aplicación que maneja sea sencilla y eficaz y por tanto sea aceptada por el mismo. Para lograr esto es recomendable hacer un diseño que se adapte al máximo a las características de los usuarios con el fin de obtener un mejor entendimiento y satisfacción del mismo.

Mediante un proceso de pruebas aparejado al desarrollo de software, se miden atributos de calidad como: la funcionalidad, la fiabilidad, la eficiencia, la mantenibilidad, la portabilidad y la usabilidad. Esta última abarca la capacidad de que un software sea comprendido, aprendido, usado y atractivo para el usuario, en condiciones específicas de uso. [1]

La usabilidad era primariamente una propiedad de presentación de la información. Por lo tanto, separar la presentación y la aplicación hacía fácil modificar tal presentación después de una prueba de usuario. Esta idea no era suficiente para alcanzar la usabilidad en los sistemas, ya que se veía grandemente afectada por la funcionalidad. Aún, si la presentación del sistema y la funcionalidad son diseñadas extremadamente bien, una no conformidad de usabilidad con impacto funcional, impacta nocivamente en la arquitectura del software, pues esta responde a un patrón de usabilidad el cuál es necesario incluir en el diseño arquitectónico. Lo anterior expuesto evidencia la estrecha relación entre usabilidad y arquitectura.

La consideración de la usabilidad como un factor importante en el desarrollo de aplicaciones ha ido en aumento en los últimos años. Basándose en el fundamento teórico de la usabilidad de software tradicional, en los principios establecidos por los estándares internacionales y en la experiencia adquirida por el diseño y desarrollo de aplicaciones, se han desarrollado metodologías y guías para el desarrollo de aplicaciones con criterios de usabilidad. [2]

Tradicionalmente se defiende la idea a nivel internacional de medir y mejorar la usabilidad una vez finalizado el software. Las investigaciones a profundidad sobre el tema han demostrado que esto no es cierto, algunos proyectos han fracasado por no tener en cuenta la usabilidad desde la creación del sistema. [3]

Los cambios a la apariencia de la interfaz de usuario del sistema, fácilmente se pueden hacer al final del desarrollo del proceso sin incurrir en grandes costos. Sin embargo, en los sistemas complejos, como

pueden ser los entornos distribuidos, los transaccionales, los multicanal y aquellos en los que pueden haber miles de usuarios conectados simultáneamente, la usabilidad tiene implicaciones arquitectónicas, por lo que hay que tenerla en cuenta desde el inicio del diseño del sistema, es decir, desde lo que se denomina momento de Arquitectura del Software, pues hacerlo en una etapa tardía del desarrollo sería muy prohibitivo y relativamente caro.

Actualmente, empresas y países desarrolladores de software han mostrado interés por la relación entre Arquitectura del Software y usabilidad, planteándose como meta lograr que el usuario interactúe de una manera fácil y segura, desarrollando productos de software en los que la usabilidad del sistema sea primordial para el éxito.

Cuba es uno de estos países que busca garantizar una mejora constante de la calidad en cuanto a usabilidad de sus productos de software, para ajustarse a un mercado altamente competitivo.

En el contexto de la Batalla de ideas surge la Universidad de las Ciencias Informáticas (UCI), que tiene como uno de sus objetivos principales producir software y servicios informáticos. Con el transcurso del tiempo esta universidad se ha convertido en la vanguardia de las empresas desarrolladoras de software en Cuba y ha sido un pilar fundamental en la informatización de algunos sectores de la sociedad. Persiguiendo este objetivo, surgen los centros de desarrollo enfocados a la producción de software en una rama determinada. Dentro de ellos se encuentra el Centro de Informatización de la Gestión de Entidades (CEIGE), donde se desarrolla un software de Planificación de Recursos Empresariales (ERP por sus siglas en inglés), denominado CEDRUX v1.0.

En el desarrollo de CEDRUX v1.0, existen grandes problemas en el tratamiento de la usabilidad:

- No se tiene en cuenta la usabilidad como atributo de calidad en la toma de requisitos funcionales, solo se hace énfasis en la usabilidad como requisito no funcional del sistema, lo que trae consigo que no se tenga en cuenta la usabilidad como elemento funcional y sólo como parte visual del sistema.
- No se vincula la usabilidad con la arquitectura de manera adecuada, lo que trae consigo incremento de los costos de desarrollo por el rediseño de productos ya terminados, debido a la detección de no conformidades en la fase de pruebas de aceptación de los productos.

Un ejemplo de esto es el caso de una no conformidad encontrada específicamente en el subsistema de Inventario, donde los desarrolladores aplicando semi-usabilidad<sup>1</sup> habían permitido que los usuarios registraran muchos productos antes de poderlos confirmar, esto en la usabilidad se conoce como el patrón de agregación de objetos, que no es más que la capacidad que provee el sistema de agrupar varios datos (Productos) y aplicarles una misma acción (Confirmar).

El problema surge cuando el sistema no notificaba al usuario el límite máximo de productos a confirmar. Cuando los arquitectos analizaron el problema se percataron de que no se tuvo en cuenta que el sistema solo permitiera seleccionar hasta 20 productos, al llegar a éste límite, el sistema debía notificar a los usuarios de que al llegaron al margen máximo para procesar lotes y debía invalidar la opción de seleccionar productos para esta funcionalidad.

A raíz de las condiciones descritas hasta el momento, se identificó como **problema a resolver** que: En el CEIGE, la insuficiente priorización de los atributos de calidad en el diseño de la arquitectura, trae consigo que se detecten no conformidades de usabilidad con impacto arquitectónico<sup>2</sup> en fases tardías del desarrollo.

Tomando como **objeto de estudio**: La usabilidad en el desarrollo de software. Siendo el **campo de acción**: La usabilidad en la arquitectura de los sistemas de gestión.

El **objetivo general** de esta investigación es: Evaluar la usabilidad en el diseño arquitectónico de los proyectos del CEIGE, basado en una herramienta para el manejo de patrones que contribuya a prevenir problemas de usabilidad con impacto arquitectónico en fases tardías del desarrollo.

Planteándose, como **objetivos específicos**:

1. Identificar actividades de las fases de análisis y diseño, teniendo en cuenta patrones de usabilidad que puedan ser incorporados al proceso de desarrollo del CEIGE.
2. Desarrollar una herramienta para el manejo de patrones que permita optimizar el proceso de evaluación de la usabilidad.
3. Validar la propuesta de solución mediante el método de evaluación SALUTA.

---

<sup>1</sup> Usabilidad alcanzada sin ninguna técnica de aseguramiento de la misma .Se logra con el sentido común y experiencia del equipo de desarrollo.

<sup>2</sup> Problemas, en su mayoría funcionales, que provocan cambios significantes en la Arquitectura de un sistema.

4. Validar la herramienta mediante pruebas de caja negra.

Las **tareas de la investigación** que se llevarán a cabo para dar solución a los objetivos trazados son:

- Analizar actividades de usabilidad que se aplican a lo largo del ciclo de vida de un software y que se puedan ajustar al entorno del centro, principalmente en las fases de análisis y diseño.
- Caracterizar el estado actual que presenta la usabilidad en los productos de software de la UCI y a nivel internacional.
- Desarrollar el catálogo de patrones de usabilidad a ser incorporados en la arquitectura.
- Desarrollar la herramienta para el manejo de patrones de usabilidad que permita optimizar la evaluación de la usabilidad.
- Evaluar la propuesta obtenida en proyectos del centro mediante el método de evaluación SALUTA.

Validar la herramienta mediante pruebas de caja negra.

Para el desarrollo de este trabajo se combinan diferentes Métodos y Técnicas, como se muestra a continuación:

## **A nivel teórico:**

**Métodos de análisis-síntesis e inducción-deducción:** Para el estudio de las concepciones y conceptos empleados en el campo de la relación entre usabilidad y arquitectura del software, permitiendo la extracción de los elementos más importantes que se relacionan en este campo.

**Análisis histórico-lógico:** Para conocer, con mayor profundidad, los antecedentes y las tendencias actuales referidas a la usabilidad; además de conceptos, términos y vocabularios propios del campo.

**Método de la observación científica:** Se conoce el problema y el objeto de investigación, estudiando su curso natural, sin alteración de las condiciones naturales, es decir que la observación tiene un aspecto contemplativo.

## **A nivel empírico:**

**Revisión de documentos:** Para el estudio y análisis, partiendo de documentación científica, de los conceptos relacionados con la usabilidad y arquitectura.

# Introducción

---

---

**La entrevista:** Para realizar una conversación planificada con los desarrolladores y arquitectos para obtener información sobre la usabilidad del Cedrux v1.0.

# Capítulo 1: Fundamentación Teórica

---

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción

En este capítulo se hace un detallado estudio del arte de los principales conceptos definidos por autores que se han especializado en el tema de la Usabilidad de Software. Se definirá su importancia, impacto en la Arquitectura de Software. Se realiza un estudio de las diferentes tendencias, métodos y modelos en la mejora de la usabilidad. Todo ello, con el objetivo de ser utilizado como soporte teórico para la investigación.

### 1.2 Definición de Usabilidad

La usabilidad se ha convertido en uno de los factores principales de estudio en el campo del desarrollo de software. Al igual que muchos términos de la ingeniería de software, tiene varias definiciones según el enfoque que se le dé en un determinado contexto como por ejemplo:

“Usabilidad es el grado en el que un producto puede ser utilizado por usuarios específicos para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un determinado contexto de uso”. [2]

“Usabilidad es la capacidad de un software de ser comprendido, aprendido, usado y atractivo para el usuario, en condiciones específicas de uso”. [1]

“Usabilidad se refiere al costo / esfuerzo de aprender a manejar un producto”. [4]

ISO 9241 define la usabilidad en términos de la calidad del trabajo de un sistema en uso, la cual depende de todos los factores que pueden influenciar el uso de un producto en el mundo real: factores organizacionales (prácticas de trabajo, ubicación o apariencia de un producto), diferencias individuales entre usuarios (factores culturales y preferencias), experiencia, etc.

En el modelo de Nielsen, la usabilidad es:

“Parte de la utilidad del sistema, la cual es parte de la aceptabilidad práctica y, finalmente parte de la aceptabilidad del sistema”, tal como muestra la figura.

# Capítulo 1: Fundamentación Teórica

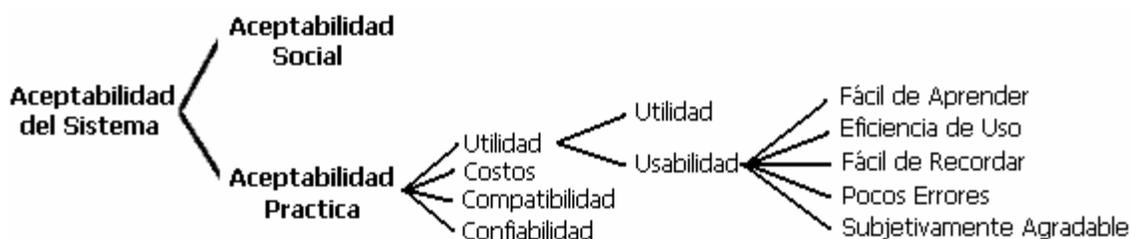


Figura 1 Marco de definición de usabilidad de acuerdo a Nielsen.

La usabilidad de los sistemas software se suele evaluar sobre el sistema finalizado, intentando asignar valores a los atributos de usabilidad clásicos [5] [6] [7]: aprendizaje, eficiencia, fiabilidad y satisfacción.

A modo de resumen de los conceptos antes mencionados, los autores de esta investigación llegan a la conclusión de que la usabilidad no es más que la simpleza, la facilidad, la comodidad y la practicidad con que un producto es usado. En otras palabras, tiene relación con la eficacia percibida de un objeto y la posibilidad de aprovechar todo su potencial.

## 1.2.1 Importancia de la Usabilidad

La usabilidad en los productos de software se ha convertido paulatinamente en una necesidad y casi obligación, para las empresas dedicadas al diseño y desarrollo de software. No contar con una evaluación<sup>3</sup> o prueba de este tipo afecta la calidad final del producto y conlleva a incrementar los gastos en el proceso de fabricación.

Algunos beneficios de la usabilidad son los siguientes:

- Reducción de los costos de aprendizaje.
- Disminución de los costos de asistencia y ayuda al usuario.
- Optimización de los costos de diseño, rediseño y mantenimiento.
- Mejora la calidad de vida de los usuarios, ya que reduce su estrés, incrementa la satisfacción y la productividad.

---

<sup>3</sup> Determina la funcionalidad del sistema en producción y la satisfacción de los usuarios.

# Capítulo 1: Fundamentación Teórica

---

A continuación se muestran datos y referencias que justifican dicha afirmación:

- La usabilidad demuestra reducciones del ciclo de desarrollo de los productos de 33-50%. [8] [9]
- 63% de todos los proyectos de desarrollo de software sobrepasan su presupuesto, siendo las cuatro causas más importantes relacionadas con usabilidad. [10] [11]
- El porcentaje de código que se dedica al desarrollo de la interfaz con los usuarios ha ido aumentando a lo largo de los años hasta un promedio 47-60% del conjunto de la aplicación. [12] [13]
- 80% de las tareas de mantenimiento se deben a requerimientos de usuarios no previstos, quedando el resto debido a fallos y errores. [14] [15]

Muchas son las ventajas que la usabilidad puede proporcionar y por ello debería ser tratada como un factor de calidad estratégico y relevante. Diseñar un producto con una alta calificación respecto a su usabilidad no es fácil de conseguir, por lo que esta debe ser considerada en todas las fases del desarrollo, desde el momento en que este comienza hasta que el producto o servicio es puesto en disposición del público.

Todos estos beneficios implican una reducción y optimización general de los costos de producción, así como un aumento en la productividad de los sistemas.

## 1.2.2 Atributos de Usabilidad

Un estudio realizado, revela la diferencia de criterios y definiciones entre investigadores acerca del término atributo de usabilidad. Generalmente se ha aceptado que es un elemento preciso y medible de la noción abstracta que es la usabilidad. Entre estos atributos de usabilidad tenemos [16]:

- **Facilidad de aprendizaje:** Es la forma rápida y sencilla con que los usuarios pueden empezar a hacer un trabajo productivo con un sistema que es nuevo para ellos.
- **Eficiencia del uso:** Es el número de tareas por unidad de tiempo que el usuario puede realizar al usar el sistema.
- **Fiabilidad:** Se refiere a la tasa de error en el uso del sistema y el tiempo que tarda en recuperarse de los errores.
- **Satisfacción:** Las opiniones subjetivas que los usuarios se forman en el uso del sistema. No es más

# Capítulo 1: Fundamentación Teórica

---

que cuando un usuario siente que tiene el control del sistema.

