



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

PROYECTO DE TITULACIÓN

**PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS
E INFORMÁTICA**

AUTOR: MOSQUERA ALAVA YANINA ANDREA

**TEMA: “ESTUDIO DE LA NOTACIÓN CONCURTASKTREES PARA EL
ANÁLISIS DE TAREAS EN EL DESARROLLO DE LA INTERACCIÓN
HUMANO-COMPUTADOR Y SU APLICACIÓN EN EL ENTORNO CTTE.”**

DIRECTOR: ING.DANILO MARTÍNEZ

SANGOLQUÍ, AGOSTO 2015

CERTIFICADO

Ing. Danilo Martínez, MSc.

CERTIFICAN

Que el trabajo titulado “ESTUDIO DE LA NOTACIÓN CONCURTASKTREES PARA EL ANÁLISIS DE TAREAS EN EL DESARROLLO DE LA INTERACCIÓN HUMANO-COMPUTADOR Y SU APLICACIÓN EN EL ENTORNO CTTE.”, realizado por la Srta. Yanina Andrea Mosquera Alava, ha sido guiado y revisado periódicamente y cumple en su totalidad con las normas establecidas por la Universidad de las Fuerzas Armadas – ESPE.

Sangolquí, 26 de agosto del 2015.



Ing. Danilo Martínez, MSc.

DIRECTOR

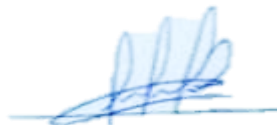
AUTORÍA

Yo, Yanina Andrea Mosquera Alava

DECLARO QUE:

El proyecto de grado denominado “ESTUDIO DE LA NOTACIÓN CONCURTASKTREES PARA EL ANÁLISIS DE TAREAS EN EL DESARROLLO DE LA INTERACCIÓN HUMANO-COMPUTADOR Y SU APLICACIÓN EN EL ENTORNO CTTE.” es de mi autoría y ha sido desarrollado en base a una investigación exhaustiva respetando derechos intelectuales de terceros e incorporando las referencias correspondientes en la bibliografía. En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, 26 de agosto del 2015.



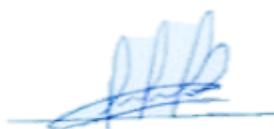
Yanina Andrea Mosquera Alava

AUTORIZACIÓN

Yo, Yanina Andrea Mosquera Alava

Autorizo a la Universidad de las Fuerzas Armadas – ESPE la publicación en la biblioteca virtual de la Institución del trabajo titulado “ESTUDIO DE LA NOTACIÓN CONCURTASKTREES PARA EL ANÁLISIS DE TAREAS EN EL DESARROLLO DE LA INTERACCIÓN HUMANO-COMPUTADOR Y SU APLICACIÓN EN EL ENTORNO CTTE.”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Sangolquí, 26 de agosto del 2015.



Yanina Andrea Mosquera Alava

DEDICATORIA

Dedico esta tesis a Dios, la Virgen de Guadalupe, quienes inspiraron mi espíritu y me llenaron de fuerzas para seguir adelante y no desmayar en los problemas que se presentaban.

A mi Madre Mirian Alava y mi Padre Luis Mosquera que es mi ángel en el cielo estoy muy agradecida por que me dieron vida, educación, apoyo y consejos, forjando mis valores, mis principios, mi carácter, mi empeño, mi perseverancia, mi coraje para conseguir mis objetivos.

A mis Hermanos Luis y Raúl, quienes siempre me brindaron su compañía, palabras de superación y un cariño necesario para este largo camino.

A mi Abuelita Magdalena Puyol, por su amor incondicional y por siempre inculcarme que siempre cada cosa con la bendición y guía de Dios es más fácil.

A todos ellos se los agradezco desde el fondo de mi alma por creer en mí.

AGRADECIMIENTO

Quisiera comenzar agradeciendo a ti mi Dios por bendecirme para llegar hasta donde he llegado, porque hiciste realidad este sueño anhelado.

A la UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE por darme la oportunidad de estudiar y ser una profesional.

A mi director de tesis, ING. Danilo Martínez, Ms por su esfuerzo y dedicación, quien con sus conocimientos, su experiencia, su paciencia y su motivación ha logrado en mí que pueda terminar mi tesis.

De igual manera agradecer a mi Mama Mirian Álava y mis hermanos que siempre me apoyaron y no dudaron en darme palabras de aliento y fueron una motivación muy fundamental en todo el proceso de estudios.

También me gustaría agradecer a mis profesores durante toda mi carrera profesional porque todos han aportado con un granito de arena a mi formación, y en especial a mi profesor ING. Mauricio Campaña por sus consejos, su enseñanza y más que todo por su amistad.

Son muchas las personas que desearía agradecer, algunas están aquí conmigo y otras en mis recuerdos y en mi corazón, sin importar en donde estén quiero darles las gracias por formar parte de mí, por todo lo que me han brindado y por todas sus bendiciones.

ÍNDICE GENERAL

CERTIFICADO	ii
AUTORÍA.....	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
RESUMEN.....	xiii
ABSTRACT.....	xiv
CAPÍTULO 1	1
INTRODUCCIÓN	1
1.1 ANTECEDENTES.....	1
1.2 PLANTEAMIENTO DEL PROBLEMA	3
1.3 JUSTIFICACIÓN	6
1.4 OBJETIVOS	7
1.4.1 Objetivo General	7
1.4.2 Objetivos Específicos.....	7
1.5 ALCANCE.....	8
CAPÍTULO 2.....	9
CONCURTASKTREES	9
2.1 DISEÑO CENTRADO EN EL USUARIO	9
2.2 USABILIDAD	9
2.3 RELACIÓN DE LA USABILIDAD Y EL DISEÑO DE LA INTERFAZ DE USUARIO.....	11
2.4 RELACIÓN ENTRE EL PROCESO DE DESARROLLO DE SOFTWARE Y EL PROCESO DE DESARROLLO DE INTERACCIÓN.....	12
2.5 ANÁLISIS DE TAREAS.....	16
2.6 NOTACIÓN CTT	17

2.7	CARACTERÍSTICAS CTT.....	18
2.8	OPERADORES	20
2.8.1	Interacción.....	21
2.8.2	Habilitación.....	21
2.8.3	Habilitación con transferencia de información	22
2.8.4	Tareas concurrentes.....	22
2.8.5	Sincronización.....	23
2.8.6	Desactivación	23
2.8.7	Independencia de tareas	24
2.8.8	Suspender-reanudar.....	24
2.9	HERRAMIENTA CTTE.....	25
CAPÍTULO 3.....		28
CONCURTASKTREES ENVIRONMENT		28
3.1	SINTAXIS CTTE	28
3.1.1	Relación Temporal Combinada	28
3.2	OPCIONES CTTE	28
3.2.1	Habilitar la opción "Modo Cooperativo"	32
3.2.2	Añadir nuevos roles.....	32
3.2.3	Centrarse en las "tareas de conexión"	33
3.2.4	Compatibilidad CTTE.....	34
3.2.5	Compatibilidad con MARIA.....	36
3.2.6	Descripción de Tarea Individual	36
CAPÍTULO 4.....		39
CASO DE ESTUDIO.....		39
4.1	PROTOTIPO BANCA VIRTUAL	39

CAPÍTULO 5.....	51
CONCLUSIONES Y RECOMENDACIONES.....	51
5.1 CONCLUSIONES	51
5.2 RECOMENDACIONES	52
BIBLIOGRAFÍA	54

ÍNDICE DE TABLAS

Tabla 1.....	34
--------------	----

ÍNDICE DE FIGURAS

Figura 1: Actividades típicas en un proceso centrado en el usuario	15
Figura 2: Relación entre actividades de Usabilidad y Esquema genérico de Actividades del Proceso de Desarrollo	16
Figura 3: Relación entre actividades de Usabilidad y Esquema genérico de Actividades del Proceso de Desarrollo	28
Figura 4: Características principales de CTTE	33
Figura 5: Interacción	35
Figura 6: Habilidad	35
Figura 7: Habilidad con transferencia	36
Figura 8: Tareas concurrentes	36
Figura 9: Sincronización	37
Figura 10: Desactivación.....	37
Figura 11: Independencia de Tareas	38
Figura 12: Suspender	38
Figura 13: Editor CTT.....	40
Figura 14: Comprobación Automática.....	43
Figura 15: Descripción Formal	44
Figura 16: Simulador de Tareas	45
Figura 17: Estadísticas	45
Figura 18: Modo Cooperativo	47
Figura 19: Nuevo Rol.....	47
Figura 20: Tareas de Conexión	48
Figura 21: Formatos disponibles para Guardar	49
Figura 22: Propiedades de una Tarea	50
Figura 23: Propiedades de una Tarea	52
Figura 24: Propiedades de una Tarea	55
Figura 25: Árbol de Identificación.....	56
Figura 26: Árbol de Prestamos.....	60
Figura 27: Árbol Actualización.....	62
Figura 28: Árbol de Tarjetas	63
Figura 29: Árbol de Perfil de Acceso.....	64

Figura 30: Árbol de Cerrar Sesión	65
Figura 29: Estadísticas del Modelo	65
Figura 30: Simulador del Modelo	66

RESUMEN

El presente proyecto tiene como objetivo realizar un estudio de la notación ConcurTaskTrees para el análisis de tareas en el desarrollo de la interacción humano-computador y su aplicación en el entorno CTTE (ConcurTaskTrees Environment).

Para lo cual se realizó una investigación detallada en donde se recopiló extractos importantes de artículos técnicos con respecto al tema, los cuales entrarán en un estudio de exclusión con el propósito de obtener información valedera y real.

Con el fin de que el proceso de desarrollo de software este centrado en el usuario, se hizo un análisis de tareas de una Banca Virtual cuyos resultados se ingresaron en la herramienta CTTE con el objetivo principal de visualizar el modelo de interacción que resulte más factible y amigable para el usuario.

El resto del trabajo de tesis, ha sido organizado de la siguiente manera: en el Capítulo 1 se describe las generalidades del proyecto, Capítulo 2 se detalla información sobre la Notación ConcurTaskTrees, Capítulo 3 se presenta todo acerca de la herramienta de modelado ConcurTaskTrees Environment, Capítulo 4 se muestra el prototipo de la banca virtual y Capítulo 5 conclusiones y recomendaciones.

PALABRAS CLAVE:

- CTT
- CTTE
- BANCA VIRTUAL
- USABILIDAD
- IPO

ABSTRACT

This project aims to conduct a study of the ConcurTaskTrees notation for task analysis in the development of human-computer interaction and its application in the environment CTTE (ConcurTaskTrees Environment).

For which a detailed investigation where important extracts compiled technical articles on the subject was made, which come in a study of exclusion in order to obtain valid and real information.

In order that the process of software development is centered on the user, a task analysis of a Virtual Bank whose results are entered in the CTTE tool with the main objective to visualize the interaction model that is most feasible became and user-friendly.

The rest of the thesis, has been organized as follows: in Chapter 1 an overview of the project, Chapter 2 describes information on ConcurTaskTrees notation, Chapter 3 details everything is presented about the modeling tool ConcurTaskTrees Environment, Chapter 4 virtual prototype shown banking and Chapter 5 Conclusions and recommendations.

KEY WORDS:

- CTT
- CTTE
- VIRTUAL BANKING
- USABILITY
- IPO

CAPÍTULO 1

INTRODUCCIÓN

1.1 ANTECEDENTES

El proceso de diseño debe estar centrado en el usuario para recoger sus necesidades y mejorar su utilización (Ferré X. 2003), por lo tanto el usuario debe tomar parte en el diseño y no ser mero espectador, (Garrido, 2008) algunos autores como Xavier Ferre en su paper (Ferré X. , Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo) ha realizado investigaciones que han profundizado este tema y han ayudado a incentivar a los desarrolladores a realizar sistemas usables.

El diseño de la interfaz es parte fundamental del proceso de desarrollo del software y debe ser considerado desde el principio para así asegurar sistemas usables. (Ferré X. , Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo).

La usabilidad ha venido evolucionando y se ha posicionado como un atributo de calidad (ISO9241) ,La usabilidad permite que un sistema sea usado con facilidad de forma eficiente y eficaz(Nielsen, 1994), es por esto que la interfaz de usuario puede definir el éxito o fracaso de una aplicación interactiva ,se puede mencionar que según los estudios realizados por MYERS (Myers & Rosson, 1992) a través de una encuesta hecha a desarrolladores, alrededor de un 48% del código de la aplicación está dedicado a la interfaz. .