## 1.2.3 Propiedades de Usabilidad

Las propiedades de la usabilidad son un conjunto de características que se pueden mejorar en un sistema.

- Proporcionar información: el sistema proporciona información continua sobre su funcionamiento al usuario.
- Gestión de errores: incluye la prevención y recuperación de errores.
- Consistencia: la coherencia tanto de la interfaz de usuario y de funcionamiento del sistema.
- Orientación: orientación en línea sobre el funcionamiento del sistema.
- Minimizar la carga cognitiva: el diseño del sistema debe reconocer humanos, limitaciones cognitivas, la memoria a corto plazo, entre otras cosas.
- Accesibilidad: incluye el acceso multimodal, la internacionalización y el apoyo a las personas con discapacidad. [18] [19]

## 1.2.4 Patrones de Usabilidad

En el mundo de los patrones de diseño en el que un patrón es una solución reutilizable a un problema que ocurre comúnmente, se ha acuñado el término patrón de usabilidad [20] para referirse a “una descripción de soluciones que mejoran los atributos de usabilidad” [19].

El patrón de usabilidad se ha definido como un mecanismo a ser incorporado en el diseño de una arquitectura software a fin de abordar una propiedad de usabilidad concreta, un patrón arquitectónico determinará cómo se incorporará este patrón de usabilidad en una arquitectura software; es decir, qué efecto tendrá la inclusión del patrón de usabilidad en los componentes de la arquitectura del sistema. [21]

## 1.2.5 Beneficios de la evaluación de Usabilidad

La evaluación de la usabilidad durante el desarrollo, reporta importantes beneficios y ahorros tanto económicos como en tiempo para obtener productos que resulten de mayor utilidad, tanto para los usuarios como para quienes desarrollan los sitios, como por ejemplo:

# Capítulo 1: Fundamentación Teórica

---

- Mayor rapidez en el diseño y producción (ayuda a determinar la dirección del diseño e identificar los problemas).
- Reducción del costo de trabajos de rediseño posterior, centrándose en los usuarios durante las fases iniciales de desarrollo.
- Aumento de la tasa de conversión, en el caso de los sitios de comercio electrónico, con el consiguiente aumento de las ventas.
- Aumento de la satisfacción de los usuarios.
- Disminución de costos en la atención al cliente/ayuda.
- Ventaja competitiva apoyándose en la calidad, mayor número de retención de usuarios, y lealtad de clientes.
- Refuerzo de imagen de marca. [22]

## 1.3 Arquitectura de Software

En los últimos años la comunidad de Ingeniería de software ha hecho conciencia de que la Arquitectura de Software es un importante instrumento para el cumplimiento de los requisitos de calidad de un sistema.

La definición oficial de Arquitectura del Software es la establecida por la IEEE Standard 1471-2000, esta plantea que: “La Arquitectura del Software es la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”.

Según RUP: La arquitectura de un sistema es la visión común en la que todos los empleados (desarrolladores y otros usuarios) deben estar de acuerdo, o como poco, deben aceptar. La arquitectura nos da una clara perspectiva del sistema completo, necesaria para controlar el desarrollo. [23]

En sí, la Arquitectura de Software da una visión de lo que el sistema debe hacer y se apoya para ello en el arquitecto que es quien define las principales herramientas y tecnologías, dígase patrones de diseño, estilos arquitectónicos, lenguajes de implementación, framework, etc. Por consiguiente la Arquitectura de Software sienta las bases para el desarrollo y la construcción de un sistema de software de manera económico y fiable con la realización de la mayor cantidad de requerimientos no funcionales, obteniendo una alta satisfacción del cliente.

A todo ello se puede agregar que la Arquitectura de Software es la encargada de la descripción y estudio

# Capítulo 1: Fundamentación Teórica

---

de las propiedades estructurales de los sistemas de software.

Es por todo lo antes mencionado que el diseño de una buena arquitectura es muy importante para que el software salga con la calidad requerida, obteniendo como resultado el éxito del proyecto. Además de eso, la arquitectura permite una muy buena comunicación entre las personas involucradas en el desarrollo del sistema y una temprana documentación de las decisiones de diseño tomadas.

## 1.3.1 Patrón Arquitectónico

Los escenarios de la usabilidad conforman cada uno un patrón arquitectónico, los cuales toman lugar dentro de la jerarquía de Ingeniería de software. Estos patrones pueden ser útiles para el diseño de la arquitectura, incluso si no son usados.

Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, responsabilidades e interrelaciones y una serie de recomendaciones para organizar los distintos componentes fundamentales de la estructura de un sistema de software. [18]

## 1.4 Impacto de la Usabilidad en la Arquitectura de Software

En la confección de un sistema no basta con tener en cuenta la presentación y la funcionalidad. Sobre todo en sistemas complejos, como pueden ser los entornos distribuidos, los transaccionales, los multicanal y aquellos en los que puede haber miles de usuarios conectados simultáneamente, hay que tener en cuenta la usabilidad desde el inicio del diseño del sistema, es decir, desde lo que se denomina momento de Arquitectura del Software.

Cuanto más tarde se detectan los problemas, más cuesta arreglarlos. Si al diseñar una interfaz, se quieren crear interacciones y diálogos en el entorno tecnológico, como que el usuario pueda visualizar el progreso de sus peticiones, que pueda deshacer acciones (undo), que pueda disponer de un entorno multilingüe, cancelar una operación que lleva mucho tiempo en espera, y reutilizar información que ha introducido anteriormente, esto no va a ser posible. La respuesta es muy sencilla, una vez hecho el diseño de la Arquitectura del Software, no se pueden realizar estos cambios a nivel de interfaz para mejorar la usabilidad, porque esto implicaría la modificación de casi el sistema completo.

# Capítulo 1: Fundamentación Teórica

---

Si se analizan los escenarios de interacción que no pudieron ser cambiados, a simple vista se puede ver que la causa de que no se puedan implementar es que hay que tener en cuenta al usuario al inicio del diseño del sistema, es decir, en la confección de la Arquitectura del Software. [23]

## 1.5 Investigaciones realizadas por CALISOFT

Este centro tiene como uno de sus objetivos: Verificar el cumplimiento de las políticas de calidad al evaluar la usabilidad de sistemas de gestión sobre plataformas web sin usuario final.

En este centro existe un procedimiento para la evaluación de la usabilidad, el cual se realiza mediante una lista de chequeo de beneficios de usabilidad y se lleva a cabo una vez finalizado el software, este procedimiento no se ajusta a las necesidades del CEIGE porque el objetivo es evaluar la usabilidad en el proceso de desarrollo del software y específicamente esta investigación se centra en la fase de Análisis y Diseño, además este procedimiento se integra solamente con la Ingeniería de la Usabilidad, dejando fuera el impacto de este atributo en la Arquitectura del Software.

## 1.6 Tendencias relacionadas con la Usabilidad

### 1.6.1 Proyecto STATUS

Por la importancia que tiene en la actualidad el estudio de la relación directa entre las decisiones arquitectónicas y la satisfacción de ciertos atributos de calidad como la usabilidad, surge el proyecto STATUS (Software Architectures That support Usability), cuyo objetivo es el desarrollo de técnicas y procedimientos a incorporar durante el diseño de un sistema software, con el fin de conseguir mejoras en la usabilidad del sistema a construir. [13]

Esta perspectiva, contrasta con la alternativa tradicional, que consiste en medir la usabilidad del sistema una vez que éste se ha construido y mejorarla después iterativamente. La propuesta que se está abordando en el proyecto STATUS, consiste en adelantar ese ciclo de evaluación/mejora al momento arquitectónico, como se muestra a continuación en la siguiente figura (Ver Figura 2):

# Capítulo 1: Fundamentación Teórica

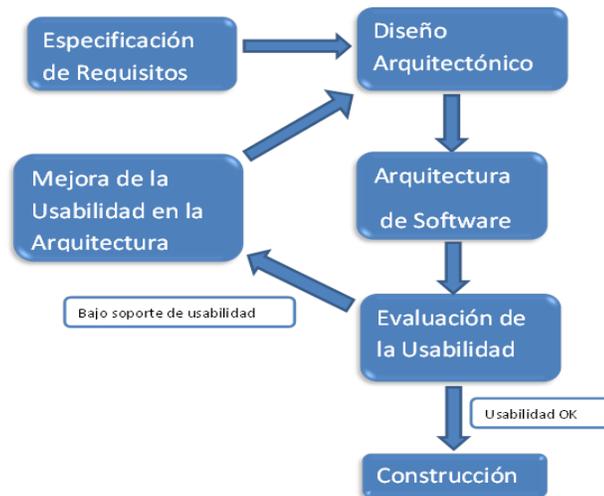


Figura 2 Método de diseño para usabilidad propuesto en STATUS.

## 1.6.1.1 Surgimiento de STATUS

Debido al intenso estudio e interés por las relaciones existentes entre la usabilidad y la Arquitectura de Software, y por la importancia que la usabilidad está adquiriendo cada vez más en el desarrollo de software, surge el proyecto STATUS. Este proyecto se inició el 1 de diciembre del 2001, por la colaboración de academias e industrias europeas, dentro de estas esta la Universidad de Groningen (Holanda), Universidad politécnica de Madrid (España). [13]

Los objetivos científicos y tecnológicos del proyecto son los siguientes:

- Identificar los atributos de usabilidad que son posiblemente afectados por la arquitectura de software.
- Estudiar cómo los atributos de usabilidad pueden ser influenciados por la arquitectura de software y el desarrollo de un estilo arquitectónico que soporta la usabilidad.
- Identificar los patrones o modelos de usabilidad que están repetidos en los dominios del comercio electrónico, para estudiar sus relaciones con la usabilidad y sus mejoras con respecto a los de atributos de la calidad.
- Proponer un proceso de desarrollo que integre las técnicas de desarrollo tradicional de software con técnicas apropiadas al campo de usabilidad.

# Capítulo 1: Fundamentación Teórica

## 1.6.1.2 Propuestas de patrones de STATUS

El proyecto STATUS propone una serie de patrones necesarios para que la arquitectura soporte la usabilidad. (Ver Tabla 1)

Tabla 1 Listado de Propiedades y Patrones de Usabilidad.

Propiedad de Usabilidad	Patrón de Usabilidad
Mapeo Natural	Diferentes lenguajes
Consistencia	
Accesibilidad	
Coherencia, Accesibilidad	Diferentes métodos de acceso
Retroalimentación	Alerta
Gestión de errores, Retroalimentación	Indicación de estado
Control explícito del usuario, Adaptabilidad	Acceso directo
Gestión de errores	Formulario / Campo de validación
Gestión de errores	Deshacer
Orientación, Gestión de errores	Ayuda contextual
Orientación, Gestión de errores	Asistente
Orientación, Gestión de errores	Ayuda estándar
Orientación, Gestión de errores	Recorrido
Minimizar la carga cognitiva, Adaptabilidad, Gestión de errores	Modelo de flujo de trabajo

# Capítulo 1: Fundamentación Teórica

Gestión de errores	Historial de Registro
Gestión de errores	Vista previa
Adaptabilidad	Historial de usuario
Control explícito del usuario, Gestión de errores	Cancelar
Control explícito del usuario	Multitarea
Minimizar la carga cognitiva, Gestión de errores	Agregación de comandos
Control explícito del usuario	Acciones para múltiples objetos
Minimizar la carga cognitiva, Gestión de errores	Reutilización de la información

## 1.6.2 Método de evaluación SALUTA

SALUTA (Escenario Basado en el Análisis de Usabilidad al Nivel de la Arquitectura) es el primer método desarrollado para evaluar arquitecturas desde la perspectiva de la usabilidad del sistema, siendo el resultado de los estudios de Folmer y Gurp. [14]

### Funcionamiento y estructura de SALUTA

Este método hace uso de marcos de referencia que expresan las relaciones que existen entre usabilidad y Arquitectura de Software. Dichos marcos de referencias, incluyen un conjunto integrado de soluciones de diseño, diseños o propiedades, que tienen un efecto positivo sobre la usabilidad en un sistema de software.

SALUTA analiza cuatro atributos que están directamente relacionados con la usabilidad de un sistema de software: facilidad de aprendizaje, eficiencia de uso, confiabilidad y satisfacción. El mismo, se basa, al igual que las dos tendencias analizadas anteriormente, en escenarios, que en este caso, son escenarios de uso que agrupan de uno a más perfiles de uso, valga la redundancia, donde cada uno representa la usabilidad requerida por el sistema. [14]

# Capítulo 1: Fundamentación Teórica

---

## 1.6.2.1 Marco SAU

SAU (Usability and Software Architecture) es un framework que utiliza el Saluta para capturar la relación entre usabilidad y Arquitectura del Software. [17]

El framework consiste en un conjunto integrado de soluciones de diseño tales como patrones y propiedades de usabilidad que han sido identificados en varios casos en la industria, el software de hoy en día, y las encuestas de la literatura. Estas soluciones, en la mayoría de los casos, tienen un efecto positivo en la usabilidad, pero son difíciles de adaptar a las aplicaciones ya que tienen un impacto arquitectónico. El framework se puede utilizar para guiar e informar a la fase de diseño arquitectónico. Esto puede disminuir los costes de desarrollo mediante la reducción de la cantidad de problemas de usabilidad que deben ser corregidos durante las últimas etapas de desarrollo. SAU es el marco de evaluación de usabilidad que utiliza SALUTA.

### Estructura del marco SAU

El marco consta de tres capas:

1. Capa de atributos: una serie de atributos de usabilidad forman el denominador más común de las nociones existentes de la usabilidad.
2. Capa de propiedades: las propiedades de la usabilidad encarnan la heurística y los principios de diseño que los investigadores, en el campo de la usabilidad, consideran que tienen una influencia directa en la usabilidad del sistema. Las propiedades de la usabilidad son un medio para vincular las pautas de usabilidad arquitectónicamente sensibles a los atributos de usabilidad.
3. Capa de patrones: consta de patrones de usabilidad arquitectónicamente sensibles, identificados a partir de varios estudios de casos y aplicaciones industriales modernas de software, así como de las colecciones de patrones de usabilidad, y se omiten los patrones en los que la sensibilidad de la arquitectura no es clara.

El Marco-Relación que se muestra a continuación (Ver Tabla 2), evidencia la estrecha relación que tienen estas capas, siendo esta la propuesta a seguir por SAU para medir el impacto de la usabilidad en la arquitectura.

# Capítulo 1: Fundamentación Teórica

Tabla 2 Relaciones entre Atributos, Propiedades y Patrones.

Atributos de Usabilidad	Propiedades de Usabilidad	Patrones de Usabilidad
Satisfacción	Control explícito de usuario	Deshacer
Capacidad de Aprendizaje	Guía	Asistente
	Control explícito de usuario	Deshacer
Eficiencia	Retroalimentación	Indicación de Estado
		Deshacer
		Alerta
Confiabilidad	Prevención de Errores	Alerta

## 1.6.3 Ingeniería de Usabilidad

La Ingeniería de Usabilidad se puede definir como una aproximación al desarrollo de sistemas en la que se especifican niveles cuantitativos de usabilidad y el sistema se construye para alcanzar dichos niveles, que se conocen como métricas. Esta ingeniería proporciona un modo práctico de asegurar que el software desarrollado alcanza un cierto nivel de usabilidad y está basada en la evaluación mediante test de usabilidad con usuarios.

El objetivo de la Ingeniería de usabilidad es realizar mejoras en la utilidad y la usabilidad de productos que se encuentran en desarrollo y entonces, aumentar el valor de un producto para un cliente. Con vistas a lograr este objetivo, esta cuenta con 10 procesos, de los cuales se escoge uno de ellos. [25]

### 1.6.3.1 Ciclo de Vida de la Ingeniería de Usabilidad

**Análisis del perfil del usuario:** Se puede obtener a través de herramientas como por ejemplo cuestionarios y entrevistas el perfil de los usuarios potenciales. Cuando se obtienen los datos se realiza su análisis con el fin de describir los factores más relevantes de impacto sobre la usabilidad del producto o servicio (por ejemplo, el tipo de uso, la cantidad de horas dedicadas al uso de sistemas informáticos y el

# Capítulo 1: Fundamentación Teórica

---

nivel de experiencia previa). Este proceso, por tanto, aporta un conjunto de datos clave al análisis de tareas.

**Análisis de tareas:** Este proceso describe las tareas realizadas actualmente por los usuarios, sus patrones definidos de flujo de trabajo, los cuales se originan de sus esquemas mentales y las necesidades de información para realizar su trabajo. Se quiere identificar: qué el usuario hace, de qué manera lo hace, y qué necesita para hacerlo. Así de esta forma, se logra el entendimiento conceptual de las tareas que deberán formar parte del sistema en desarrollo.

Para la obtención de dicho entendimiento se pueden utilizar varias técnicas tales como entrevistas, observación sistemática, estudio de diagramas de afinidad, entre otras.

**Definición de los objetivos de usabilidad:** Este proceso es responsable de la especificación de los objetivos cualitativos y cuantitativos de usabilidad. Estos se relacionan con los resultados obtenidos en los dos procesos anteriores y con la especificación de requisitos de aceptabilidad y satisfacción del usuario, respectivamente. En este sentido, los objetivos de usabilidad serán utilizados como parámetros clave durante los procedimientos de test.

**Diseño del sistema:** Este proceso se basa en un conjunto de actividades, que consiste en dos aspectos principales:

1. El análisis estructurado del sistema, en el cual se diseña su modelo conceptual considerando la organización y el flujo de trabajo de la funcionalidad del producto o servicio propuesto.
2. La definición y diseño de la interfaz del sistema. Para llevar a cabo este proceso se utilizan los resultados del análisis de las tareas y los objetivos predeterminados.

También se propone el uso de técnicas auxiliares así como el diseño paralelo, el diseño de alternativas y el diseño de participación, a su vez engloba, los procedimientos de diseño en el cual los usuarios participan activamente.

**Implementación de prototipos:** Este proceso consiste en un estudio experimental de determinados aspectos del sistema. Su propósito es reducir el tiempo y coste de desarrollo del producto o servicio, permitiendo, de esta manera, la realización de test con los usuarios potenciales del sistema. La implementación de prototipos es más rápida y más barata y por tanto, se puede llevar a cabo cuántas veces sean necesarias. [25]

# Capítulo 1: Fundamentación Teórica

---

**Realización de test:** En este proceso no sólo se verifica y valida los prototipos, sino también se evalúa su usabilidad. Con el uso de procedimientos formales de test o técnicas de inspección y métodos de adquisición de datos de usabilidad como herramientas de apoyo, se examinan todos los aspectos del prototipo en relación a los requisitos predeterminados. No obstante, este proceso también puede ser realizado con la versión final del producto o servicio.

**Rediseño:** Más que un proceso, el rediseño se caracteriza por ser un indicador de decisión basado en los resultados de los análisis de los test. De esta manera, si se identifica que el prototipo, producto o servicio no cumplen con los requisitos y estándares establecidos, se desvía el flujo del ciclo de desarrollo a la definición de los objetivos de usabilidad, con el objetivo de verificar su validez. Sin embargo, en algunos casos, se inicia el rediseño en proceso de análisis de tareas.

**Implementación del producto o servicio:** Después de la evaluación de los prototipos y de su aceptación, se inicia la implementación del producto o servicio con toda su funcionalidad y prestaciones previstas. Este proceso está interconectado con las actividades de actualización y mantenimiento del sistema.

**Retroalimentación del usuario:** Finalmente, cuando se ha realizado la instalación del producto o servicio, se obtienen nuevas informaciones complementarias del usuario con el propósito de usarlas para mejorar e intensificar el diseño del sistema, de nuevas versiones y de nuevos productos o servicios con características similares. Para ello, se utilizan test de usabilidad formales, cuestionarios, entrevistas, entre otros. [25]

## 1.6.4 ISO 13407

La ISO 13407 proporciona una guía para alcanzar la calidad en el uso mediante la incorporación de actividades de naturaleza iterativa involucradas en el Diseño Centrado en el Usuario (DCU).

El Diseño Centrado en el Usuario (DCU) lo describe como una actividad multidisciplinar, que incluye factores humanos y conocimientos y técnicas de ergonomía con el objetivo de mejorar la efectividad y eficiencia, las condiciones de trabajo y contrarrestar los posibles efectos adversos de su uso. [22]

Describe los cuatro principios del Diseño Centrado en el Usuario:

- Involucrar activamente a los usuarios

# Capítulo 1: Fundamentación Teórica

---

- Asignación adecuada de funciones al sistema y el usuario
- Soluciones de diseño iterativas
- Diseño multidisciplinar

Y las cuatro actividades del Diseño Centrado en el Usuario:

- Entender y especificar el contexto de uso
- Especificar los requisitos del usuario y de la organización
- Producir más de una solución de diseño candidata
- Contrastar los diseños con los requisitos.

## 1.6.5 Comparación entre tendencias internacionales estudiadas

En la Tabla 3 se pretende reflexionar sobre los aspectos comunes y diferenciales entre las diferentes tendencias a la mejora de usabilidad. Mediante el método de observación científica de la literatura se llegó a la conclusión de que algunos de los indicadores utilizados para comparar modelos y procedimientos son:

- **Ámbito de aplicación:** Determina en qué etapa del proceso de desarrollo es aplicado.
- **Integrable al proceso de desarrollo:** Determina actividades para su integración en el proceso de desarrollo.
- **Procesos:** Determina si tiene procesos definidos para su aplicación.
- **Validación:** Define qué métodos de evaluación están definidos.
- **Objetivo:** Define el objetivo perseguido por el mismo.
- **Representación:** Define elemento o gráficas de los componentes que intervienen.

Tabla 3 Comparación entre tendencias internacionales estudiadas.

<b>Indicadores</b>	<b>Saluta</b>	<b>STATUS</b>	<b>Ingeniería de la Usabilidad</b>	<b>ISO 13407</b>
--------------------	---------------	---------------	------------------------------------	------------------

# Capítulo 1: Fundamentación Teórica

Ámbito de Aplicación	Se centra en la etapa de diseño del producto.	Se centra en la etapa de diseño del producto.	Procesos de desarrollo iterativos.	Genérico. Se aplica a cualquier ciclo de vida y a la construcción de sistemas iterativos.
Integrable al Proceso de Desarrollo	No es integrable al Proceso de Desarrollo. Su integración se basa solo en la fase de diseño.	No es integrable al Proceso de Desarrollo, su integración se basa solo en la fase de diseño.	Sí es integrable al Proceso de Desarrollo. Integra gradualmente técnicas dependiendo del proceso de desarrollo de software.	No es integrable al Proceso de Desarrollo. Su aplicación se realiza en el diseño y depende del ambiente y estado del proceso de diseño.
Procesos	Define varios procesos.	No se definen procesos. Cuenta con una serie de patrones que se aplican en el diseño.	Estructura Propia. Define varios procesos o técnicas a utilizar dependiendo del estado del proceso de desarrollo software.	No se definen procesos. Describe unos principios básicos sin estipular métodos específicos.
Validación	Define métodos de validación.	No cuenta con un modelo que evalúe el nivel de usabilidad del diseño gracias a la utilización de	No define método de Validación.	No define métodos de Validación.

# Capítulo 1: Fundamentación Teórica

		los patrones.		
Objetivo	Desarrolla una serie de sistemas iterativos a través del proceso de desarrollo software.	Mejora la usabilidad de un software aplicando un proceso específico de diseño para la usabilidad.	Orienta en la incorporación de técnicas de usabilidad, en el punto adecuado del proceso de desarrollo software.	Desarrolla sistemas iterativos a través de la incorporación del DCU en el proceso de desarrollo software.
Representación	Plano e iterativo. Consta de cuatro fases iterativas hasta alcanzar los resultados esperados.	Mejoramiento continuo de la Arquitectura hasta que cumpla con criterios de usabilidad y, posteriormente, es plana hasta su entrega.	Presenta una representación continua que depende del nivel del proceso de desarrollo software.	Plano e iterativo. Consta de cinco fases iterativas hasta alcanzar los resultados esperados.

Es notable la creciente preocupación sobre el papel que puede jugar la usabilidad para el éxito de un producto, son diferentes los modelos y enfoques, los cuales se encuentran dispersos, no existe un procedimiento que integre dichos modelos.

La integración de estas tendencias sería de gran importancia ya que se lograría hacer partícipe al usuario en todo el ciclo de vida del desarrollo del software y se mejoraría el producto constantemente en cada etapa, reduciendo los costos y el tiempo.

De las tendencias estudiadas se seleccionaron las mejores prácticas como:

# Capítulo 1: Fundamentación Teórica

---

- SALUTA: se hará uso de todos los pasos de este método de evaluación para determinar el soporte de la usabilidad en la arquitectura de software.
- STATUS: se seleccionará de este proyecto el catálogo de patrones de usabilidad propuesto por el mismo para incluir en el diseño arquitectónico.
- Ingeniería de la Usabilidad: se hará uso del primer proceso de este método de diseño y solución el cual es el Análisis del perfil del usuario, donde se caracteriza al usuario según sus conocimientos.
- ISO 13407: se toman actividades propuestas por este estándar como especificar los requisitos del usuario y de la organización y contrastar dichos requisitos con el diseño.

La problemática es esencial, entender la relación entre la arquitectura de software y la usabilidad para asegurar que el sistema últimamente la alcance, de tal forma que es importante que tanto ingenieros de software como especialistas en usabilidad entiendan dicha relación, por lo que se hace evidente la necesidad de un procedimiento que integre las tendencias estudiadas anteriormente.

## 1.7 Conclusiones del Capítulo

Los elementos analizados en las secciones anteriores apuntan la necesidad de definición y especificación de un procedimiento que evalúe la usabilidad en todas las fases de desarrollo de los productos del CEIGE principalmente en el diseño arquitectónico.

La revisión de los principales métodos, modelos y procedimiento para la evaluación de la usabilidad, demostró la existencia de dificultades en la adecuación de estos para su aplicación en los proyectos del sistema CEDRUX por diversos factores explicados en epígrafes anteriores.

# Capítulo 2: Solución Propuesta

---

## CAPÍTULO 2: SOLUCIÓN PROPUESTA

### 2.1 Introducción

El presente capítulo describe la propuesta de un procedimiento para realizar la evaluación de la usabilidad en el diseño arquitectónico en los proyectos del CEIGE, integrado por tres fases fundamentales.

Dicho procedimiento, fue creado para dar cumplimiento a los objetivos planteados en la investigación y se centra en definir los roles y responsabilidades, las actividades correspondientes a cada fase y los artefactos de entrada y salida que intervienen en cada una de ellas. Además, en este capítulo se hará referencia a la herramienta Patrones de Usabilidad y Arquitectura (USPA – Usability Patterns and Architecture), la cual fue desarrollada por los autores de este trabajo y propuesta a ser utilizada durante la puesta en práctica del procedimiento propuesto, y se especificará su uso a modo de manual de usuario.

### 2.2 Solución propuesta

#### 2.2.1 Nombre

Procedimiento para la evaluación de la usabilidad en el diseño arquitectónico de los proyectos del CEIGE.

#### 2.2.2 Objetivos

- Evaluar la usabilidad en la arquitectura desde fases tempranas del desarrollo en los productos del CEIGE.
- Mejorar la usabilidad de los productos a través de las decisiones de diseño incorporados en la arquitectura del software, logrando así un buen soporte de la misma.
- Determinar si una arquitectura candidata es compatible con las características necesarias de usabilidad.

#### 2.2.3 Alcance

La propuesta comprende la evaluación de la usabilidad en el diseño arquitectónico de proyectos del CEIGE.

#### 2.2.4 Responsables

Durante el procedimiento intervienen diferentes roles, que cuentan con sus respectivas responsabilidades, dichos roles serán los siguientes:

## Capítulo 2: Solución Propuesta

---

- Analista del sistema

Es el responsable de realizar el análisis del perfil de usuario y de los requisitos de usabilidad durante la fase de Requisitos.

- Arquitecto del sistema

Es el responsable de determinar los escenarios y patrones de usabilidad, para la toma de decisiones en la arquitectura del software, durante la fase de Diseño.