El gran problema que ha existido en el proceso de desarrollo de software es que dicho proceso se centra más en las posibilidades tecnológicas que en sus usuarios (D. Hix, 1993), lo correcto sería involucrar a los usuarios desde las etapas tempranas del proceso de desarrollo de software.

Las actividades en un proceso centrado en los usuarios tienen como objetivo poner al usuario como su punto clave, estas actividades ayudan a obtener un

prototipo usable. Dentro de las actividades de análisis que permiten cumplir este objetivo se tiene al análisis de tareas. (Ferré X. , Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo)

El proceso de análisis de tareas se compone de dos partes muy importantes. La primera es la obtención de la información necesaria para comprender las actividades que realiza el usuario (fase de análisis). En tanto que la segunda fase, es proporcionar una representación de esta información sobre un modelo adecuado (fase de modelado) (Simarro, CoCTT: Modelado de tareas en un entorno colaborativo, 2007).

El análisis de tareas se emplea para que la aplicación sea amigable para los usuarios finales, la interfaz es una parte clave en el desarrollo de cualquier aplicación (Simarro, CoCTT: Modelado de tareas en un entorno colaborativo, 2007).

Existen notaciones para poder elaborar un diseño usable, entre las cuales se puede mencionar 3 notaciones que son HTA (Hierarchical Task Analysis), CTT (ConcurTaskTrees), MARIA (Universal Declarative Language).

ConcurTaskTrees (CTT) es una notación desarrollada por Fabio Paternò utilizada para modelar las tareas que un usuario puede llevar a cabo en una aplicación interactiva. Posee un conjunto rico de operadores que permite expresar de forma clara las relaciones temporales existentes entre las tareas y los usuarios encargados de llevarlas a cabo fomentando la interacción entre el usuario y el computador (Paternò, 2007).

Algunas de las principales ventajas de la notación ConcurTaskTrees son su estructura jerárquica que la hace muy intuitiva, su sintaxis gráfica que la hace sencilla de interpretar, y un conjunto rico de operadores que permite expresar de forma clara las relaciones temporales existentes entre las tareas y los usuarios encargados de llevarlas a cabo (Paternò, 2007).

ConcurTaskTrees Environment (CTTE) es una herramienta gratuita desarrollada en lenguaje Java, una de las características más positivas de esta herramienta es que permite guardar los modelos de tareas creados en varios formatos: como imagen en

formato JPG, en formato LOTOS, HTML Y XML (Simarro, CoCTT: Modelado de tareas en un entorno colaborativo, 2007).

1.2 PLANTEAMIENTO DEL PROBLEMA

Durante el desarrollo de una aplicación, se genera una brecha entre el equipo de desarrollo de software y el equipo de desarrollo de la interacción, acuerdo a los expuesto por (Seffah A, 2003).

Para (Ferre, 2005) existe un divorcio de la metodología de desarrollo y el proceso de interacción humano - computador. Ambos autores concuerdan que una solución sería que los dos procesos compartieran actividades comunes y de esta manera disminuir la brecha existente.

Esto ayudaría a mejorar la usabilidad y la experiencia de usuario en un el sistema desarrollado.

Xavier Ferre en su publicación Incorporación de Técnicas de Interacción Persona-Ordenador al Proceso de Desarrollo propone la incorporación del desarrollo de HCI dentro del proceso de desarrollo de Software, para la elaboración de sistemas usables.

El proceso para desarrollar un modelo de interacción usable está en función de varios aspectos, entre ellos el análisis de tareas, como se puede apreciar en la Figura1.

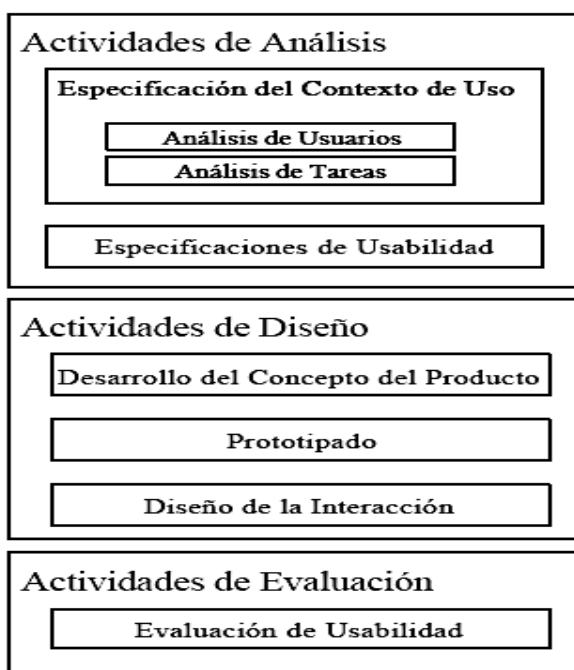


Figura 1: Actividades típicas en un proceso centrado en el usuario

Fuente: Ferré, X. (s.f.). Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo (p.4).

El presente proyecto está centrado en la fase de análisis de tareas que forma parte de Actividades típicas en un proceso centrado en el usuario que se relaciona con la comprensión del problema del proceso de desarrollo de Software. Como se puede observar en la Figura 2.

Uno de los inconvenientes en esta fase es la forma de representar las tareas de interacción de una forma sencilla y estandarizada.

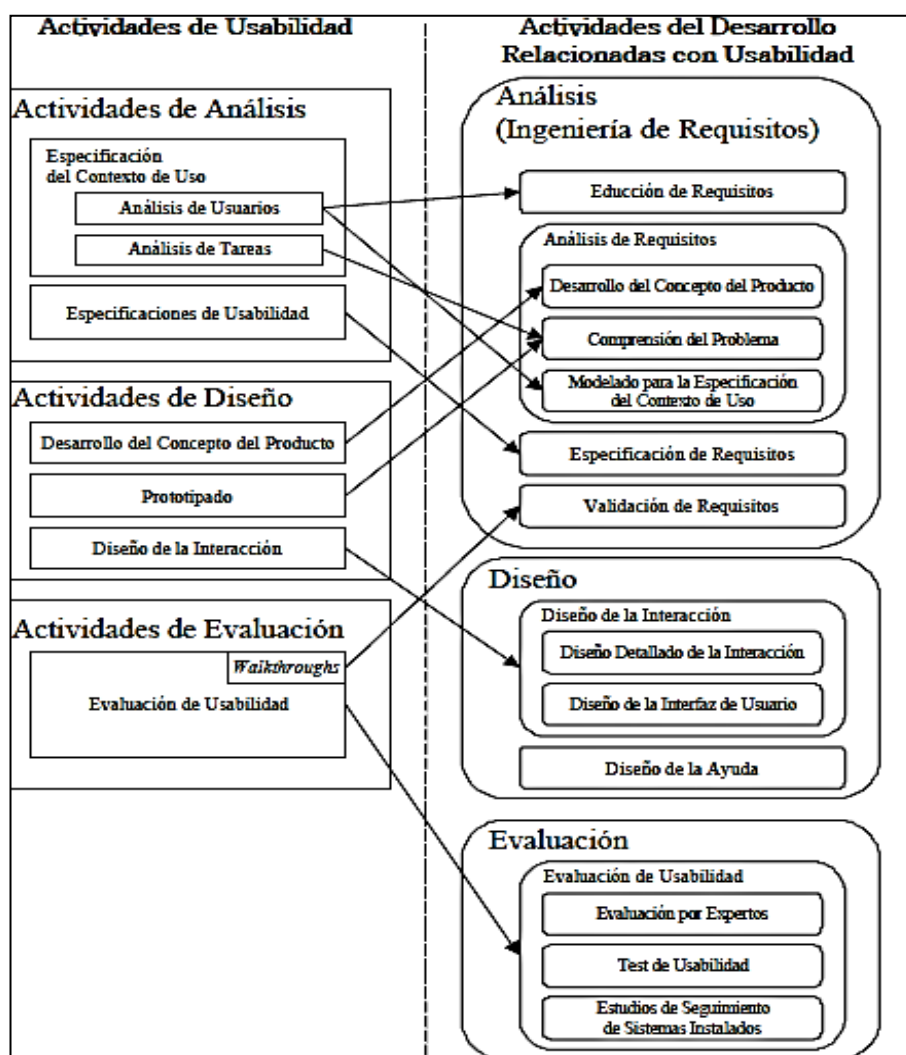


Figura 2. Relación entre actividades de usabilidad y esquema genérico de actividades del proceso de desarrollo

Fuente: Ferré, X. (s.f.). Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo (p.5).

En la figura 2 se puede apreciar una correspondencia de actividades entre los procesos de desarrollo y las actividades de usabilidad, el problema se presenta cuando ambos procesos no se ejecutan en forma paralela.

Además el análisis de tareas en el proceso de desarrollo de software se centra en los procesos y las transacciones, dejando de lado la interacción con los usuarios.

En tanto que el análisis de tareas desde la perspectiva de la usabilidad complementa el proceso que busca modular la interacción de las tareas con el usuario.

Este proyecto busca caracterizar una notación de análisis de tareas desde la perspectiva de la usabilidad.

1.3 JUSTIFICACIÓN

La interfaz de Usuario es la parte (hardware y software) del sistema informático que facilita al usuario el acceso a los recursos del ordenador. En este sentido, Harold Thimbleby sugiere que la interfaz determinará en gran medida la percepción e impresión que el usuario poseerá de la aplicación (Thimbleby, 1990).

El usuario no se interesa por conocer la parte interna de una aplicación, sino de como la aplicación le beneficia o ayuda en algo. No se puede realizar la especificación, diseñar las funciones, estructuras de datos, escribir el código y una vez casi terminado el proceso de desarrollo de la aplicación plantearse el diseño de la interfaz de usuario (Garrido, 2008). Es por eso que se debe empezar con un idea clara de los requerimientos de usabilidad des sistema que determinará cómo se quiere la interfaz y cuál sería la manera eficaz y eficiente que interactúe con el usuario.

La usabilidad está cobrando una importancia cada vez mayor en el desarrollo de software, ya que en el desarrollo de un sistema interactivo deberá tener en cuenta a los participantes que van a intervenir en el mismo: el usuario que posee la capacidad de elección y actuación, la computadora que ofrece un programa y mecanismos para su acceso (Gea Miguel, 2002).

Un buen análisis de tareas aporta valores añadidos de calidad a la hora de realizar modelos de tareas en la construcción de interfaces de usuario (Simarro, CoCTT: Modelado de tareas en un entorno colaborativo, 2007), también ayudará a comprender mejor el dominio de la aplicación dejando como resultado que el sistema sea usable.

Existen varias notaciones de modelado de tareas, de los cuales ya se han realizado varios estudios. En el presente proyecto se propone el análisis de la notación CurrentTaskTree CTT que posee una estructura jerárquica que la hace muy intuitiva y su sintaxis gráfica que la hace sencilla de interpretar.

Para abordar el proceso de análisis de tareas se propone utilizar la herramienta CTTE(Concur TaskTrees Environment) que es gratuita y está desarrollada en lenguaje java, posee diferentes iconos que representaran tanto las tareas, como las relaciones temporales permitiendo un diseño jerárquico de la estructura de los modelos de tareas.

1.4 OBJETIVOS

1.4.1 Objetivo General

Realizar un Estudio de la notación ConcurTaskTrees para el análisis de tareas en el desarrollo de la interacción humano-computador y su aplicación en el entorno CTTE (ConcurTaskTrees Environment).

1.4.2 Objetivos Específicos

- Realizar el análisis descriptivo de la notación ConcurTaskTrees.
- Analizar la Herramienta CTTE para aplicar la notación CTT.
- Crear un prototipo utilizando la herramienta CTTE.

1.5 ALCANCE

El presente proyecto de tesis de grado culminará con la entrega de un documento con el estudio de la notación ConcurTaskTrees para el análisis de tareas en el desarrollo de la interacción humano-computador.

Para lo cual se realizará una investigación detallada en donde se recopilará información importante de artículos técnicos con respecto al tema los cuales entrarán en un estudio de exclusión con el fin de obtener información valiedera y real.