- Probador

Es el encargado de asegurar, en la fase de Pruebas de Aceptación, que el sistema cuente con la usabilidad requerida por el usuario, mediante el empleo de la Lista de chequeo de usabilidad (Ver ANEXO 1).

- Desarrollador

Es el encargado de corregir las No Conformidades de Usabilidad detectadas por el Probador durante la fase de Pruebas de Aceptación.

### 2.2.5 Términos

Artefacto: Producto tangible del proyecto que es producido, modificado y usado en las actividades.

Actividad: Tarea que tiene un propósito y es asignada a un rol.

Rol: Función que una persona desempeña en una situación determinada.

No Conformidades de Usabilidad (NCU): Errores existentes en el producto que hacen que el mismo no cumpla con los requisitos de usabilidad especificados por el cliente para su uso.

### 2.2.6 Referencias

- IPP 1000:2008 Elaboración y aprobación de los procedimientos y lineamientos para la actividad productiva.
- STATUS "Software Architecture for Usability".
- Experiences with Software Architecture Analysis of Usability.
- SALUTA: Compressor & Esuite.

# Capítulo 2: Solución Propuesta

## 2.3 Procedimiento para la evaluación de la usabilidad en el diseño arquitectónico de los proyectos del CEIGE.

El presente procedimiento propone seguir tres fases fundamentales para llevar a cabo la evaluación de la usabilidad en el diseño arquitectónico en los proyectos del CEIGE, dichas fases se muestran relacionadas a continuación en la figura 3:

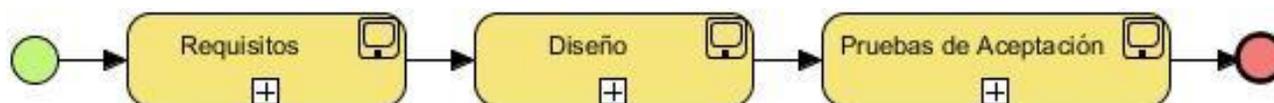


Figura 3 Fases del procedimiento.

### 2.3.1 Fase: Requisitos

Tabla 4 Fase: Requisitos.

Fase: Requisitos			
El objetivo de esta fase, consiste en determinar, enumerar y clasificar todas las características, capacidades y restricciones que ha de cumplir y a las que se verá sometido el sistema, así como los beneficios de usabilidad que debe brindar el mismo.			
Roles	Actividades	Artefactos de entrada	Artefactos de salida
Analista del sistema.	<ul style="list-style-type: none"><li>- Caracterizar usuarios del sistema.</li><li>- Definir requisitos de usabilidad.</li></ul>	-	<ul style="list-style-type: none"><li>- Listado de atributos y propiedades de usabilidad. (Creado)</li></ul>

En la fase de Requisitos, inicialmente, se aplica la Ingeniería de Usabilidad, de la cual se propone realizar la tarea de caracterizar los usuarios que van a usar el sistema en cuanto a:

- Conocimiento del dominio: Consiste en saber cuánto sabe el usuario sobre el dominio.

## Capítulo 2: Solución Propuesta

---

- Conocimiento del sistema: Conocimiento, experiencia o estudios de alguna herramienta, cómo opera y cómo usarlo.
- Evaluación de la Interacción: Consiste en obtener los patrones de uso del sistema por parte de los usuarios, frecuencia con la que el usuario emplea el sistema.
- Complejidad: Si las tareas a realizar son muy complejas.
- Foco de control: Quién dirige la interacción usuario-proceso.
- Información: Determina donde se origina la información y como fluye entre el usuario y el sistema o viceversa.

Una vez caracterizados los usuarios, el Analista del sistema deberá entrevistarse con los mismos, para realizar un análisis detallado del negocio, donde estos especifiquen qué requisitos de usabilidad debe tener el sistema. Posteriormente, el Analista del sistema enfoca estos requisitos como propiedades que responden a atributos de usabilidad. Para ello, se debe tener en cuenta que existen atributos de usabilidad que son más relevantes para el sistema, como son:

- Satisfacción: El usuario siente que tiene el control del sistema.
- Facilidad de aprendizaje: Facilidad con que el usuario realiza una tarea en el sistema.
- Eficiencia: Tiempo que demora el sistema en dar respuesta a una tarea del usuario.
- Fiabilidad: No debe existir errores en la realización de una tarea.

Finalmente, el Analista del sistema elabora un Listado de atributos y propiedades de usabilidad (**Ver ANEXO 2**): Este documento recoge cada atributo de usabilidad y las propiedades que le van a dar cumplimiento a los mismos, basándose en los requisitos de usabilidad descritos por el usuario.

### **Representación gráfica**

# Capítulo 2: Solución Propuesta

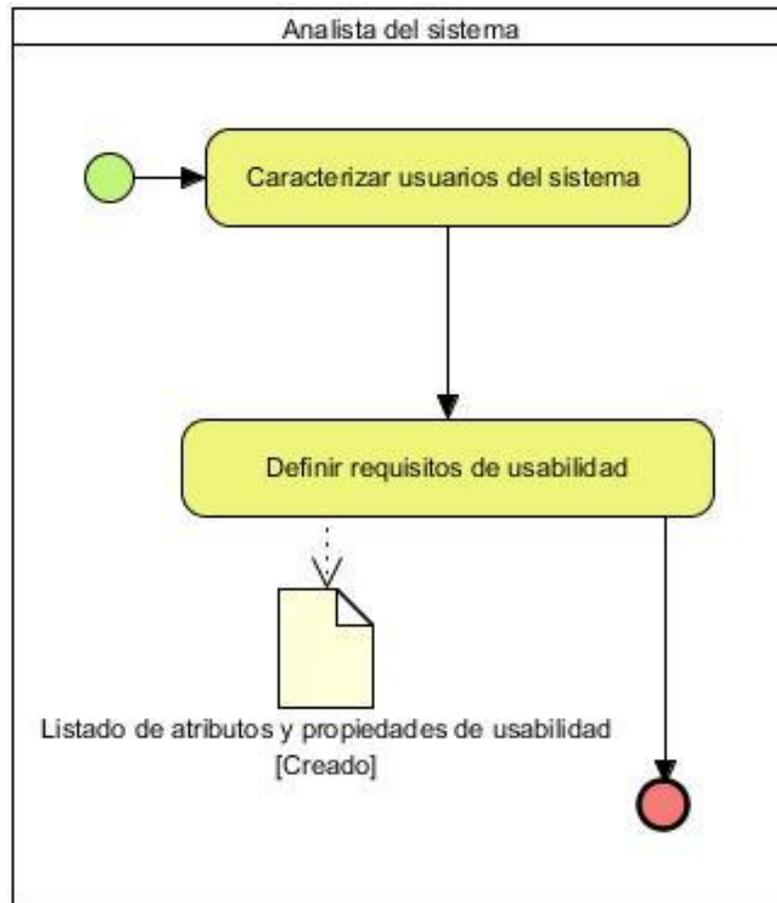


Figura 4 Fase: Requisitos

## 2.3.2 Fase: Diseño

Tabla 5 Fase: Diseño

Fase: Diseño			
El objetivo de esta fase, consiste en garantizar que los atributos y propiedades de la usabilidad sean incorporados en el diseño arquitectónico, mediante los patrones de usabilidad.			
Roles	Actividades	Artefactos de entrada	Artefactos de salida
Arquitecto del sistema.	- Crear perfiles de uso.	- Listado de atributos y	- Listado de patrones

## Capítulo 2: Solución Propuesta

	<ul style="list-style-type: none"><li>- Describir la arquitectura.</li><li>- Evaluar escenarios.</li><li>- Interpretar resultados.</li></ul>	<ul style="list-style-type: none"><li>propiedades de usabilidad. (Consultado)</li><li>- Diseño arquitectónico. (Consultado)</li><li>- Listado de escenarios de uso. (Consultado)</li><li>- Elementos de usabilidad del sistema. (Consultado)</li></ul>	<ul style="list-style-type: none"><li>de usabilidad generados por USPA. (Creado)</li><li>- Listado de escenarios de uso. (Creado)</li><li>- Elementos de usabilidad del sistema. (Creado)</li></ul>
--	--	--	---

En la fase de Diseño, básicamente, se propone aplicar los pasos que plantea el método de evaluación SALUTA, como actividades de dicha fase. Esta fase será llevada a cabo totalmente por el Arquitecto del sistema, el cual deberá, primeramente, crear los perfiles de uso.

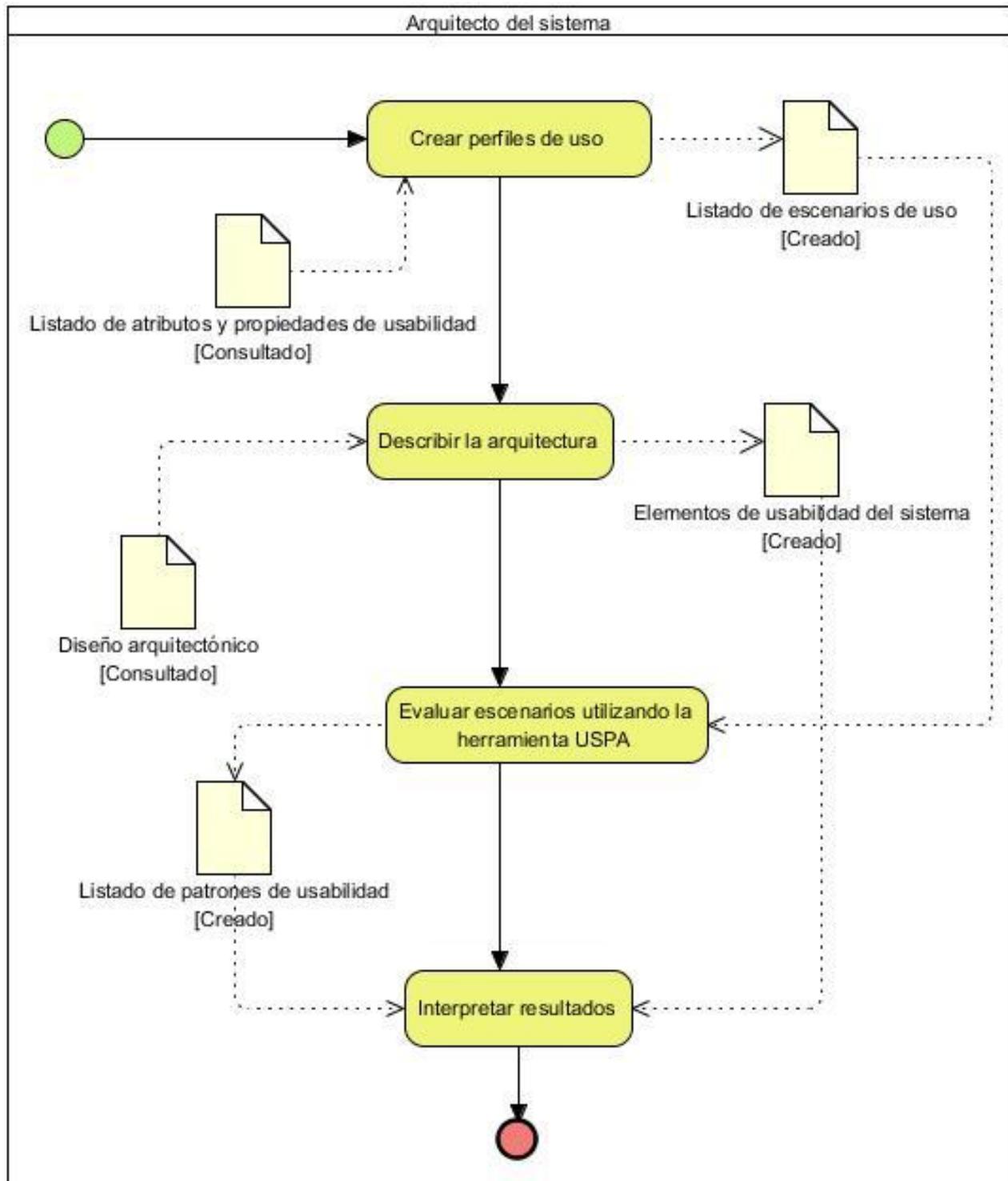
De esta actividad, se obtiene un Listado de escenarios de uso (**Ver ANEXO 3**) que expresan la usabilidad requerida por el sistema y se determina el atributo de prioridad para cada escenario seleccionado. Para ello se cuenta con el Listado de atributos y propiedades de usabilidad generado en la fase de Requisitos.

Posteriormente, se describe la arquitectura, donde se deberán identificar los elementos de usabilidad del sistema, haciendo uso del diseño arquitectónico. Luego, se deben evaluar los escenarios mediante el uso de la herramienta USPA, para realizar dicha evaluación, por cada escenario se analiza qué atributo y propiedad de usabilidad lo afecta. Como resultado de esta evaluación se obtiene un Listado de patrones de usabilidad (**Ver ANEXO 4**) a incorporar en el diseño arquitectónico.

Por último, se interpretan los resultados para arribar a conclusiones acerca del soporte de la usabilidad en la arquitectura del software.

### Representación gráfica

## Capítulo 2: Solución Propuesta



## Capítulo 2: Solución Propuesta

Figura 5 Fase: Diseño.

### 2.3.3 Fase: Pruebas de aceptación

Tabla 6 Fase: Pruebas de aceptación

Fase: Pruebas de aceptación			
El objetivo de esta fase, consiste en verificar que se cumplan los atributos y propiedades de usabilidad determinados en la fase de Requisitos.			
Roles	Actividades	Artefactos de entrada	Artefactos de salida
Probador. Desarrollador.	<ul style="list-style-type: none"><li>- Verificar el cumplimiento de atributos y propiedades de usabilidad mediante el uso de la Lista de chequeo de usabilidad.</li><li>- Elaborar Registro de NCU.</li><li>- Corregir las NCU detectadas.</li></ul>	<ul style="list-style-type: none"><li>- Lista de chequeo de usabilidad. (Consultada)</li><li>- Registro de NCU. (Consultado)</li></ul>	<ul style="list-style-type: none"><li>- Registro de NCU. (Creado)</li></ul>

En la fase de Pruebas de aceptación, inicialmente, el Probador utiliza la Lista de chequeo de usabilidad, que consiste en una serie de interrogantes mediante las que se verifica el cumplimiento de los atributos y propiedades de usabilidad que fueron determinados en la fase de Requisitos. Luego, si se detectan NCU, el Probador, deberá elaborar el Registro de NCU (**Ver ANEXO 5**) para que, posteriormente, el Desarrollador corrija las NCU detectadas. En caso de que no se hayan detectado NCU, concluye el procedimiento y se libera el producto.

#### Representación gráfica

# Capítulo 2: Solución Propuesta

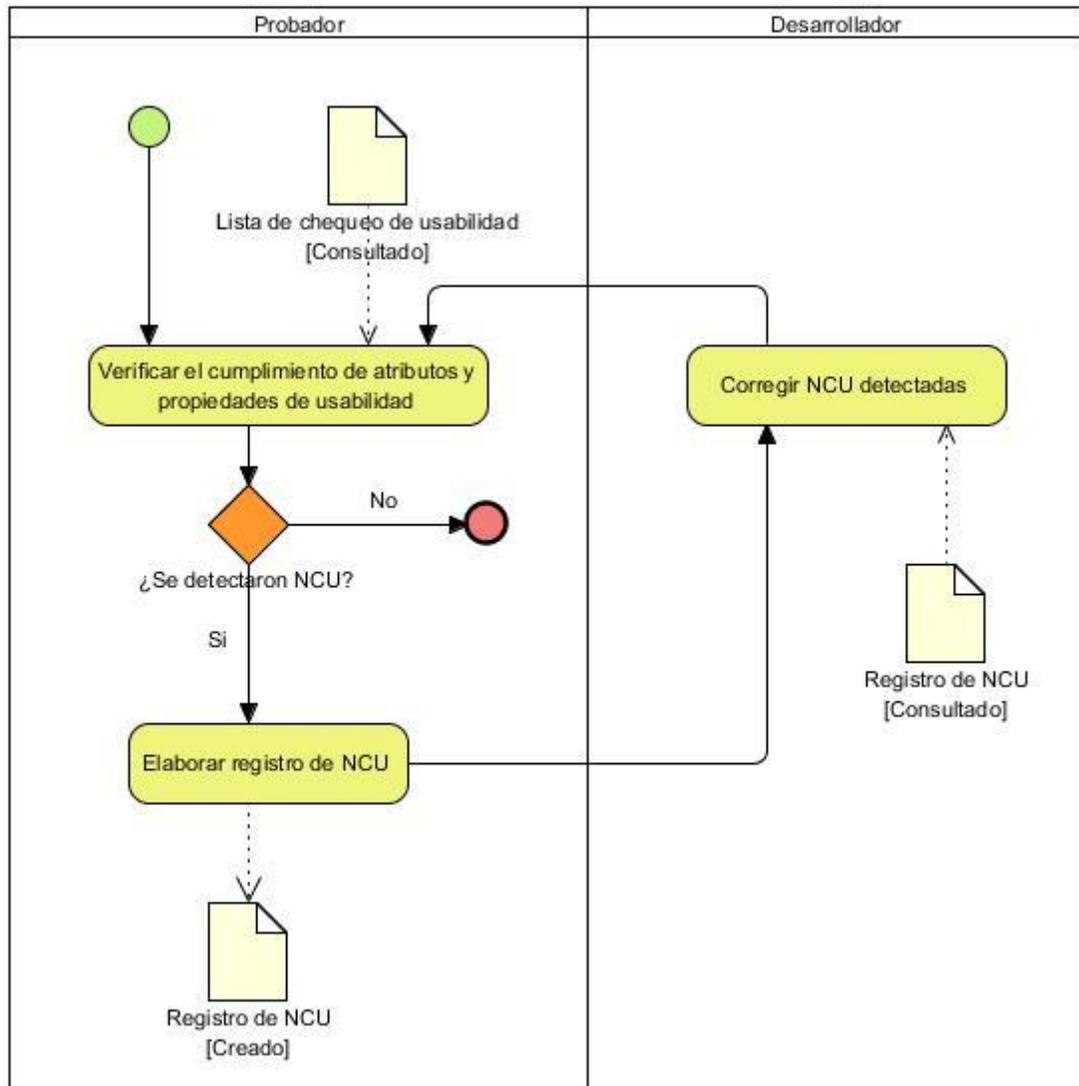


Figura 6 Fase: Pruebas de aceptación.

## 2.4 Herramienta USPA

USPA es una aplicación que surge a partir de la necesidad de informatizar el funcionamiento del marco de trabajo SAU para el presente trabajo. Este marco de trabajo consiste en un conjunto integrado de soluciones de diseño, tales como patrones y propiedades de usabilidad. Se puede utilizar para guiar e informar en la fase de diseño arquitectónico, al Arquitecto del sistema, sobre la usabilidad necesaria para que dicha arquitectura tenga un buen soporte de usabilidad.

## Capítulo 2: Solución Propuesta

---

La herramienta puede disminuir los costos de desarrollo mediante la reducción de la cantidad de problemas de usabilidad que deben ser corregidos durante las últimas etapas de desarrollo. USPA fue desarrollada por los autores de este trabajo y validada mediante pruebas de Caja Negra realizadas en el laboratorio de Calidad del CEIGE.

### 2.4.1 Características y requerimientos técnicos

La herramienta posee ciertas características que serán mostradas a continuación, al igual que los requerimientos técnicos que esta necesita para su funcionamiento.

#### Características

- Aplicación web
- Multiplataforma

#### Requerimientos

Sistema operativo:  Windows,  Linux,  Mac. (Cualquier versión).

Licencia: Gratis.

Idioma:  Español.

Tamaño: 27.1 MB.

Memoria Ram: Mínimo 128 MB.

Lenguaje de Implementación: Grails.

IDE: JetBrains IntelliJ IDEA v11.0 Ultimate Edition.

### 2.4.2 Uso de la herramienta USPA

A continuación se muestra una descripción de los pasos a seguir para trabajar con la herramienta USPA.

- Una vez ejecutada la aplicación, se muestra la pantalla de bienvenida de la misma (**Ver FIGURA 7**).

## Capítulo 2: Solución Propuesta

---



Figura 7 Pantalla de bienvenida de la herramienta USPA.

Pasado un tiempo determinado, la aplicación re-direcciona hacia la pantalla principal donde serán mostradas las diferentes opciones que brinda la herramienta. (Ver **FIGURA 8**).

## Capítulo 2: Solución Propuesta



Figura 8 Interfaz principal de USPA.

Para adicionar un escenario, se da clic en el botón Nuevo Escenario, se introduce el nombre del escenario, se le da un valor del 1 al 4 a los atributos de acuerdo a la importancia que se le desee otorgar y se presiona el botón crear. (Ver FIGURA 9)

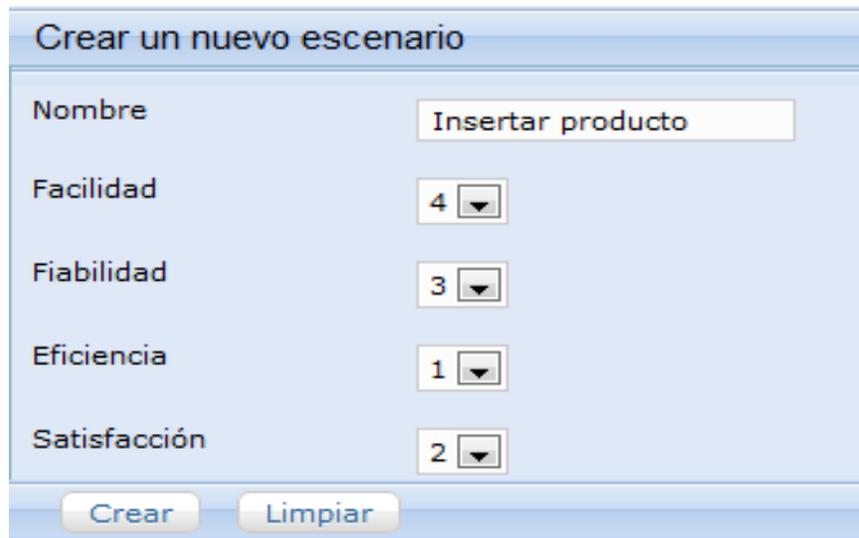
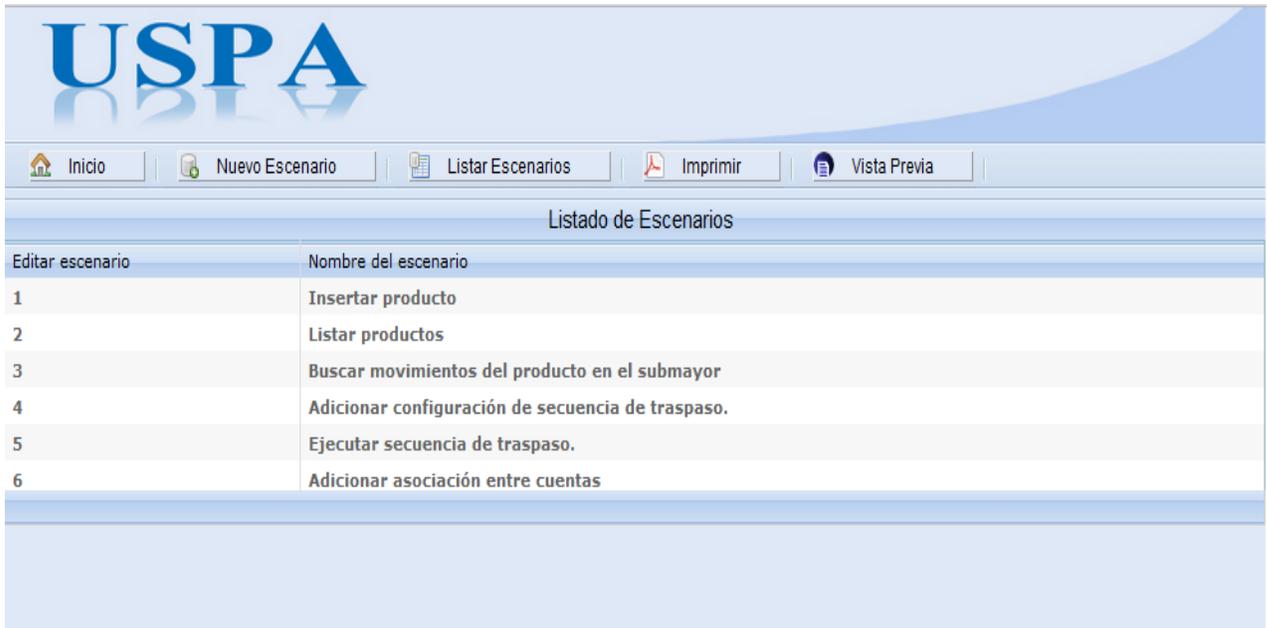
The image shows the 'Crear un nuevo escenario' (Create a new scenario) form. It has a title bar with the text 'Crear un nuevo escenario'. Below the title bar are five input fields, each with a label and a value: 'Nombre' with the value 'Insertar producto', 'Facilidad' with the value '4', 'Fiabilidad' with the value '3', 'Eficiencia' with the value '1', and 'Satisfacción' with the value '2'. At the bottom of the form are two buttons: 'Crear' and 'Limpiar'.

Figura 9 Interfaz Crear Escenario.

## Capítulo 2: Solución Propuesta

Después de crear todos los escenarios se listan los mismos en la página principal (Ver FIGURA 10).



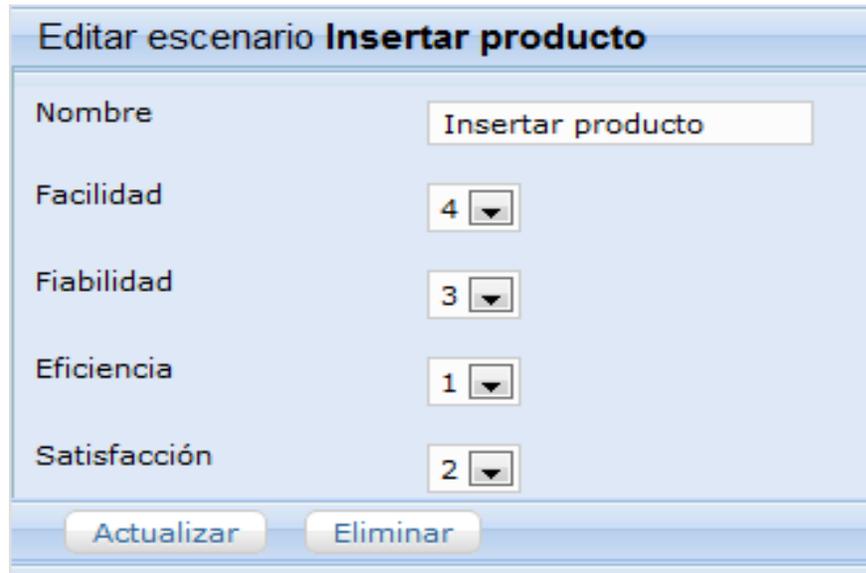
The screenshot shows the USPA web application interface. At the top, there is a navigation bar with the USPA logo and several menu items: Inicio, Nuevo Escenario, Listar Escenarios, Imprimir, and Vista Previa. Below the navigation bar, there is a section titled "Listado de Escenarios" which contains a table with two columns: "Editar escenario" and "Nombre del escenario". The table lists six scenarios with their respective identifiers and names.

Editar escenario	Nombre del escenario
1	Insertar producto
2	Listar productos
3	Buscar movimientos del producto en el submayor
4	Adicionar configuración de secuencia de traspaso.
5	Ejecutar secuencia de traspaso.
6	Adicionar asociación entre cuentas

Figura 10 Interfaz Listado de Escenarios.

Cada escenario contiene un identificador, si se quiere editar un escenario se da clic encima de su identificador, e inmediatamente, se muestra una página de edición en la cual se puede modificar o eliminar el escenario seleccionado (Ver FIGURA 11).