Con el fin de que el proceso de desarrollo de software este centrado en el usuario se realizará un análisis de tareas cuyos resultados se apliquen en la herramienta CTTE con el objetivo principal de visualizar el modelo de interacción que resulte más factible y amigable para el usuario.

Se realizará una aplicación prototipo desarrollada utilizando la notación CTT, que será un caso práctico en donde se podrá observar claramente la importancia de usar la notación a la hora de desarrollar un sistema.

Es por eso que se utilizará la herramienta CTTE para desarrollar el modelo del prototipo de banca virtual en la cual se podrá consultar saldos y realizar transferencias.

El modelo de prototipo de la Banca virtual ayudará a visualizar todas las tareas abstractas, de usuario, del sistema y colaborativas, mostrando la relación que existe entre ellas y cómo influyen en el diseño de aplicaciones usables y exitosas.

CAPÍTULO 2

CONCURTASKTREES

2.1 DISEÑO CENTRADO EN EL USUARIO

La interfaz es el sitio donde los bits y las personas se encuentran (Negroponte, 1994), por esta razón la eficiencia debe jugar un papel muy importante para que la usabilidad por parte del usuario sea exitosa.

El diseño de centrado en el usuario se lo puede describir como la manera eficaz y eficiente de identificar dependencia de las necesidades de los sistemas a los requerimientos de las personas. Este proceso se enfoca en el usuario quien será la persona que use el producto final esto proporcionará información guía para satisfacer expectativas y facilitar el uso del sistema (Pressman, 2001).

La mayoría de los procesos que se centran en el Usuario suponen las siguientes etapas: (Pressman, 2001)

- Conocer a fondo a los usuarios finales, normalmente usando investigación cualitativa o investigación cuantitativa
- Diseñar un producto que resuelva sus necesidades y se ajuste a sus capacidades, expectativas y motivaciones
- Poner a prueba lo diseñado, normalmente usando test de usuarios (Pressman, 2001)

2.2 USABILIDAD

La 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” (ISO9241).

La facilidad de uso de un sistema se puede medir mediante la usabilidad y sus atributos en los cuales se puede establecer rangos para así lograr lo requerido. Este

término en la actualidad es muy usado por las empresas ya que es su objetivo que todo producto final contenga esta característica para asegurar éxito en el mercado.

Pero el grave obstáculo es que en tiempos pasados la ingeniería de software se centraba en requerimientos más internos que el usuario final desconoce.

“La usabilidad se presenta como un atributo del software al que se presta especial atención en determinados proyectos software, de forma similar a como se trata la seguridad” (Ferré), es por esto que en la actualidad se puede ver que en la norma ISO 9126 acerca de la calidad de software ya identifican las diferencias entre la calidad inherente del software y la calidad del uso.

La usabilidad posee atributos los cuales de acuerdo a (Nielsen., 1993) son los siguientes:

- Facilidad de aprendizaje: Es el tiempo en poder utilizar el sistema es medido hasta poder alcanzar el óptimo.
- Eficiencia: Lo que se desea una velocidad óptima en la cual el usuario final logre cumplir una tarea.
- Recuerdo en el tiempo: Este atributo ayuda que el usuario no olvide el uso de un sistema sin necesidad de estarlo usando frecuentemente.
- Tasa de errores: este atributo en cuanto más bajo sea será más usable el sistema. Ya que los errores en el proceso de realizar una tarea reducirían la eficiencia y causarían frustración en el usuario.
- Satisfacción: este atributo es el más abstracto que se debe lograr para obtener usabilidad de un sistema, ya que un usuario satisfecho es parte del objetivo de la usabilidad.

Estos atributos se pueden alcanzar pero a su vez estos no son dependientes ya que si se alcanza uno y se descuida otro, varía el nivel de usabilidad, hay que tener muy en cuenta estos atributos ya que pueden desencadenar sub atributos que también generarían valores agregados.

2.3 RELACIÓN DE LA USABILIDAD Y EL DISEÑO DE LA INTERFAZ DE USUARIO

La interfaz gráfica mostrará la interacción entre el usuario y la máquina, es ahí donde entra la usabilidad como la manera eficiente de que esas operaciones se den exitosamente. La interfaz de usuario debe estar ligada completamente al diseño de la interacción ya que si trabajan independientemente no se alcanzaría el nivel de usabilidad óptima. Es por esto que el papel de la usabilidad esta debe estar presente desde el inicio hasta el final del proceso de desarrollo de software, ya que estará presente no solo en el diseño de la interacción si no que debe siempre respaldarse y estar vinculado con la lógica del negocio.

Los principales problemas para alcanzar un nivel aceptable de Usabilidad dentro de un proceso de desarrollo de Software son los siguientes: (Ferré X. , Principios Básicos de Usabilidad para Ingenieros Software)

- Los ingenieros software entre sus objetivos dentro de su formación son realizar algoritmos bien elaborados, mantenimiento óptimo y esto puede repercutir negativamente en la usabilidad.
- Los conceptos que se manejan en la Ingeniería de Usabilidad no se relacionan en lo absoluto a los manejados en la Ingeniería del Software, y esto genera discusiones en los equipos de trabajo.
- Es necesario poder incluir a la ingeniería de la Usabilidad dentro del proceso del Software como un pilar fundamental para todo el ciclo de vida y sus diferentes etapas.

Es fácil inferir que un buen diseño deberá ser comprensible, fácil de usar, amigable, claro, intuitivo y de fácil aprendizaje para el usuario (Ferre, 2005). Para poder asegurar que un diseño cumple con estos requisitos no basta simplemente con una actitud empática del diseñador durante el desarrollo de la aplicación; es imprescindible la adopción por parte de éste de técnicas, procedimientos y métodos que aseguren empíricamente la adecuación del diseño a las necesidades, habilidades y objetivos del usuario (Iazza).

2.4 RELACIÓN ENTRE EL PROCESO DE DESARROLLO DE SOFTWARE Y EL PROCESO DE DESARROLLO DE INTERACCIÓN.

El proceso de desarrollo de Sistemas de Información está compuesto por cuatro fases bien definidas, que de acuerdo a (Pressman, 2001) que son las siguientes:

- Análisis
- Diseño
- Codificación
- Prueba
- Mantenimiento

Algunas de las metodologías tradicionales más utilizadas para el desarrollo de software son las denominadas “proceso personal de software (Personal Software Process PSP)” y el “proceso en equipo para el software Team Software Process TSP”, también existen las metodologías ágiles como Scrum, XP, DAS y metodologías tradicionales como RUP (Sommerville, 2005).

El proceso de desarrollo de interacción entre el usuario y la computadora (Human Computer Interaction HCI), está compuesto por un conjunto de tareas o actividades.

Pressman (Pressman, 2001) define tres actividades, la primera actividad está enfocada a conocer a los usuarios finales aplicando técnicas de investigación cualitativa o cuantitativa.

La segunda actividad es el diseño de un producto que resuelva sus necesidades y se ajuste a sus capacidades, expectativas y motivaciones. Finalmente es necesario poner a prueba lo diseñado, normalmente usando test de usuarios.

En tanto que (Ferre, 2005) propone 3 actividades principales. Las primeras actividades están enfocadas al Análisis que comprende la especificación de contexto de uso, en donde se resalta el análisis de los usuarios y el análisis de tareas conjuntamente con la especificación de requerimientos de usabilidad. El segundo conjunto de actividades están orientadas al diseño y comprenden el desarrollo del

concepto, el prototipo y el diseño de la interacción. Finalmente propone las actividades de Evaluación de la usabilidad.

La interacción es una parte importante de un sistema, y el proceso de desarrollo de interacción debería ser complementario al proceso de desarrollo de software, sin embargo es aquí donde surge la brecha entre estos dos procesos ya que, las técnicas HCI tienen como objetivo incrementar el nivel de usabilidad del producto software, sin embargo hay un gran desconocimiento entre los desarrolladores acerca de las técnicas HCI (Seffah A, 2003) & (Ferré, Juristo, H, & Constantine, 2001) , ambos autores coinciden que el proceso de desarrollo de HCI debe iniciarse en etapas tempranas del desarrollo del software y que los dos procesos pueden realizarse a la par.

Los sistemas interactivos están siendo diseñados y probados por el software y los ingenieros informáticos no están familiarizados con UCD y las técnicas de ingeniería de usabilidad relacionadas.

Aunque la mayoría de los desarrolladores de software pueden haber estado expuestos a los conceptos básicos, como pautas de diseño GUI, pocos desarrolladores son capaces de entender la caja de herramientas de diseño humano / centrado en el usuario (UCD) a un nivel que les permita incorporar en el ciclo de vida de desarrollo de software (Seffah A, 2003).

La aplicación de técnicas de usabilidad en el desarrollo de software no es fácil (Ferré, Juristo, H, & Constantine, 2001); una de las virtudes del campo de la HCI reside en su carácter multidisciplinar, pero esta característica es al mismo tiempo uno de los inconvenientes principales para su integración con la Ingeniería del Software.

Mientras la base teórica y práctica de la HCI se nutre de la sociología, psicología, diseño industrial, diseño gráfico, etc; los ingenieros software tienen un enfoque netamente ingenieril. Ambas disciplinas hablan un lenguaje distinto y enfocan el desarrollo de software desde una perspectiva distinta (Ferré X. , Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo). Según describen Constantine y Lockwood (Constantine & Lockwood, 1999), la Ingeniería del

Software desarrolla sistemas software centrándose en la parte interna del sistema, en la lógica del proceso y en la organización de los datos.

Por tanto la calidad software se define con criterios tales como eficiencia, fiabilidad, reusabilidad o modularidad. Estos aspectos del sistema son ajenos casi por completo al usuario final del producto. Sin embargo, la interacción con el usuario se ha tratado habitualmente como un tema de importancia secundaria. (Ferré X. , Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo)

Las distintas teorías y modelos asociados al diseño de interfaces gráficas han llevado a confusiones acerca de qué sucede durante la interacción y cómo llevar a cabo este diseño orientado al usuario. Los actuales enfoques no dan cuenta de la realidad epistemológica que la conforma, pues se limitan a establecer una visión genérica de la interfaz desde su perspectiva estética, cuando en realidad corresponde más a la dimensión de la Comunicación Visual, como soporte de la articulación verbo-icónica de los distintos elementos gráficos que cumplen un objetivo completamente funcional.(Luis, 2010).

Ferre ante esta brecha propone una adecuación del esquema de Actividades de Usabilidad a los Conceptos y Terminología de la Ingeniería del Software, dando como solución la propuesta que se necesita adaptar el esquema de actividades (y con él el de técnicas) a las actividades del proceso software general. Para ello relaciona primero cada actividad de usabilidad con las actividades correspondientes en un proceso de desarrollo software genérico, y a continuación asigna cada técnica de usabilidad a la actividad correspondiente del nuevo esquema genérico.

Para la definición de un esquema de actividades de un proceso software genérico se basó en el SWEBOK (SoftWare Engineering Body Of Knowledge) (Committee, 2001) desarrollado por la IEEE Computing Society. La propuesta se visualiza en la Figura 3.