## Capítulo 2: Solución Propuesta

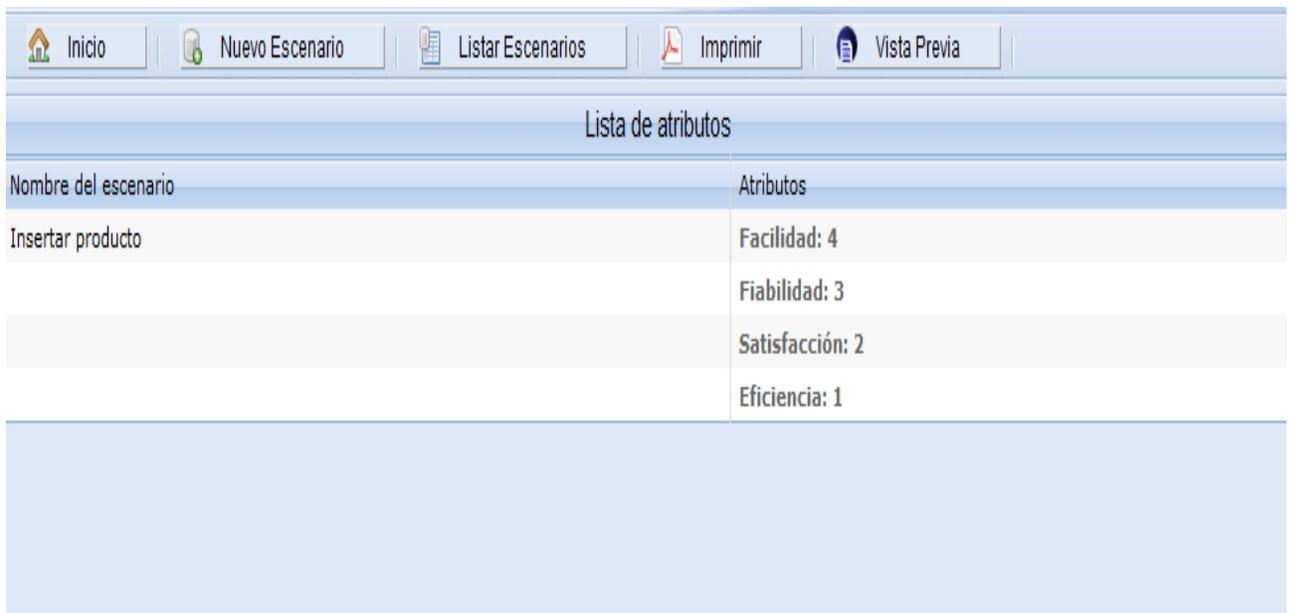


Editar escenario **Insertar producto**

Nombre	<input type="text" value="Insertar producto"/>
Facilidad	4 ▼
Fiabilidad	3 ▼
Eficiencia	1 ▼
Satisfacción	2 ▼

Figura 11 Editar Escenario

Para acceder a los atributos de un escenario se da clic en el nombre del mismo y estos son mostrados según el orden de prioridad (**Ver FIGURA 12**).



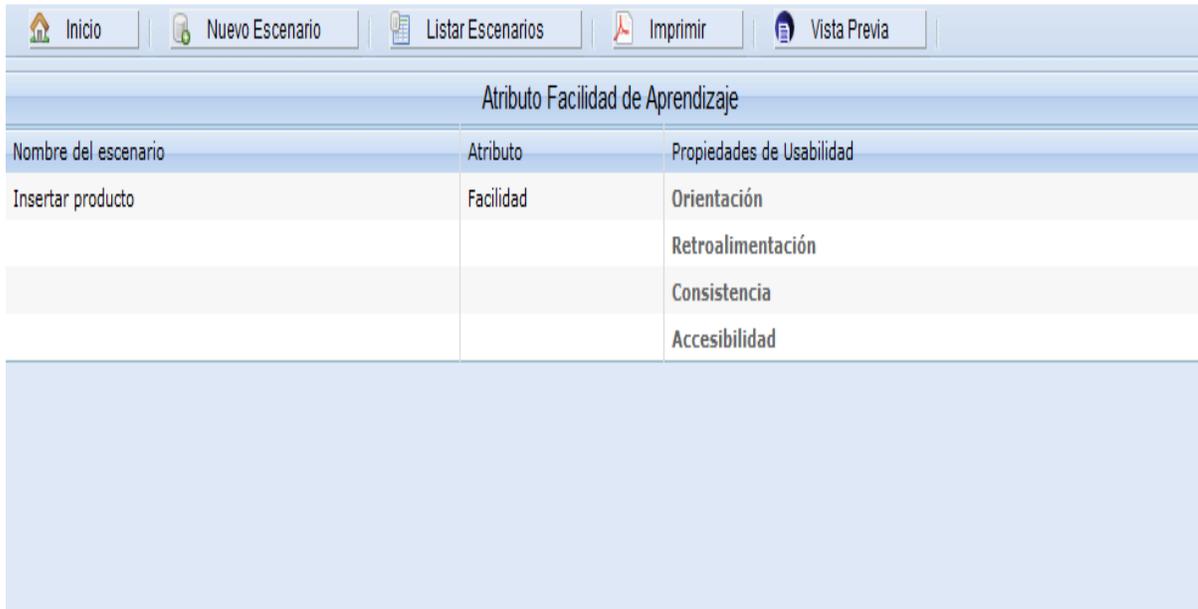
Inicio | Nuevo Escenario | Listar Escenarios | Imprimir | Vista Previa

Lista de atributos	
Nombre del escenario	Atributos
Insertar producto	Facilidad: 4
	Fiabilidad: 3
	Satisfacción: 2
	Eficiencia: 1

Figura 12 Interfaz Listado de Atributos del Escenario.

Al seleccionar un atributo se accede a sus propiedades (**Ver FIGURA 13**).

## Capítulo 2: Solución Propuesta



The screenshot shows a software interface with a menu bar at the top containing 'Inicio', 'Nuevo Escenario', 'Listar Escenarios', 'Imprimir', and 'Vista Previa'. Below the menu bar is a table titled 'Atributo Facilidad de Aprendizaje'. The table has three columns: 'Nombre del escenario', 'Atributo', and 'Propiedades de Usabilidad'. The first row shows 'Insertar producto' in the first column, 'Facilidad' in the second, and a list of usability properties in the third: 'Orientación', 'Retroalimentación', 'Consistencia', and 'Accesibilidad'.

Atributo Facilidad de Aprendizaje		
Nombre del escenario	Atributo	Propiedades de Usabilidad
Insertar producto	Facilidad	Orientación Retroalimentación Consistencia Accesibilidad

Figura 13 Propiedades del Atributo Priorizado.

Finalmente para acceder a los patrones de usabilidad que dan cumplimiento a una propiedad, se selecciona esa propiedad y se muestra un listado de los mismos (**Ver FIGURA 14**).



The screenshot shows the USP software interface with the 'USPA' logo at the top left. The menu bar is the same as in Figure 13. Below the menu bar is a table titled 'Orientación'. The table has four columns: 'Nombre del escenario', 'Atributo', 'Propiedades de Usabilidad', and 'Patrón de Usabilidad'. The first row shows 'Adicionar Persona' in the first column, 'Facilidad' in the second, 'Orientación' in the third, and a list of usability patterns in the fourth: 'Asistente', 'Ayuda Contextual', and 'Diferentes lenguajes'.

Orientación			
Nombre del escenario	Atributo	Propiedades de Usabilidad	Patrón de Usabilidad
Adicionar Persona	Facilidad	Orientación	Asistente Ayuda Contextual Diferentes lenguajes

Figura 14 Patrones de usabilidad de la propiedad seleccionada.

## Capítulo 2: Solución Propuesta

Cada patrón de usabilidad tiene su descripción a la cual se puede acceder seleccionando el patrón deseado (Ver FIGURA 15).



The screenshot shows the USPA application interface. At the top, the logo 'USPA' is displayed in blue. Below the logo is a navigation bar with buttons for 'Inicio', 'Nuevo Escenario', 'Listar Escenarios', 'Imprimir', and 'Vista Previa'. The main content area is titled 'Asistente' and contains a table with the following rows:

Asistente	
Descripción del patrón	La arquitectura tecnológica de la plataforma proveerá la capacidad de presentar a los usuarios información que explica cómo hacer las tareas rutinarias del sistema, proporcionando una guía paso a paso.
Relación Arquitectura de Software	Se necesita de provisión en la arquitectura para el componente asistente, que puede ser conectado a otros componentes relevantes, ese es el componente que activa las operaciones y el componente que recibe los datos coleccionado por el asistente.
Relación con las propiedades de la usabilidad	El patrón asistente ayuda con la guía al usuario, mostrandole al usuario cada paso para realizar el proceso deseado.
Patrón de Solución del escenario	En la figura se muestran los componentes que participantes en este escenario:  Interfaz: recoge la solicitud de ayuda guiada y la envía al componente Ayuda guiada (1) .Además, mostrará la información de ayuda que recibe de la Ayuda guiada (2). Ayuda guiada: muestra una guía de ayuda para la solicitud de que se ha descrito por la ayuda (2). Esta ayuda puede variar de una solicitud de recorrido antes registrado, a un recorrido interactivo, que implica el desarrollo de una aplicación independiente. Si la ayuda no se almacena internamente en este componente, esta ayuda es proporcionada por cualquier otra parte del sistema a través del flujo de información del sistema (3). Sistema: este es un componente opcional y representa parte de sistema en el que se almacena la ayuda si la Ayuda Guiada no almacena la información internamente. El sistema, por lo tanto, tiene la responsabilidad de proporcionar la Ayuda Guiada a través de (3).
Restricciones del patrón solución	El rendimiento del sistema estará mejor si los archivos de ayuda se almacenan en el módulo estándar de ayuda más que en cualquier otra parte del sistema, ya que requiere menos interacciones.
Atributos de usabilidad que impacta	Eficiencia,Facilidad,Fiabilidad
Beneficios de Usabilidad	La disposición de una gira dará la orientación del usuario y mejorar la gestión de errores, tanto en la detección de errores y corrección de errores.
Ejemplo	El asistente usado por la mayoría de los programas de Windows, guía a los usuarios a través de varias opciones para la instalación. 

Figura 15 Descripción de un patrón de usabilidad

USPA consta de una serie de acciones a las cuales se puede acceder desde cualquier parte de la aplicación como son: listar escenarios (re-direcciona a la página principal), vista previa (muestra un

## Capítulo 2: Solución Propuesta

ejemplo de cómo quedará el documento generado por la aplicación antes de su creación), un botón inicio (re-direcciona a la página de bienvenida) y un botón imprimir (genera un documento con el nombre del escenarios, el o los atributos más priorizados, las propiedades y patrones de usabilidad). (Ver FIGURA 16)



Escenarios	Atributos	Propiedades	Patrones
Insertar producto	Facilidad	Retroalimentación	Indicación de estado
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes
		Consistencia Accesibilidad	Vista previa Multitarea
Listar productos	Facilidad	Retroalimentación	Indicación de estado
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes
		Consistencia Accesibilidad	Vista previa Multitarea
Buscar movimientos del producto en el submayor	Fiabilidad	Minimizar la carga cognitiva	Modelo de flujo Asistente Perfil del usuario
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes
		Gestión de errores Consistencia	Historial de Registro Cancelar Vista previa Deshacer Reutilización de la información Vista previa
Adicionar configuración de secuencia de traspaso.	Facilidad	Retroalimentación	Indicación de estado
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes
		Consistencia Accesibilidad	Vista previa Multitarea
Ejecutar secuencia de traspaso.	Fiabilidad	Minimizar la carga cognitiva	Modelo de flujo Asistente Perfil del usuario
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes

Figura 16 Listado de Patrones de cada escenario de uso.

## Capítulo 2: Solución Propuesta

---

### 2.5 Conclusiones del Capítulo

En este capítulo se realizó la propuesta de un procedimiento para evaluar la usabilidad en el diseño arquitectónico de los proyectos del CEIGE.

Este se compuso de tres fases, mediante las cuales se identificarán los requisitos de usabilidad que debe tener un sistema, de los cuales se derivan los atributos y propiedades de la usabilidad a ser incorporados en el diseño arquitectónico mediante los patrones de usabilidad.

Las fases organizativas que requiere el procedimiento propuesto se ajusta sin mayores dificultades a las fases existente en el proceso de desarrollo del CEIGE.

El procedimiento propuesto incidirá en la reducción de costos de producción y de mantenimiento, mejorará la imagen corporativa del centro, será aplicable para los sistemas de gestión del CEIGE.

La herramienta USPA, que fue propuesta a utilizar en el procedimiento planteado, permitirá generar un listado de patrones y propiedades de usabilidad para ser incorporado en la arquitectura. De dicha herramienta, se especificó el uso, a modo de manual de usuario para facilitar el entendimiento del trabajo con la misma, por parte del Arquitecto del sistema.

# Capítulo 3: Validación de la Propuesta de Solución

---

## CAPÍTULO 3: VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN

### 3.1 Introducción al Capítulo

La validación de la solución propuesta se realizará mediante la aplicación del procedimiento en los subsistemas Inventario y Costo y Procesos del proyecto Cedrux v1.0 del CEIGE. Para lograr evaluar correctamente la usabilidad en la arquitectura del software de estos subsistemas, se realizarán todas las actividades definidas en el procedimiento, participarán los roles previstos y se generarán los artefactos pertinentes. Como parte de la aplicación del procedimiento, se llevarán a cabo actividades referentes al método de evaluación SALUTA y se hará uso de la herramienta USPA.

### 3.2 Descripción del subsistema Inventario

El proyecto abarcará la informatización de los distintos procesos que componen dicho sistema, gestionando los documentos primarios que se utilizan para el control de las existencias, así como los movimientos del producto dentro del almacén.

El desarrollo del sistema para el control de los medios materiales almacenados, pretende proporcionar innumerables beneficios, entre ellos la rapidez en la obtención de resultados, almacenamiento de grandes volúmenes de información y facilidades para encontrar información adecuada y/o actualizada por parte de los especialistas que interactúen con el sistema.

El objetivo del proyecto es informatizar los distintos procesos que componen dicho sistema, gestionando los documentos primarios que se utilizan para el control de las existencias, así como los movimientos del producto dentro del almacén: Apertura, Recepción de productos, Despacho de productos, Inventario físico, Ajuste de inventario, Baja de productos en el almacén, Conciliación entre almacenes.

### 3.3 Descripción del subsistema Costo y procesos

La Gestión de Costos consiste en el registro, clasificación, resumen y presentación de los costos de la entidad. Contiene las funcionalidades para el trabajo con los nomencladores de centros de Costos, Grupos Presupuestarios, Elementos de gastos y Objetos del Gastos, las configuraciones de cuentas de gastos tanto patrimoniales como presupuestadas y las diferentes asociaciones que se puedan establecer entre ellos.