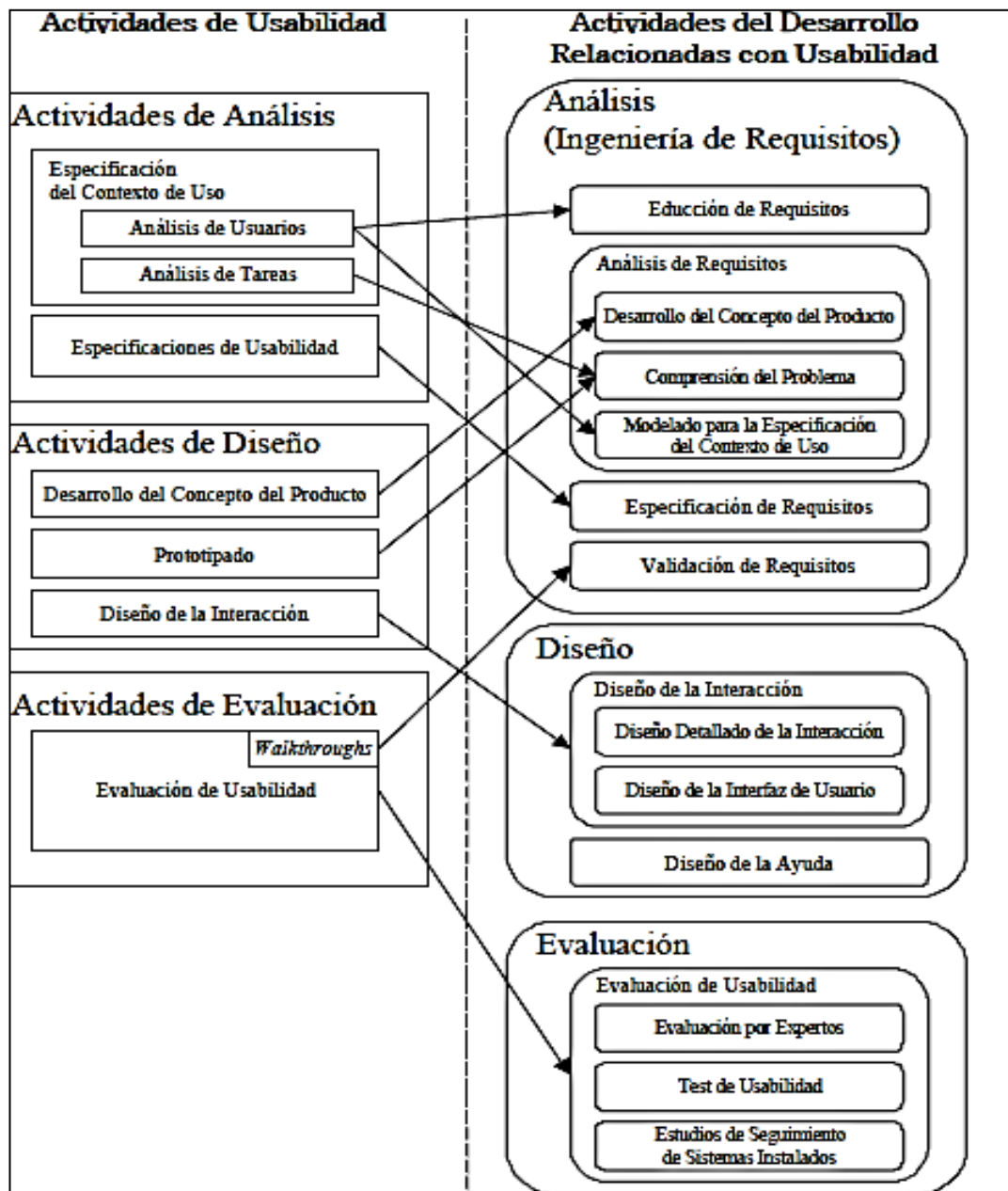


Figura 3. Relación entre actividades de usabilidad y esquema genérico de actividades del proceso de desarrollo

Fuente: Ferré, X. (s.f.). Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo (p.5).

Como se observa en la figura X, las actividades de Usabilidad, guardaran una relación con casi todo el proceso de desarrollo de software. En este punto se puede resaltar los dos procesos contemplan actividades relacionadas al análisis de las tareas que realiza el usuario. En este trabajo de investigación se centra en el análisis de tareas desde la perspectiva de la usabilidad.

2.5 ANÁLISIS DE TAREAS

“El modelado de tareas para el diseño y desarrollo de interfaces de usuario consigue un nivel de abstracción superior, haciendo que el desarrollo de software se convierta en un proceso de ingeniería lo que conlleva a conseguir un software de calidad y permite obtener aplicaciones interactivas centradas en el usuario” (B.A., 2007).

Un buen análisis de tareas implica conocer el dominio del sistema a desarrollar, para alinearse de una forma óptima a los objetivos que desea alcanzar el sistema, además conocer los futuros potenciales usuarios a los que estará dirigido la aplicación. (Ferre, 2005)

Una de las Actividades para conseguir un diseño centrado en el usuario es el análisis de tareas que consiste en un levantamiento de lo que un usuario está obligado a hacer en términos de acciones y / o procesos cognitivos para lograr una tarea (Iazza). En la actualidad existe una gama bien amplia de notaciones que son de ayuda a la hora de modelar, que van a depender de los requerimientos. En la siguiente sección se muestra las notaciones más destacadas

GTA: Esta centrado en el grupo, usuario y proceso, emplea ontologías para la validación de modelos además no posee una herramienta de modelado (Molina, Redondo, & Ortega).

CTT: Notación altamente difundida entre la comunidad IPO, está centrado en el grupo y en el usuario además posee un editor llamado CTTE (Molina, Redondo, & Ortega).

APM: Centrado en el proceso, Modela recursos junto con la interacción entre los usuarios, realiza un modelado de la interacción basada en la teoría de actos del habla

y hace uso de templates o bloques de construcción y clichés o patrones, pero no posee herramienta de modelado (Molina, Redondo, & Ortega).


MAD: Es muy similar a la notación CTT, facilidad para el modelado de tareas cooperativas y dispone de un entorno grafico llamado K-MADe (Molina, Redondo, & Ortega).

MARIA: El marco MARIA apoya descripción de interfaces de usuario en los niveles abstractos y concretos. El lenguaje abstracto es independiente de la plataforma de interacción. Posee diversas plataformas (gráfico de escritorio, teléfono inteligente gráfica basada en el contacto, móvil gráfica, vocal, multimodal (combinación de gráfica y vocal), la herramienta de modelado se llama MARIAE (Molina, Redondo, & Ortega).


2.6 NOTACIÓN CTT


Es una notación utilizada para realizar análisis de tareas de forma estándar, desarrollada por Fabio Paternò, empleada para modelar las tareas que un usuario puede llevar a cabo en una aplicación interactiva (Paternò, 2007)

Algunas de las principales ventajas de esta notación radica en su estructura jerárquica que la hace muy intuitiva. La sintaxis gráfica empleada que la hace sencilla de interpretar, y finalmente un conjunto rico de operadores que permite expresar de forma clara las relaciones temporales existentes entre las tareas y los usuarios encargados de llevarlas a cabo (Paternò, 2007). “En CTT, se identifican 4 tipos de tareas, en función del actor que la llevará a cabo:

Tareas de usuario (): Tareas realizadas completamente por el usuario, son tareas cognitivas o físicas que no interactúan con el sistema.

Tareas de aplicación (): Tareas realizadas por la propia aplicación.

Tareas de interacción (): Son tareas que realiza el usuario interactuando con la aplicación por medio de alguna técnica de interacción.

Tareas abstractas (): Tareas que requieren acciones complejas y que por ello no es fácil decidir donde se van a realizar exactamente. Son tareas que van a ser descompuestas en un conjunto de nuevas subtareas.” (Paternò, 2007).

2.7 CARACTERÍSTICAS CTT

CCT es un método de análisis de tareas que permite mostrar, gráficamente, las relaciones temporales (de secuencialidad, concurrencia, recursión, etc.) existentes entre las actividades y los usuarios que son necesarios para llevar a cabo un conjunto de tareas. A continuación se presentan diferentes características particulares que posee la notación CTT.

Características Principales:

- Estructura jerárquica: ofrece una amplia gama de granularidad, permitiendo que las estructuras de trabajo grandes y pequeños puedan ser reutilizados, y permite a las estructuras reutilizables de las tareas ser definida en bajos y altos niveles semánticos (Avilez, 2012).
- Notación concurrente - Operadores LOTOS: Puede ser definido un rico conjunto de posibles relaciones temporales entre las tareas. Este tipo de aspecto esta generalmente implícito, expresado de manera informal en la producción de análisis de tareas, además los operadores LOTOS es una notación concurrente formal que era una buena opción para especificar las interfaces de usuario, ya que permite a los diseñadores describir los comportamientos basados en eventos y modificaciones del estado (Avilez, 2012).

- Actividades Enfocadas: se permite a los diseñadores centrarse en las actividades que los usuarios realizan, que son los aspectos más relevantes en el diseño interactivo aplicaciones que abarcan tanto al usuario como aspectos relacionados con el sistema (Avilez, 2012).
- Sintaxis Grafica: una sintaxis gráfica a menudo, no siempre, es más fácil interpretar, en este caso se refleja la estructura lógica por lo que tiene una forma de árbol (Avilez, 2012).

El objetivo de la notación CTT es superar las limitaciones de las notaciones utilizadas anteriormente para diseñar aplicaciones interactivas como es el caso de las notaciones APM, PROCLETS que se centran en el proceso y no en el usuario ni el grupo (Molina, Redondo, & Ortega). Además como ventaja se tiene que es una notación fácil de usar que puede apoyar en el diseño de aplicaciones. (Paternò, Report and Refine Usability Evaluation Methods, 2007), La Figura 4 muestra la ubicación de las principales ventajas de la notación CTT:

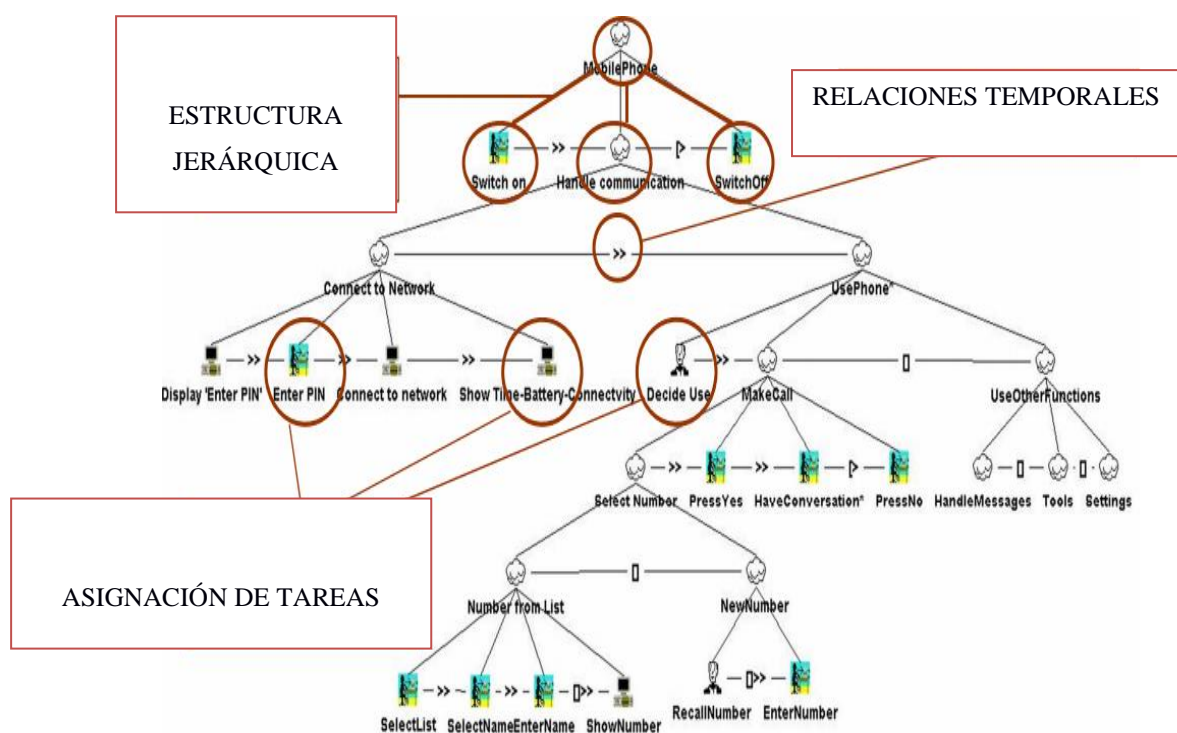


Figura 4: Características principales de CTTE

Fuente: (Avilez, 2012).

2.8 OPERADORES

Para la descripción de la secuencialidad del modelo se utilizan una serie de operadores temporales, extensión de los existentes en LOTOS3 (Bolognesi & Brinksma, 1987), que facilitan la descripción de las relaciones temporales existentes entre tareas (Paternò, 2007).

La tabla 1 presenta la variedad de los operadores temporales de La notación CTT (Tabla 1):

Tabla ¡Error! Secuencia no especificada. **Operadores Temporales**

NOMBRE	SINTAXI
Elección	T1 [] T2
Independencia de Tareas	T1 = T2
Tareas Concurrentes	T1 T2
Sincronización	T1 [] T2
Habilitación	T1 >> T2
Habilitación con transferencia de información	T1 []>> T2
Desactivación	T1 [> T2
Interacción	T1*
Tarea Opcional	[T1]
Suspender-Reanudar	T1 > T2

Fuente: (Paternò, 2007)

Orden de prioridad de la relación temporal, ordenadas de mayor a menor:

Operador unitario [] |=| ||| [|] | [> |> >> [] >>

2.8.1 Interacción

Las tareas en un mismo nivel representan diferentes opciones o diferentes tareas al mismo nivel de abstracción que tienen que haber realizado. En la Figura 5, se puede apreciar que para obtener la tarea 1, se debe antes realizar la tarea 2 y tarea 3.

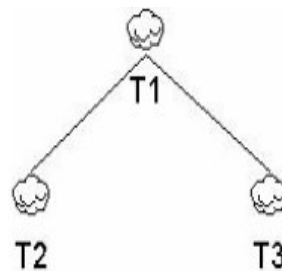


Figura 5: Interacción de Tareas

Fuente: (Consiglio Nazionale, 2010)

2.8.2 Habilitación

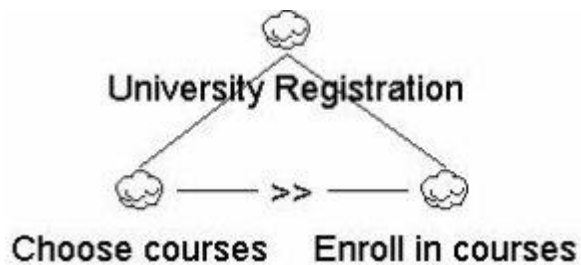


Figura 6: Habilitación

Fuente: (Consiglio Nazionale, 2010)

En la Figura 6 especifica que la segunda tarea no puede realizarse hasta que haya realizado la primera tarea.

2.8.3 Habilitación con transferencia de información

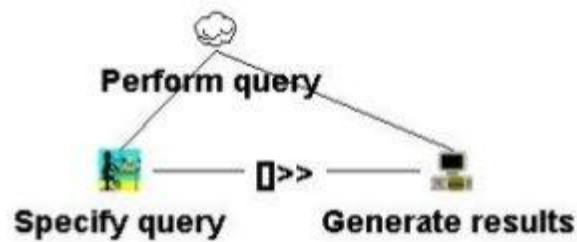


Figura 7: Habilitación Info

Fuente: (Consiglio Nazionale, 2010)

En la Figura 7 especifica que la segunda tarea no puede realizarse hasta que la primera tarea se lleva a cabo, y que la información producida en la primera tarea es utilizada como insumo para la segunda tarea.

2.8.4 Tareas concurrentes

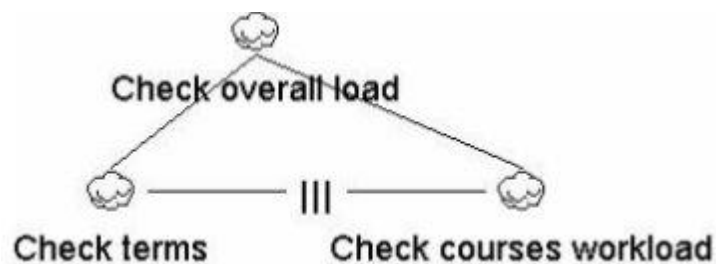


Figura 8: Tareas concurrentes

Fuente: (Consiglio Nazionale, 2010)

En la Figura 8 las tareas se pueden realizar en cualquier orden, o al mismo tiempo, incluyendo la posibilidad de iniciar una tarea antes de que la otra tarea haya sido completada.

2.8.5 Sincronización

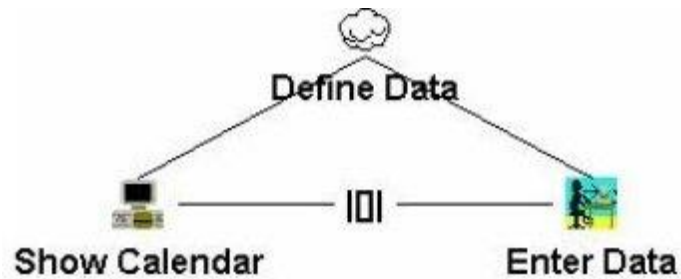


Figura 9: Comunicación Tareas Conc

Fuente: (Consiglio Nazionale, 2010)

En la Figura 9 se muestra las tareas que pueden intercambiar información mientras se realiza simultáneamente.

2.8.6 Desactivación

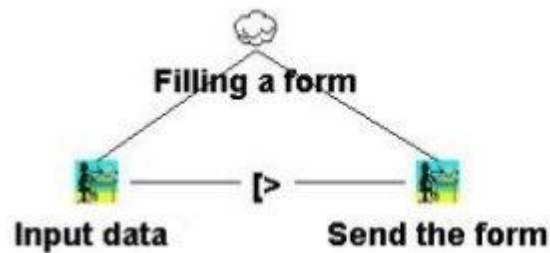


Figura 10: Desactivación

Fuente: (Consiglio Nazionale, 2010)

En la Figura 10 se muestra que la primera tarea (por lo general una tarea iterativa) es completamente interrumpida por la segunda tarea.

2.8.7 Independencia de tareas

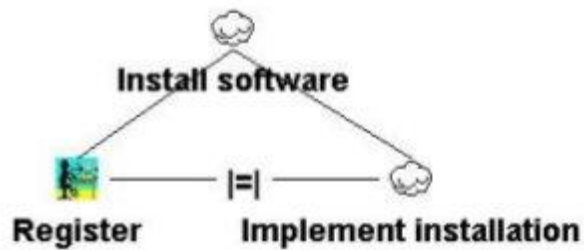


Figura 11: Independencia de tareas

Fuente: (Consiglio Nazionale, 2010)

En la Figura 11 se muestra que las tareas se pueden realizar en cualquier orden, pero cuando uno empieza entonces tiene que terminar antes de que el otro pueda comenzar.

2.8.8 Suspende-reanudar

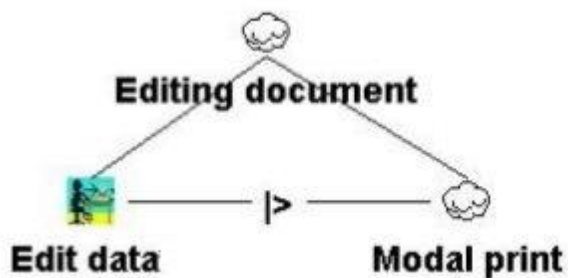


Figura 12: Suspende-Reanudar

Fuente: (Consiglio Nazionale, 2010)

En la Figura 12 se muestra que la primera tarea puede ser interrumpida por la segunda. Cuando la segunda finaliza entonces el primero puede ser reactivado desde el estado alcanzado antes.

2.9 HERRAMIENTA CTTE

CTTE (The ConcurTaskTrees Environment) es una herramienta gratuita desarrollada en lenguaje java que cuenta con diferentes iconos que representaran tanto las tareas, como las relaciones temporales que se pueden dar entre ellas, permitiendo un diseño jerárquico de la estructura de los modelos de tareas. (Consiglio Nazionale, 2010).

La notación CTT se puede usar en la Herramienta CTT Enviroment CTTE, que es un editor en el que se puede construir el modelo de tareas y generar un modelo basado en modelo interactivo arquitectónico (Paternò, Report and Refine Usability Evaluation Methods, 2007).

Es posible construir tareas cooperativas Modelos o solo Modelo de tareas de Usuarios.

Otra de las características más importantes de esta herramienta es que permite guardar los modelos de tareas creados en varios formatos: como imagen en formato jpg, en formato LOTOS, HTML Y XML. Este último es el formato que hace más interesante el uso de esta herramienta (<http://giove.cnuce.cnr.it/ctte.html>).

La herramienta CTTE permite a los diseñadores elaborar modelos de tareas utilizando la notación CTT, en la Figura 13 se observa la interfaz gráfica del editor CTTE.

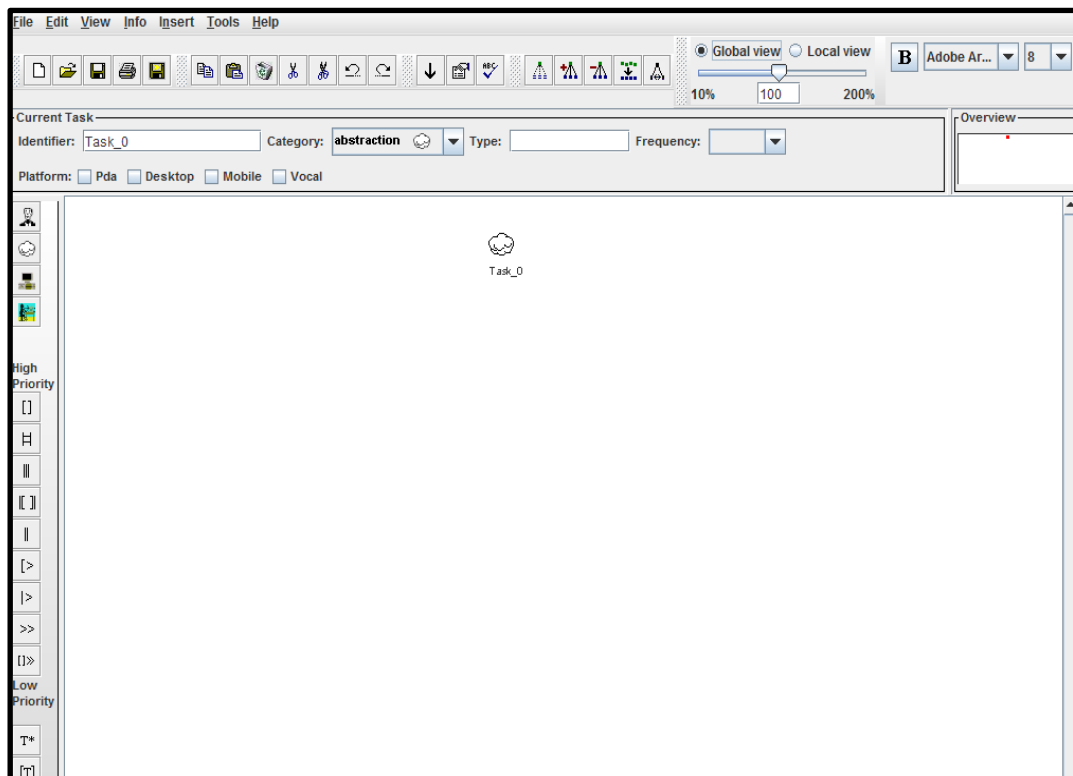


Figura ¡Error! Secuencia no especificada.**3: Editor CTTE**

Fuente: (Consiglio Nazionale, 2010)

Esta herramienta surgió para apoyar el diseño de aplicaciones interactivas. El código ejecutable está disponible al público y es gratuito, ha recibido miles de descargas en diferentes partes del mundo (Consiglio Nazionale, 2010).

Los modelos de tarea creados por la herramienta CTTE son compatibles con la notación ConcurTaskTrees. sin embargo, pueden ser guardados mediante diferentes extensiones de archivo (Consiglio Nazionale, 2010).

Se identifica las siguientes extensiones de archivo para el modelo de tarea CTT que se puede generar con CTTE:

- .ctt / .ctte /.cctt /.xml. El formato XML, ayuda a mejorar el intercambio con otras notaciones y formatos de uso común

También se menciona la posibilidad de guardar los modelos de tareas CTT como archivos JPG, que puede ser muy útil para leer las especificaciones del modelo de tareas.

En resumen, CTTE cuenta con una interfaz gráfica que permite la manipulación de un conjunto de elementos (acciones de usuario), agrupados jerárquicamente y conectados por comandos temporales. Estos elementos y los comandos son los proporcionados por la notación CTT.

CAPÍTULO 3

CONCURTASKTREES ENVIRONMENT

3.1 SINTAXIS CTTE

El editor CTTE posee una amplia gama de opciones que permiten crear y probar un prototipo. CTTE cuenta con varias opciones, entre las que se tiene:

3.1.1 Relación Temporal Combinada

Existe la posibilidad de combinar los operadores unarios con los operadores que ofrece CTT, pero en este caso algunas expresiones pueden resultar erróneas.

La herramienta CTTE ofrece la posibilidad de comprobar la inconsistencia.

A continuación se presenta la posible expresión equivocada que se puede generar.

- La expresión [A] *>> B está mal, B nunca es alcanzable

Nota: a la izquierda y derecha del operador |>, [> y [] no puede ser opcional.