## Capítulo 3: Validación de la Propuesta de Solución

El proyecto tiene como objetivo desarrollar un software para la gestión de costos de las entidades cubanas que permita realizar recuperaciones sobre los costos incurridos por la entidad

### 3.4 Aplicación del procedimiento en los subsistemas Inventario y Costo y procesos.

#### 3.4.1 Fase: Requisitos

##### Definir requisitos de usabilidad

Conjuntamente con los Analistas del sistema, se identificaron como requisitos de usabilidad de los subsistemas Inventario y Costo y procesos. (Ver **Tabla 7**)

Tabla 7 Listado de priorización de requisitos de usabilidad del producto

Selección (X)	Requisitos funcionales de usabilidad	Descripción	
X	Alerta	Req 1	Si ocurre un error la aplicación debe notificarle al usuario para que el mismo decida que hará.
		Req 2	Las alertas deben emitirse por medio de mensajes visibles que el usuario puede ver.
	Deshacer	Req 1	Si el usuario comete un error entrando datos o quiere volver a los valores por defectos que estaban cuando se realizó una petición, la aplicación debe permitirle al usuario retornar a dichos valores o poder ver los valores entrados con anterioridad.
	Diferentes lenguajes	Req 1	Capacidad del software para interactuar con los usuarios en diferentes idiomas.

## Capítulo 3: Validación de la Propuesta de Solución

X	Indicación del estado	<b>Req 1</b>	Los usuarios estén informados de la situación actual del sistema.
X	Validación de campo	<b>Req 1</b>	Cuando un usuario rellena en varios campos, el sistema puede validar estos campos por completo cuando se envían los datos (validación de formularios) o de forma individual cada vez que se llena un campo (validación de campo).
	Asistente	<b>Req 1</b>	Presenta al usuario una secuencia estructurada de pasos para llevar a cabo una tarea, proporcionando al usuario una guía para cada paso. El usuario puede volver atrás y cambiar cualquiera de las etapas anteriores en el proceso.
	Ayuda Contextual	<b>Req 1</b>	Ayuda a lo que el usuario está haciendo actualmente y proporciona información relevante para la finalización de la tarea en cuestión.
	Modelo de flujo de trabajo	<b>Req 1</b>	El sistema proporciona a los usuarios una serie de herramientas o acciones para la ejecución de una tarea específica en un conjunto de datos, antes de enviar estos datos a la siguiente persona en la cadena de flujo de trabajo que se ejecutará una tarea diferente.

## Capítulo 3: Validación de la Propuesta de Solución

	Historial de registro	<b>Req 1</b>	El sistema registra las acciones realizadas por los usuarios, lo que les permite ver el registro de lo que han hecho.
X	Vista previa	<b>Req 1</b>	El sistema permite a los usuarios ver los resultados de una acción antes de ejecutar esta acción.
	Perfil de usuario	<b>Req 1</b>	El sistema crea y guarda perfiles de usuario para poder recuperar los atributos específicos del sistema asociado con un usuario cada vez que este usuario utiliza el sistema.
X	Cancelar	<b>Req 1</b>	Los usuarios pueden cancelar lo que están haciendo, si se dan cuenta que ellos han hecho algo mal antes de un estado de error se alcanza.
	Multi –tarea	<b>Req 1</b>	La situación en la que el sistema (y el usuario) puede manejar varias tareas al mismo tiempo, se cambia de una tarea a otra.
X	Acción sobre varios objetos	<b>Req 1</b>	El sistema permite a los usuarios seleccionar varios objetos en el que se ejecuta una y la misma acción al mismo tiempo.
	Agregación de comandos	<b>Req 1</b>	El sistema permite a los usuarios crear y guardar una secuencia de comandos

## Capítulo 3: Validación de la Propuesta de Solución

			o acciones que pueden ejecutarse en diferentes objetos.
X	Reutilización de la información	<b>Req 1</b>	Este patrón permite a los usuarios mover o copiar datos de forma manual o automática de una parte del sistema, lo que permite su reutilización.

### 3.4.2 Fase: Diseño

#### 3.4.2.1 Creación del perfil de uso

Para crear el perfil de uso, se crearon seis escenarios de uso, priorizando en cada uno de ellos un atributo de usabilidad en un rango de 1 a 4, donde el atributo que obtuvo valor 4 fue el más significativo en ese escenario. Esta actividad fue llevada a cabo en los subsistemas Inventario y Costo y procesos, a continuación se muestra un ejemplo de los resultados de dicha actividad en un escenario del subsistema Inventario. (Ver TABLA 8)

Tabla 8 Escenario de Uso: Insertar Producto

Subsistema	Usuarios	Escenario	Satisfacción	Facilidad de aprendizaje	Eficiencia	Fiabilidad
Inventario		Insertar producto	El usuario siente que tiene el control del sistema.	Facilidad con que el usuario realiza tareas en el sistema.	Tiempo de respuesta del sistema en cada tarea.	No deben existir errores en la realización de las tareas.
	<b>Perfil de usuario</b>		1	4	3	2
		<b>Requisitos de usabilidad.</b>				

Para la realización de los demás escenarios, las tareas similares se combinaron y ciertas tareas, con frecuencias de ejecución baja, no se tuvieron en cuenta. Se trató de minimizar el número de escenarios, sin perder la representatividad de la usabilidad que requiere el sistema.

# Capítulo 3: Validación de la Propuesta de Solución

Tabla 9 Escenarios seleccionados.

No.	Proyecto	Usuario	Escenario	Eficiencia	Facilidad de aprendizaje	Fiabilidad	Satisfacción
1	Inventario	Usuario del sistema	Insertar producto	1	4	3	2
2	Inventario	Usuario del sistema	Listar productos	1	4	2	3
3	Inventario	Usuario del sistema	Buscar movimientos del producto en el submayor de inventario	1	3	4	2
4	Costo y Procesos	Usuario del sistema	Adicionar configuración de secuencia de traspaso.	1	4	3	2
5	Costo y Procesos	Usuario del sistema	Ejecutar secuencia de traspaso.	3	2	4	1
6	Costo y Procesos	Usuario del sistema	Adicionar asociación entre cuentas de gastos y centros de costos.	3	2	4	1

### 3.4.2.2 Descripción de la arquitectura

En el siguiente paso se analizaron los diseños de la arquitectura del software antes de la aplicación del procedimiento y se identificaron los patrones de usabilidad que, de cierta forma, se habían implementado en el sistema. De manera general, se puede decir que los subsistemas seleccionados presentan

# Capítulo 3: Validación de la Propuesta de Solución

elementos de usabilidad, los cuales, por sus características, se agruparon según los patrones de usabilidad que daban respuesta a estos elementos en colaboración con los arquitectos del sistema. Ya que estos elementos no fueron edificados ni concebidos como patrones de usabilidad durante el diseño arquitectónico se hacía difícil medirlos puesto que no estaban estandarizados en el sistema. (Ver FIGURA 17)



Figura 17 Descripción de la usabilidad en la Arquitectura.

### 3.4.2.3 Evaluación de escenarios utilizando la herramienta USPA

Utilizando la herramienta USPA se identificaron qué propiedades y patrones de usabilidad deben ser implementados para dar respuesta al atributo priorizado en cada escenario. Este análisis es importante para la correcta toma de decisiones durante el diseño arquitectónico. Siguiendo el ejemplo para el subsistema Inventario, en el escenario Insertar productos, el atributo de usabilidad priorizado en este caso fue la facilidad de aprendizaje. A continuación, mediante el uso de la herramienta USPA, se muestran qué propiedades y patrones de usabilidad debían de haber sido implementados para que el sistema final cumpliera con esos atributos. (Ver FIGURA 18)

# Capítulo 3: Validación de la Propuesta de Solución

Escenarios	Atributos	Propiedades	Patrones
Insertar producto	Facilidad	Retroalimentación	Indicación de estado
		Orientación	Asistente Ayuda contextual Diferentes Lenguajes
		Consistencia	Vista previa
		Accesibilidad	Multitarea

Figura 18 Listado de patrones de usabilidad a incorporar en la arquitectura.

Fue importante analizar si el patrón de usabilidad es asimilado por la lógica del negocio (el concepto en base al cual funciona), por tanto, se evidenció la interacción del patrón de usabilidad junto con la lógica de negocio, para asegurar que el sistema sea usable.

Como resultado de un análisis similar de cada patrón descrito por USPA, la siguiente matriz muestra qué patrones y propiedades deben tenerse en cuenta para el diseño arquitectónico de los escenarios. (Ver **TABLA 10**)

Tabla 10 Matriz Escenarios /patrones y propiedades de usabilidad.

		Escenarios											
		Sin Procedimiento						Con Procedimiento					
Propiedades de Usabilidad		1	2	3	4	5	6	1	2	3	4	5	6
Propiedades de Usabilidad	Minimizar la carga cognitiva					X		X				X	X
	Accesibilidad												
	Control explícito del usuario				X				X		X		
	Orientación												
	Adaptabilidad							X				X	
	Consistencia												

## Capítulo 3: Validación de la Propuesta de Solución

	Gestión de errores		X		X	X	X	X		X		X	X	X	X	
	Retroalimentación		X	X	X	X				X	X	X	X			
<b>Patrones de Usabilidad</b>	Ayuda contextual															
	Asistente		X							X						
	Deshacer												X		X	
	Vista previa													X		
	Historial de usuario									X				X		
	Validación de datos		X		X	X	X	X		X		X	X	X	X	X
	Cancelar					X					X	X	X			
	Indicación de estado		X	X	X	X	X	X		X	X	X	X	X	X	X
	Multi-Tareas															
	Reutilización de la información															
	Agregación de objetos									X						X

### 3.4.2.4 Interpretación de resultados

A continuación se presenta una tabla resumen con la cantidad de patrones y propiedades de usabilidad por cada escenario y el soporte de cada escenario en la arquitectura de los subsistemas Inventario y

## Capítulo 3: Validación de la Propuesta de Solución

Costo y procesos. El soporte fue expresado en una escala de tres niveles: + (Soporte alto), +/- (Soporte medio) y - (Soporte bajo). (Ver TABLA 11)

Tabla 11 Soporte de usabilidad para cada escenario.

Etapas	Proyecto	Escenarios	Número de Propiedades (Cant actual/ Cant total)	Número de patrones (Cant actual/ Cant total)	Soporte de la usabilidad en la arquitectura + (Soporte alto ) +/- (Soporte medio) - (Soporte bajo)
Sin procedimiento	Inventario	1	2/4	3/8	-
		2	1/4	1/8	-
		3	2/4	2/8	-
	Costo y Procesos	4	3/4	3/8	-
		5	2/4	2/8	-
		6	1/4	2/8	-
Con procedimiento	Inventario	1	4/4	5/8	+
		2	2/4	2/8	-
		3	2/4	3/8	-
	Costo y	4	3/4	4/8	+/-

## Capítulo 3: Validación de la Propuesta de Solución

	Procesos	5	3/4	4/8	+/-
		6	2/4	4/8	+/-

En las **Figuras 19 y 20** se expone gráficamente la comparación de los indicadores Número de patrones y propiedades de usabilidad, respectivamente.

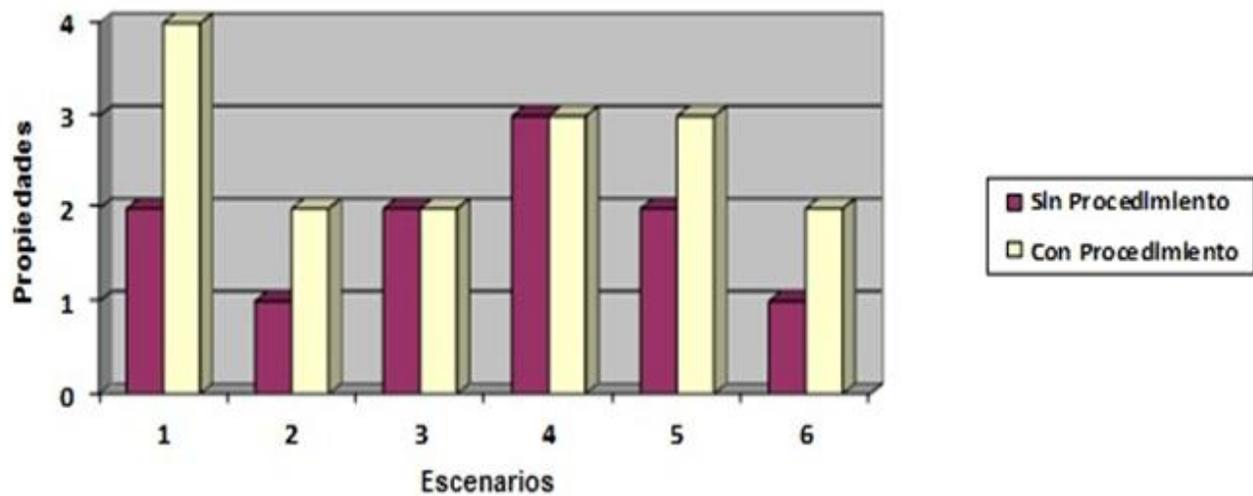


Figura 19 Número de propiedades de usabilidad por cada escenario.

## Capítulo 3: Validación de la Propuesta de Solución

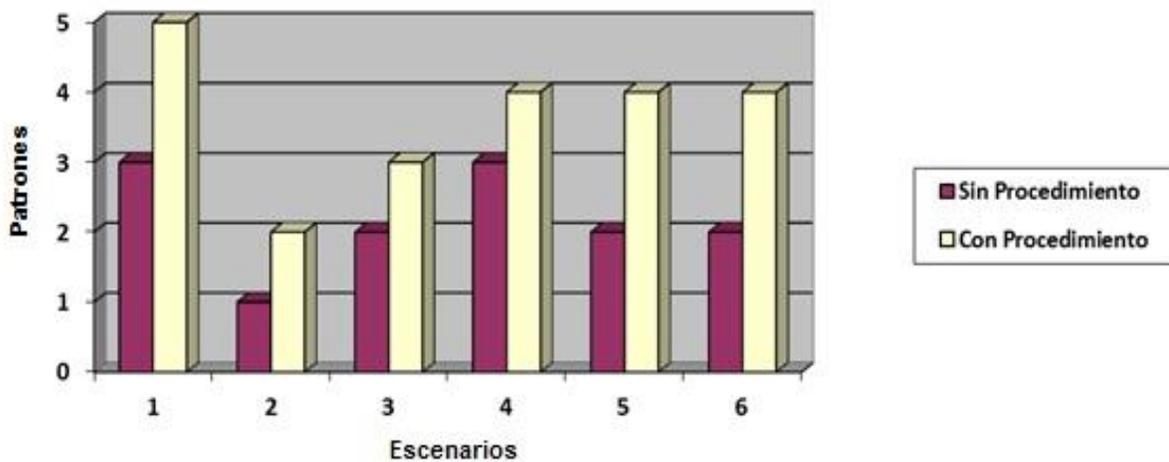


Figura 20 Número de patrones de usabilidad por cada escenario.