[A]: Significa que la tarea A es opcional.

Iteración A * (donde "A*" significa que la tarea A es iterativo)

3.2 OPCIONES CTTE

La herramienta CTTE facilita y automatiza varias opciones como las siguientes:

- Con la finalidad de facilitar a los usuarios la correcta construcción de especificaciones del modelo de tarea, CTTE ofrece la posibilidad de realizar algunas comprobaciones automáticas, orientadas a detectar posibles errores que podrían hacer a la especificación incompleta o ambigua. Para habilitar dicha funcionalidad se debe seleccionar la opción herramientas luego la opción mirar la estructura del modelo (Tools|Check Model Structure), y como resultado se observa una ventana con la lista de errores como lo muestra la Figura 14

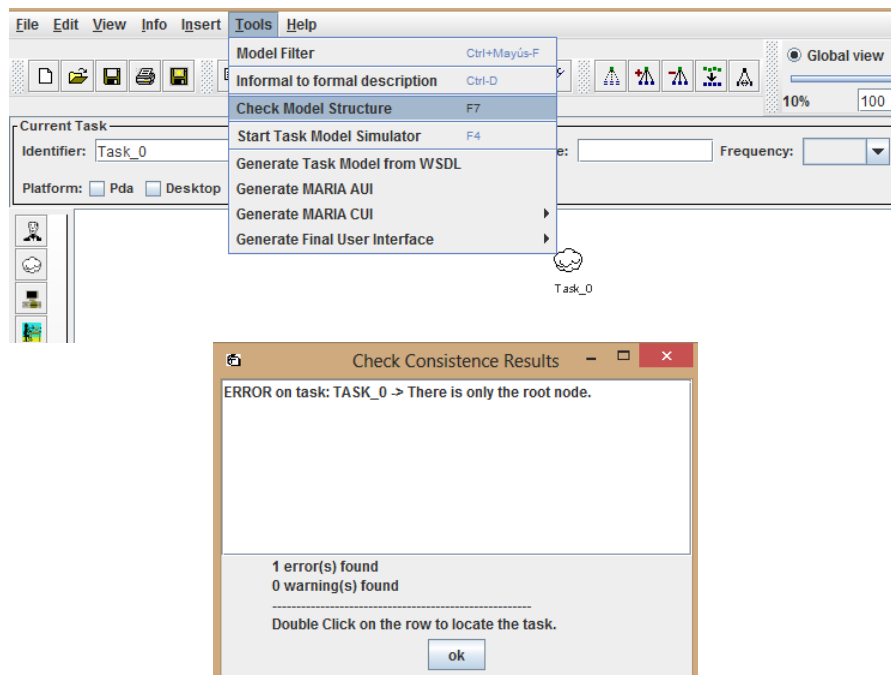
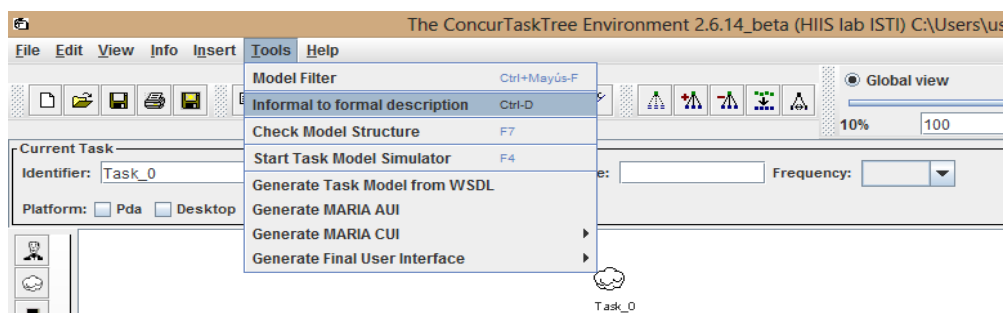


Figura 14. Comprobación Automática

- Los usuarios pueden tener una descripción informal de un sistema, o parte de la muestra de escenarios que describen el comportamiento típico del sistema. CTTE ofrece una opción útil para apoyar a los usuarios para estructurar un modelo formal mientras extraen la información pertinente de los escenarios descritos (Tool | Informal to Formal Description), se muestra en la figura 15.



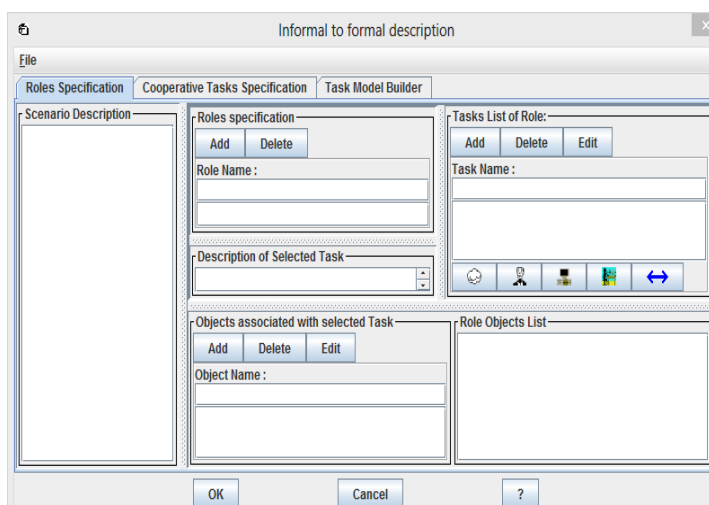


Figura 15. Descripción Formal

- Especialmente con grandes especificaciones, es muy útil tener la posibilidad de generar y seguir algunas secuencias o caminos que son posibles dentro de un modelo de tarea. Una vez que haya especificado el comportamiento de un sistema específico, CTTE permita "Simular" un comportamiento por medio de la opción Simular Modelo. (Tools| Start Task Model Simulator).

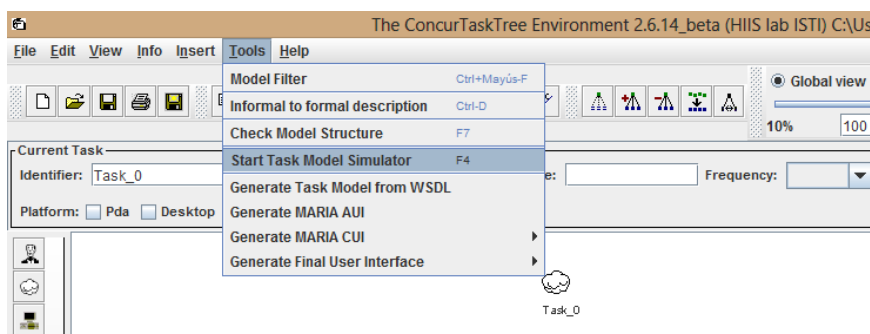


Figura 16. Simulador de Tareas

El simulador ha demostrado ser útil en varios casos:

- Los diseñadores pueden comprobar si el comportamiento especificado es realmente lo que pretenden describir. Esto es importante porque, especialmente en el caso de grandes especificaciones, es difícil de entender

de manera inmediata el comportamiento general que se deriva de la combinación de la estructura jerárquica y los operadores temporales.

- Puede soportar una discusión multidisciplinar donde personas con diferentes orígenes (diseñadores, desarrollador, usuarios finales) discuten sobre la decisión del diseño en el nivel de tarea.
- Puede ser empleado como documentación interactiva de una aplicación para explicar a los usuarios finales cómo usarlo (indicando en que las tareas de orden pueden ser realizados, opciones posibles, y otra información dinámica).
- CTTE facilita la opción de observar las estadísticas del modelo de tarea según los roles propuestos, es algo muy útil a la hora de mostrar resultados y tener una idea general de todo el modelo, como observamos en la Figura 17(Info| Task Model Statistics).

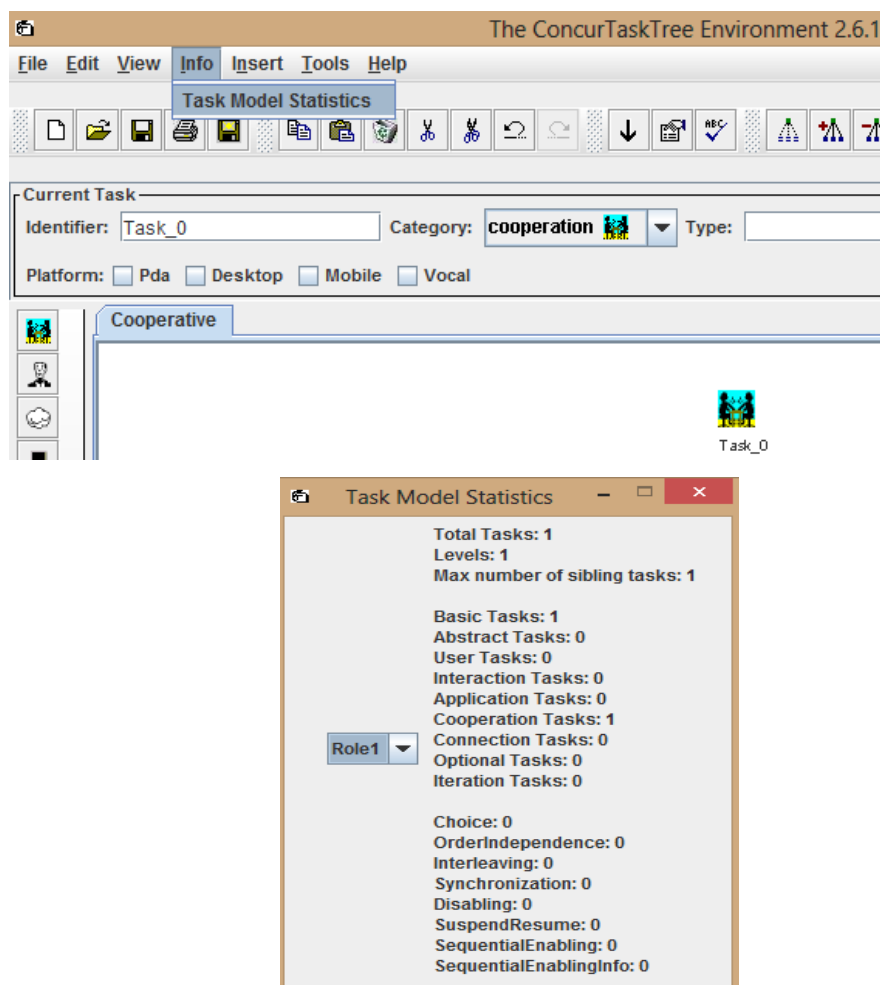


Figura 17. Estadísticas del Modelo

3.2.1 Habilitar la opción "Modo Cooperativo"

Al iniciar CTTE, por defecto permite construir un modelo de tarea de usuario único.

Si se desea crear un modelo de tarea cooperativa se tiene que habilitar la opción de Ver, opción de modo cooperativo. Como resultado, la raíz de la tarea Cooperativa modelo ("root Cooperativa") se mostrará en el panel del editar y la etiqueta "Cooperativa" se muestra en el panel de pestañas dentro de la ventana principal (Edit | Cooperative Mode) como se muestra en la Figura 18.

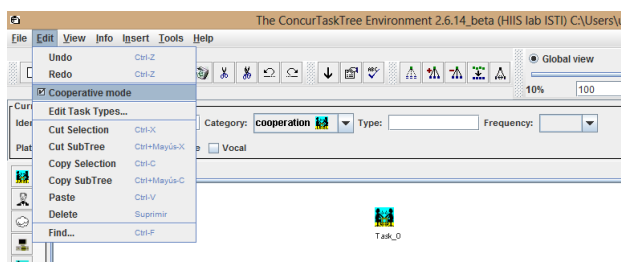


Figura 18. Modo Cooperativo

3.2.2 Añadir nuevos roles

Antes de construir el modelo de tarea cooperativa, debe especificar el modelo de "roles" involucrados, por lo que hay que añadir nuevos roles para el modelo de tarea cooperativa, como se muestra en la Figura 19.

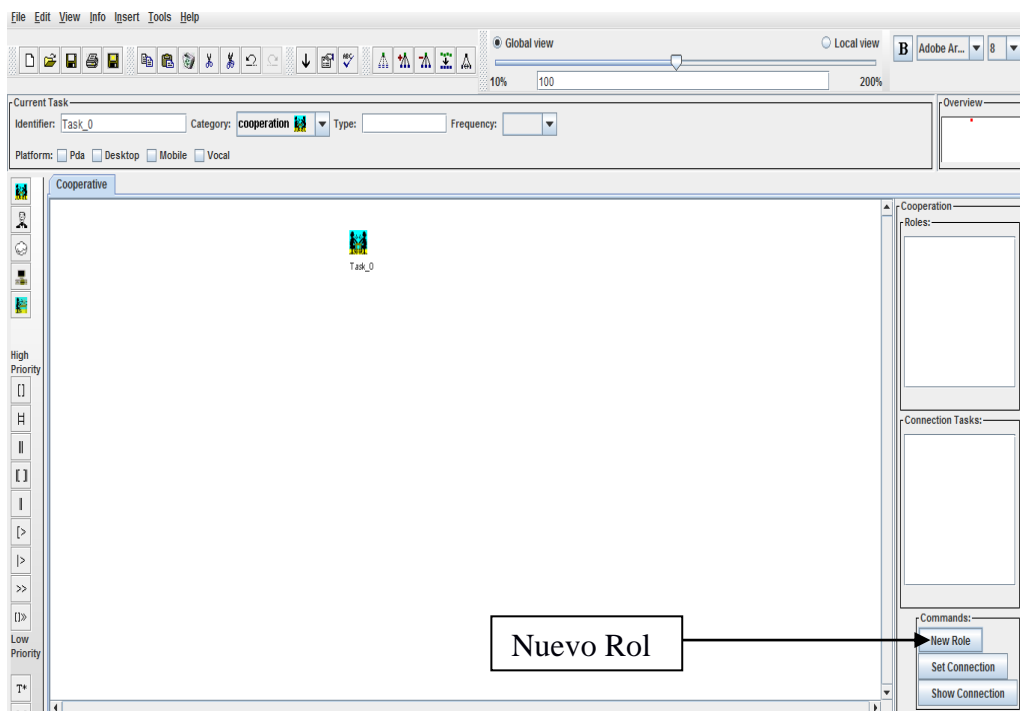


Figura 19. Añadir nuevo Rol

3.2.3 Centrarse en las "tareas de conexión"

Si se está construyendo un modelo de tarea cooperativa, se podrá observar las tareas para cada uno de los roles, se selecciona el modelo de tarea cooperativa en la parte derecha de la ventana principal, se visualiza en la Figura 20.

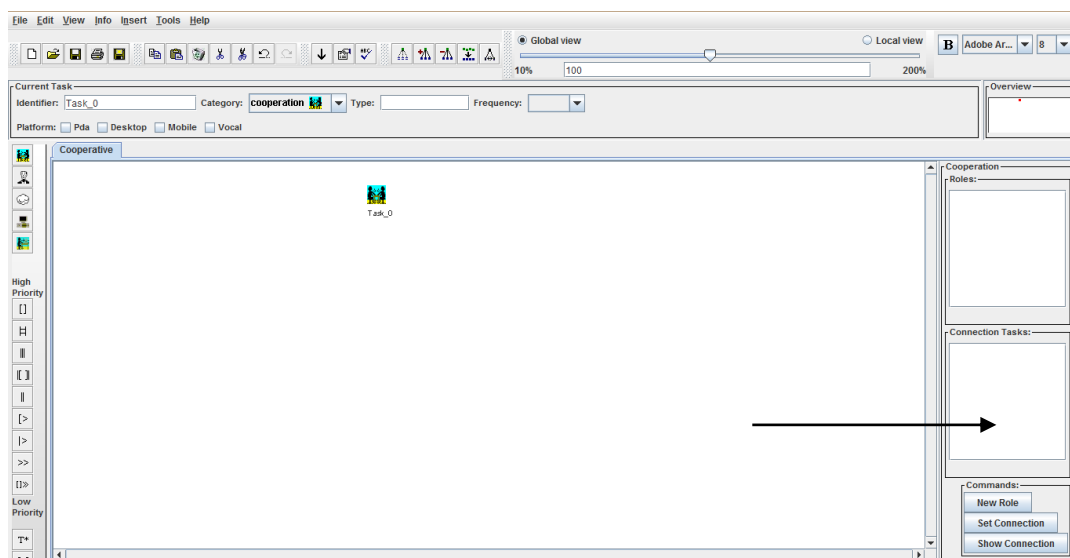


Figura 20. Tareas de Conexión

3.2.4 Compatibilidad CTTE

Hay tres formatos de archivo principales en las que se pueden guardar un modelo de tarea (o una parte de él):

- ".cctt" formatos / ". Ctt"

La ampliación de una especificación CTT es ".ctt" para los modelos de trabajo de un solo usuario y ".cctt" para los modelos de trabajo cooperativos. Por defecto se guardará el modelo de tarea entera, pero puede también guardar solo una parte de ella, al seleccionar la raíz del subárbol se procede a la opción (File | Save SubTree As).

- El formato ".jpg"

El CTTE da a los usuarios la posibilidad de guardar sus modelos de tarea en formato ".jpg" (File | Save As JPG , para guardar un árbol entero, o | Save subTree As).

- Formato ".xml"

Una especificación CTT se puede guardar también en formato XML para mejorar el intercambio con otras notaciones y formatos de uso común (File | Save As XML), como se muestra en la Figura 21.

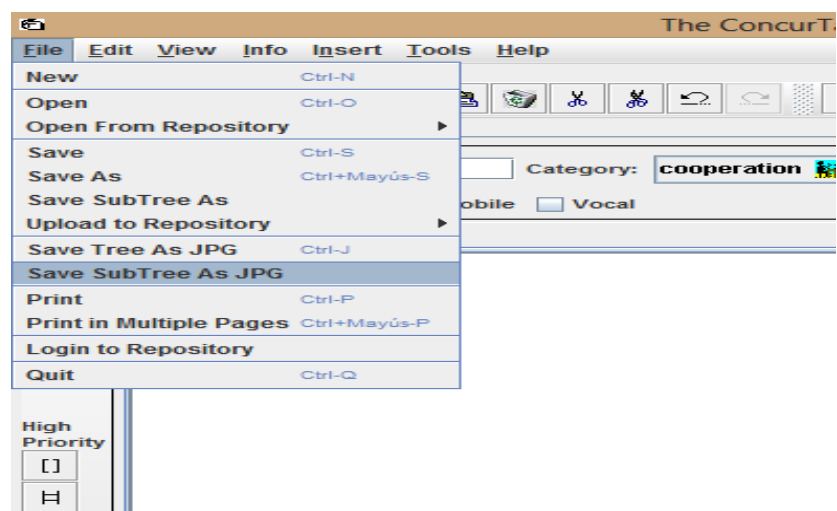
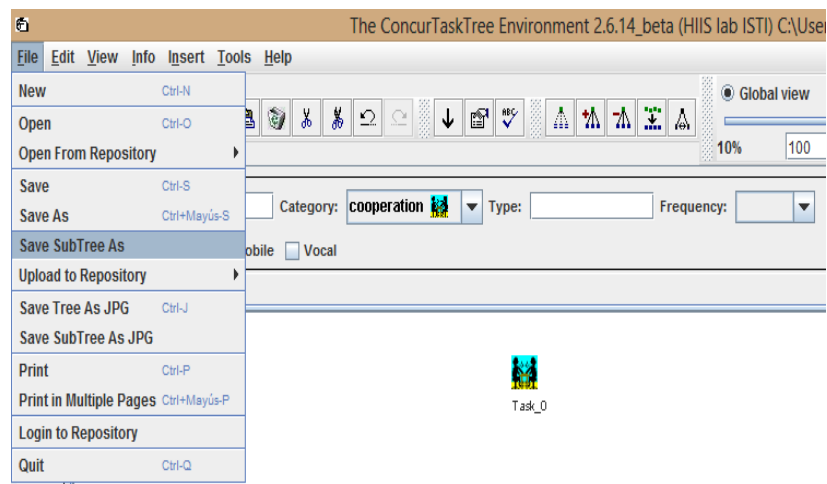
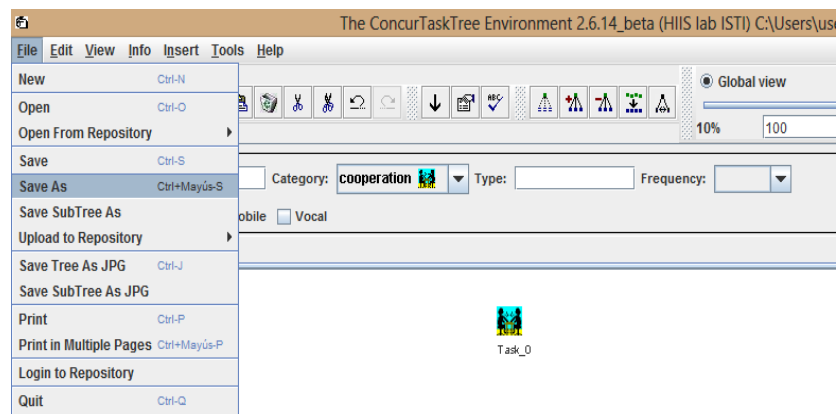


Figura 21. Formatos Disponibles para Guardar

3.2.5 Compatibilidad con MARIA

CTTE proporciona la opción de guardar el modelo de tareas y poder exportarlo y trabajar en MARIAE (Consiglio Nazionale, 2010). Las diferentes opciones que tiene MARIA son:

- MARIA CUI DESKTOP
- MARIA CUI MOBILE
- MARIACUI VOCAL
- MARIA MULTIMODAL DESKTOP
- MARIA MULTIMODAL MOBILE

3.2.6 Descripción de Tarea Individual

Por cada tarea en CTTE, existe un número de atributos modificables, hay 3 secciones que son:

General: Información general que incluye el identificador y nombre extendido de la tarea, su categoría y tipo, frecuencia de uso, alguna anotación informal que el diseñador quiera almacenar, indicación de posibles condiciones previas, y, si es un proceso iterativo, o una tarea de conexión opcional.

Mientras que la categoría de una tarea indica la asignación de su rendimiento. El tipo de tarea permite a los diseñadores definir tareas de grupo, dependiendo de su semántica. Cada categoría tiene sus propios tipos de tareas.

Para la categoría de tarea de interacción los tipos de tarea son de selección (la tarea permite al usuario seleccionar una pieza de información), control (la tarea permite al usuario activar un evento de control que puede activar una funcionalidad), y edición (la tarea permite al usuario introducir un valor). Esta clasificación es útil para conducir la elección de la interacción o técnicas de presentación más adecuada para apoyar la ejecución de la tarea.

La frecuencia de uso es otra información útil porque las técnicas de interacción asociados con las tareas más frecuentes tienen que estar bien definidos para obtener una interfaz de usuario eficiente, como se visualiza en la Figura 22.

Task Properties

General Objects Time Performance

Task Properties

Identifier: Task_0

Name: name

Category: abstraction

Type: None

Frequency:

Platform: Pda Desktop Mobile Vocal others

Context:

Description:

Iterative Optional Part of Cooperative Task

Precondition: none

Postcondition: none

Update Cancel Clear Ok

Figura 22. Propiedades de una Tarea

Objetos: Es posible indicar los objetos que se dispone y que pueden ser manipulados para realizar una tarea. Los objetos pueden ser objetos de interfaz de usuario u objetos de aplicación dominio.

También es posible indicar los permisos de acceso de usuario para manipular los objetos. En aplicaciones multiusuario, diferentes usuarios pueden tener acceso diferente a los permisos, como se visualiza en la Figura 23.