En la **Tabla 12** se presenta los resultados de las evaluaciones para el total de escenarios, mostrando el número de escenarios analizados, y el resultado del soporte de la usabilidad, en función de cuántos escenarios tienen soporte bajo, medio y alto, antes y después de aplicar el procedimiento.

Tabla 12 Soporte de usabilidad para cada escenario.

Fases	Nro. de escenarios	Soporte bajo (-)	Soporte medio (+/-)	Soporte alto (+)
Sin el procedimiento	6	6	0	0
Con el procedimiento	6	2	3	1

En la siguiente figura se expone gráficamente la comparación del comportamiento antes mencionada. (**Ver FIGURA 21**)

## Capítulo 3: Validación de la Propuesta de Solución

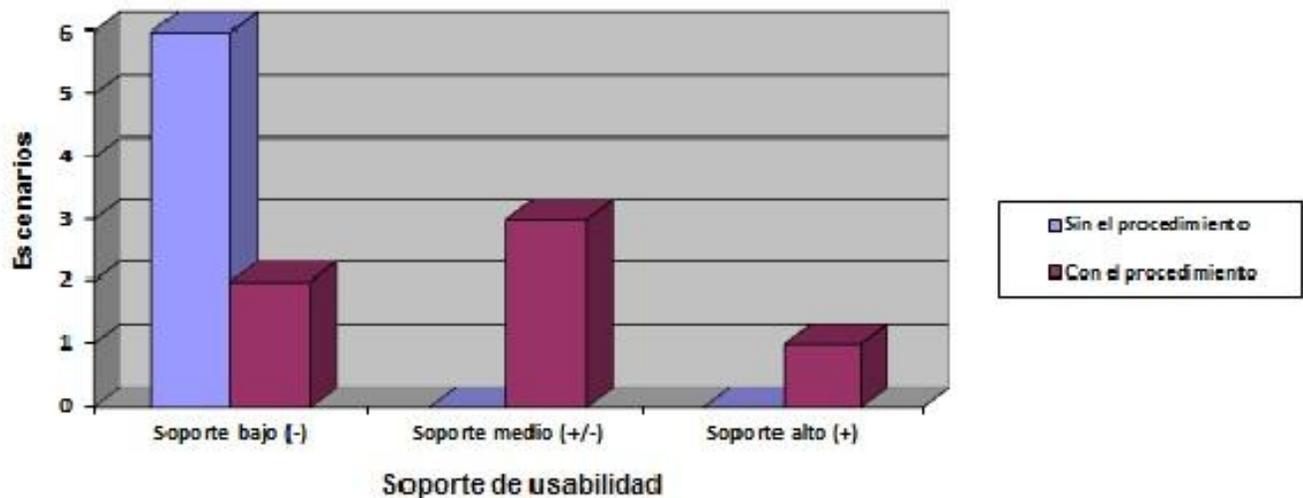


Figura 21 Soporte de la usabilidad.

Con la aplicación del procedimiento hubo un aumento en el número de atributos, propiedades y patrones de usabilidad que antes no se tenían en cuenta y, por consiguiente, un aumento del soporte de la usabilidad en la arquitectura de los subsistemas Inventario y Costo y procesos.

### 3.5 Determinar beneficios financieros económicos de la propuesta

Algunos de los beneficios identificados en la norma ISO TR 18529, y que la propuesta de solución trae consigo, es la disminución de los costos de desarrollo debido a cambios impactantes en el diseño arquitectónico.

La disminución de los costos por rediseño o el cambio radical de la arquitectura se ven expresados en los costos producidos durante la fase de pruebas, ante la petición del cliente de incorporar beneficios de usabilidad con impacto en la arquitectura. Esto es ocasionado porque durante la fase de diseño arquitectónico no se tuvieron en cuenta estos requisitos del cliente como patrones de usabilidad.

Unos de los problemas detectados en la fase de pruebas, por no tener en cuenta los requisitos del cliente y que evidencian un aumento en los costos de desarrollo son:

- El sistema no permite adicionar más de 40 productos a la misma vez a una hoja de apertura. Esta No Conformidad responde al patrón de usabilidad agregación de múltiples objetos y fue detectada en el subsistema Inventario de CEDRUX v1.0.

## Capítulo 3: Validación de la Propuesta de Solución

---

- Al realizar la búsqueda a partir de un criterio que no coincide con lo que aparece en la Base de Datos el sistema se queda buscando por mucho tiempo sin mostrar ningún mensaje que indique al usuario lo que está haciendo el sistema. Esta No Conformidad responde al patrón de usabilidad retroalimentación y fue detectada en el subsistema Costo y procesos de CEDRUX v1.0.

Se realizó un análisis con los desarrolladores para comprobar el incremento de los costos por el rediseño de la arquitectura debido a estas no conformidades, calculando el beneficio financiero total.

Estas funcionalidades requirieron de 5 días con dos desarrolladores, que reciben un salario diario de \$26.56 , lo que implicó un aumento de los costos en la fase de pruebas de \$265.6 CUP dado por la fórmula:

Costo Total= cant de personas \* cant de días\* salario diario

Por citar solo los dos ejemplos anteriores, el total de los beneficios financieros económicos aumentaron dado por la fórmula:

**Beneficio financiero total =  $\sum$  beneficios financieros de cada ahorro identificado**

**Beneficio financiero total = \$12876,288 + \$265.6= \$13141.888**

Si se hubiese tenido en cuenta la solución propuesta en esta investigación desde los inicios del desarrollo del sistema CEDRUX, se hubiesen disminuidos las NCU con impacto arquitectónico en las pruebas de aceptación y por consiguiente los costes de desarrollo.

### 3.6 Validación de la herramienta USPA mediante las pruebas de caja negra

La herramienta fue probada por el Departamento de Calidad del CEIGE donde se comprobó el buen funcionamiento de la misma y quedó avalada por el Acta de Liberación (**Ver Anexo 6**) expuesta.

Para la aplicación de estas pruebas se realizaron un total de 3 iteraciones empleándose en tiempo 4 horas efectivas de trabajo. La actividad estuvo dirigida por un jefe de pruebas hasta obtener un total de 0 no conformidades.

# Capítulo 3: Validación de la Propuesta de Solución

---

## 3.7 Conclusiones del Capítulo

En este capítulo se llevó a cabo la aplicación del procedimiento, definido en el capítulo anterior, en los subsistemas Inventario y Costo y procesos del producto Cedrux v1.0.

Se realizaron una serie de actividades donde se tuvo en cuenta la participación de los roles involucrados en las mismas. Se realizó la evaluación de la usabilidad haciendo uso de la herramienta USPA lo que permitió realizar dicha evaluación de modo más ágil y eficiente. Mediante la aplicación del procedimiento, en los subsistemas Inventario Y Costo y procesos, se demostró que hubo un aumento en el número de atributos, propiedades y patrones de usabilidad, que antes no se tenían en cuenta y que deben ser incorporados en el diseño, lo que evidenció un aumento del soporte de la usabilidad en la arquitectura de estos subsistemas.

Se realizó un análisis de los beneficios económicos del procedimiento donde se arrojó que la aplicación del procedimiento incide en la disminución de los costos de desarrollo.

# Conclusiones Generales

---

---

## CONCLUSIONES GENERALES

En el presente trabajo de diploma, se define y describe un procedimiento con el fin de evaluar la usabilidad en el diseño arquitectónico de los proyectos del CEIGE, haciendo uso de la herramienta USPA, desarrollada por los autores de la investigación, para el manejo de patrones.

Con el fin de cumplir los objetivos planteados en este trabajo, se llevaron a cabo una serie de tareas de las que se puede concluir, de forma general, lo siguiente:

- Las actividades definidas en las fases de requisitos y diseño, en el procedimiento, pueden ser incorporadas al proceso de desarrollo del CEIGE.
- A través de la definición de los roles participantes y la descripción las actividades y los artefactos a desarrollar para llevar a cabo la evaluación de la usabilidad en los proyectos del CEIGE, se logró mayor completitud y entendimiento de la solución propuesta.
- El manejo de los patrones de usabilidad mediante el uso de la herramienta USPA, permitió optimizar el proceso de evaluación de la usabilidad.
- El procedimiento propuesto se realizó acorde a las necesidades del centro, y busca una mejora de la usabilidad durante el diseño arquitectónico de los proyectos del CEIGE.
- La validación del procedimiento demostró que hubo un aumento en el número de atributos, propiedades y patrones de usabilidad que fueron añadidos al diseño, logrando así un mejor soporte de usabilidad.
- La propuesta fue evaluada en los subsistemas Inventario y Costo y procesos del Cedrux v1.0, obteniéndose buenos resultados.

## RECOMENDACIONES

Se recomienda:

- Poner en práctica la propuesta del procedimiento en los proyectos de desarrollo del CEIGE.
- Continuar evaluando la usabilidad en fases tempranas del desarrollo de los productos del CEIGE.
- Proponer el resultado de esta investigación como base referencial o material auxiliar en el desarrollo de futuros trabajos.

## BIBLIOGRAFÍA

- [1] ISO 9241-11
- [2] Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos, María Elena Alva Obeso, Oviedo, Marzo de 2005.
- [3] Casanovas, Joseph. Usabilidad y arquitectura del software, Septiembre de 2004.
- [4] McCall et al, 1977, Arquitecturas de Software, Abril – 2004.
- [5] L. L. Constantine, L. A. D. Lockwood. Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design. Addison-Wesley, Nueva York, NY, 1999
- [6] J. Nielsen. Usability Engineering. AP Professional, 1993.
- [7] B. Shackel. "Usability – context, framework, design and evaluation". En Human Factors for Informatics Usability. pp 21-38. B. Shackel and S. Richardson (eds.). Cambridge University Press, 1991.
- [8] Bosert 1991, Los costes y la Usabilidad.
- [9] <http://creative.bsil.com/download/ROI-on-usability.pdf>
- [10] Lederer y Prassad 1992, Usability Quotes.
- [11] <http://portal.acm.org/citation.cfm?id=129632>
- [12] MacIntyre et al. 1990, UsabilityWorks Quotes.
- [13] <http://portal.acm.org/citation.cfm?id=83217>
- [14] Martin y McClure 1993; Pressman 1992
- [15] [http://highered.mcgraw-ill.com/sites/0072853182/information\\_center\\_view0/](http://highered.mcgraw-ill.com/sites/0072853182/information_center_view0/)
- [16] Folmer and Bosch, 2004, Architecting for usability: a survey.
- [17] Eelke Folmer, Jilles Van Gorp, Jan Bosch, and Software Process: Improvement and Practice, Volume 8, Issue 2, Pages 67-87, June 2003
- [18] Busschmann 1996, A Survey of Three Approaches to the Automation of Design Patterns.
- [19] Perzel, 99, Usability Patterns for Applications on the World Wide Web.

- [20] Larman, C. (2004). Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd. ed.). Prentice Hall Professional Technical Reference.
- [21] Ana M. Moreno y Maribel Sánchez-Segura, Patrones de Usabilidad: Mejora de la Usabilidad del Software desde el momento de Arquitectónico.
- [22] Guillermo M. Martínez de la Teja, USABILIDAD Y ACCESIBILIDAD EN WEB, Junio 2003.
- [23] Saz, Jesús Tramullas. Usabilidad y arquitectura de la información. 7 de febrero de 2006.
- [24] Sánchez-Segura, Ana M. Moreno y Maribel. Patrones de Usabilidad: Mejora de la Usabilidad del Software desde el momento de Arquitectónicos. s.l. : Facultad de Informática Universidad Politécnica de Madrid, España.
- [25] Ferreras, Hayser Jacquelin Beltré. APLICACIÓN DE LA USABILIDAD AL PROCESO DE DESARROLLO DE PÁGINAS WEB. S.I.: FACULTAD DE INFORMATICA UNIVERSIDAD POLITECNICA DE MADRID, Septiembre 2008.
- Arciniegas, J., Contribution to quality-driven evolutionary software development for service oriented architecture. Universidad Politécnica, Madrid, Ph.D. degree thesis, 2006.
  - Arciniegas, J., Fernández, V., Tulande, A., Hormiga, A. Urbano, F., Collazos, C., Proceso de requerimiento y análisis para la definición de la arquitectura desde la perspectiva de usabilidad para el desarrollo de una aplicación en la Web., Revista Avances n Sistemas e Informática, Vol 6, No. 2, Septiembre, 2009, pp. 205 - 210.
  - Bosch, J., Design and use of software architectures: adopting and evolving a product line approach., Pearson Education, Addison-Wesley, 2000.
  - Documents in computing and information science, 5 (4), 2000, pp. 243-250.
  - Eduardo Manchón, Qué es la usabilidad [http://www.ainda.info/que\\_es\\_usabilidad.htm](http://www.ainda.info/que_es_usabilidad.htm) Último acceso: 2008.
  - Ergonomic requirements for office work with visual display terminals (VDTs)-Part 11: Guidance on usability. 1998.

- Fernández, V., Hormiga, M., Tulande, A., Marco de referencia centrado en la arquitectura para la mejora de características de usabilidad en el desarrollo de aplicaciones web construidas por MIPYMES., Undergraduate thesis, Universidad del Cauca, 2008.
- Ferré, X., Juristo, N., Windl, H., Constantine, L., Usability basics for software developers. IEEE Software,
- ISO 25010, International Standard.
- ISO 9241-11., International Standard.
- ISO/IEC 9126-1., International standard, software engineering-product quality-Part 1: Quality model., 2001.
- Software product Quality Requirements and Evaluation (SQuaRE): Quality model., 2007. Vol. 18, No. 11, 2001, pp. 22-30.