Task Properties

General Objects Time Performance

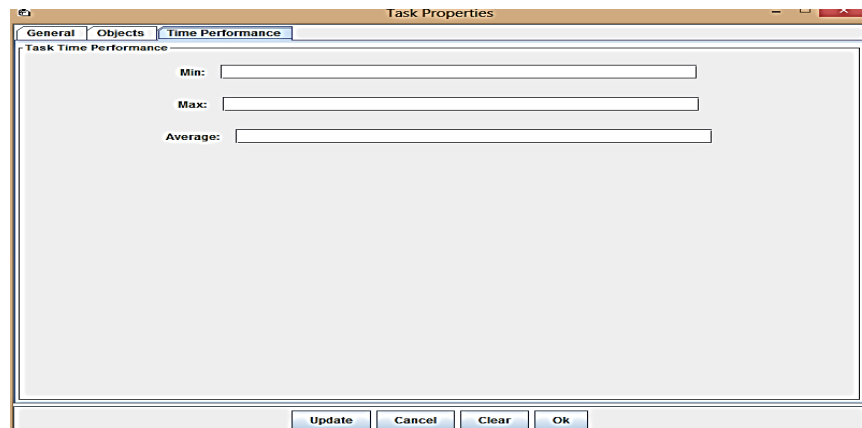
Objects Required to Perform the Task

Name	Class	Type	AccessMode	Cardinality	Platforms:	Pda	Des	Mob	Voc
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
others						<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Update Cancel Clear Ok

Figura 23. Propiedades de una Tarea

Tiempo de Actuación: También es posible indicar tiempo estimado de rendimiento (incluyendo una distinción entre un mínimo de tiempo, máximo y promedio de rendimiento) para permitir algún tipo de evaluación de rendimiento de tareas, como se visualiza en la Figura 24.



The image shows a screenshot of a software dialog box titled "Task Properties". It has three tabs: "General", "Objects", and "Time Performance", with the "Time Performance" tab selected. The dialog box contains three input fields labeled "Min:", "Max:", and "Average:". At the bottom of the dialog box, there are four buttons: "Update", "Cancel", "Clear", and "Ok".

Figura 24. Propiedades de una Tarea

CAPÍTULO 4

CASO DE ESTUDIO

4.1 PROTOTIPO BANCA VIRTUAL

Luego de un análisis de los servicios de banca virtual que ofertan las principales entidades bancarias en el Ecuador como lo son:

- Banco de Pichincha
- Banco Guayaquil
- Banco Produbanco
- Banco General Rumiñahui.

Se identificó las tareas comunes que presenta dichas instituciones bancarias a sus clientes. Con base a este análisis se procedió a generar un modelo con las tareas que presentan a continuación.

Nombre de la Tarea: Identificación

Categoría: Interacción

Subtareas: Ingreso login, Control.

La Figura 25 muestra el árbol de identificación

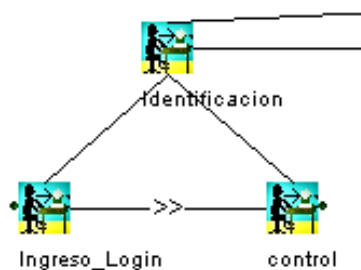


Figura 25. Árbol de Identificación

Nombre de la Tarea: Ingreso_Login
Categoría: Interacción
Descripción: Opción en la cual se solicita al usuario el ingreso de un identificador, su clave de acceso. Si la combinación de éstas es correcta, se habilita el acceso a la siguiente etapa de control.
Parámetros: Usuario: String Clave: String Login: Boolean
Precondiciones : Login=False

Nombre de la Tarea: Control
Categoría: Interacción
Descripción: En esta tarea se solicita la respuesta a una pregunta de seguridad y la figura correspondiente del usuario. Si una de los controles es incorrecto o nulo se dará como terminada la simulación ya que no paso las condiciones necesarias, caso contrario se habilita la página principal.
Parámetros de Control: Figura: String Pregunta: String Login: Boolean
Postcondiciones de la Tarea de Control (Pregunta!= AND Figura!= AND Login=true) OR (Pregunta= AND Figura= AND login=False)

Todas las tareas de la página principal tendrán como precondition que Login=True, es decir solo si acceden correctamente se habilitaran las opciones de la página principal.

Nombre de la Tarea: Página Principal
Categoría: Abstracta
Descripción: La página principal contiene las opciones de préstamo, transferencias, actualización de datos, cuentas, tarjetas, pago de servicios, perfil de acceso y por ultimo cerrar sesión, las misma que pueden ser accedidas en cualquier momento y sin un orden especifico ya que se encuentran enlazadas con el operador (parallel) que permite escoger cualquiera.
Subtareas: préstamo, transferencias, actualización de datos, cuentas, tarjetas, pago de servicios, perfil de acceso y por ultimo cerrar sesión.

El modelo a crear en la herramienta CTTE representara estas tareas comunes de las 4 bancas.

Como se puede observar en la figura 26 se presentan las tareas de préstamos y transferencias.

Nombre de la Tarea: Préstamo
Categoría: Aplicación
Descripción: Préstamo: Permitirá visualizar los préstamos contraídos con el banco.
Precondiciones: Login=True

Nombre de la tarea: Transferencias
Categoría: Abstracta
Descripción: existe de 2 tipos de transferencias las internas y las interbancarias, las internas tienen un monto menor e igual a 500, por otro lado las interbancarias tienen un monto menor e igual a 300.
Precondiciones: Login=True
Subtareas: Transferencias Internas y Transferencias Interbancarias.

Nombre de la Tarea: Transferencias Internas
Categoría: Abstracta
Descripción: Si exitoso es igual a True y Monto es menor e igual a 500, Se desplegará el mensaje de exitoso.
Parámetros: Monto: Decimal Cuenta: Integer Institución: String Beneficiario: String Identificación: Integer Ctaacreditar: Integer Detalle: String Login: Boolean Exitosa: Boolean
Precondiciones: Exitosa=False

Nombre de la Tarea: Transferencias Interbancarias
Categoría: Abstracta
Descripción: Si exitoso es igual a True y Monto es menor e igual a 300, Se desplegara el mensaje de exitoso.
Parámetros: Monto: Decimal Cuenta: Integer Concepto: String Exitosa: Boolean Login: Boolean

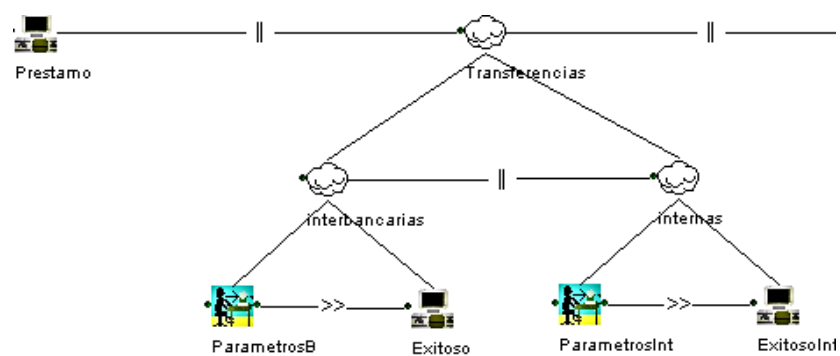


Figura 26. Árbol de Préstamo y Transferencias

Nombre de la Tarea: Actualización de Datos
Categoría: Interacción
Descripción: permite actualizar los datos personales del cliente.
Parámetros Estado civil: String Actividad: String Mail: String Celular: Integer Dirección: String Referencias: Text Login: Boolean
Precondiciones
Login= True

Nombre de la Tarea: Cuentas
Categoría: Abstracta
Descripción: permite visualizar la cuenta y sus movimientos, la condición necesaria es que el número de cuenta sea introducido y no sea null.
Parámetros Cuenta: Integer Fecha: Date Login: Boolean
Precondiciones
Login= True

Se visualiza los arboles antes mencionados en la Figura 27.

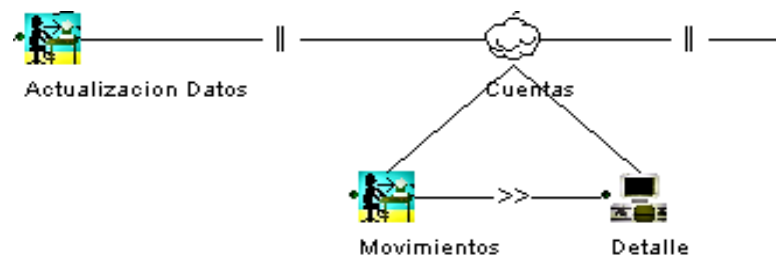


Figura 27. Actualización de datos y Cuentas

Nombre de la Tarea: Tarjetas
Categoría: Aplicación
Descripción: esta opción presenta información de las tarjetas del cliente.
Parámetros: Login= True

Nombre de la Tarea: Pago de Servicios
Categoría: Interacción
Descripción: permite el pago de servicios básicos como luz, agua, teléfono o tarjetas del mismo banco, las opciones serán enumeradas, así el usuario podrá escoger entre las 4 alternativas y así poder acceder a cualquiera de ellas.
Parámetros Login: Boolean Opción: Integer
Precondiciones Login= True

Árbol de Tarjetas y pago de servicios en la figura 28.

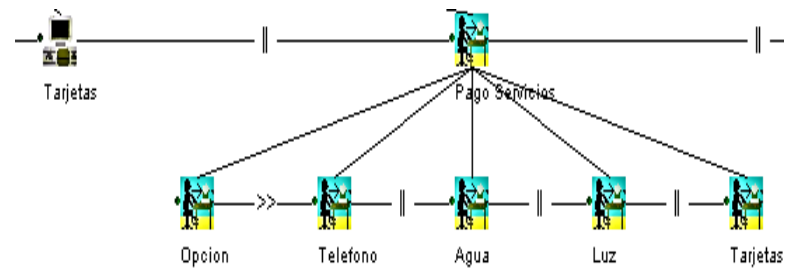


Figura 28. Árbol de Tarjetas y Pago de Servicios

Nombre de la Tarea: Perfil de Acceso
Categoría: Interacción
Descripción: permite hacer cambio de usuario o de la clave de la banca virtual, dependiendo de la opción que escoja el usuario se presentara el cambio del usuario o el cambio de la clave.
Parámetros Login: Boolean Usuario1: Boolean Clave1: Boolean
Precondiciones
Login= True AND clave1=False AND usuario1=False

La Figura 29 muestra el árbol mencionado.

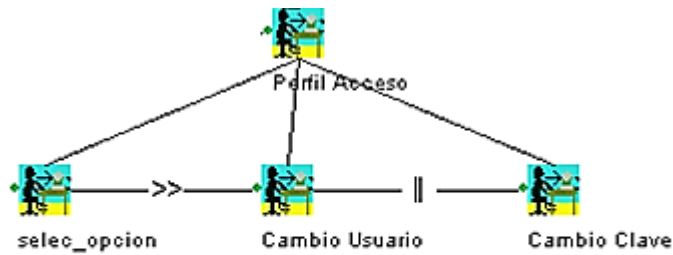


Figura 29. Árbol de Perfil de Acceso

Nombre de la Tarea: Cerrar sesión
Categoría: Abstracta
Descripción: Permite el cierre de sesión y da por terminado la simulación.
Parámetros: Login: Boolean Precondiciones: Login= True Postcondición: Login=False Pregunta esta NULL Figura esta NULL

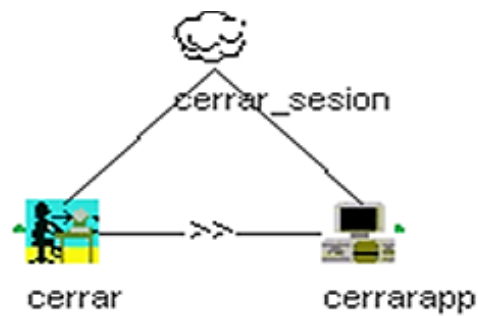


Figura 30. Árbol de Cerrar sesión

Una de las opciones de la herramienta CTTE es la de estadísticas del modelo de la banca virtual, se observa en la Figura 31.

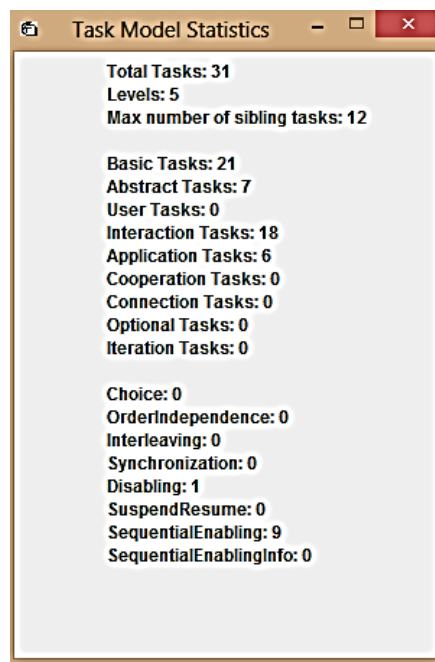


Figura 31. Estadísticas del Modelo

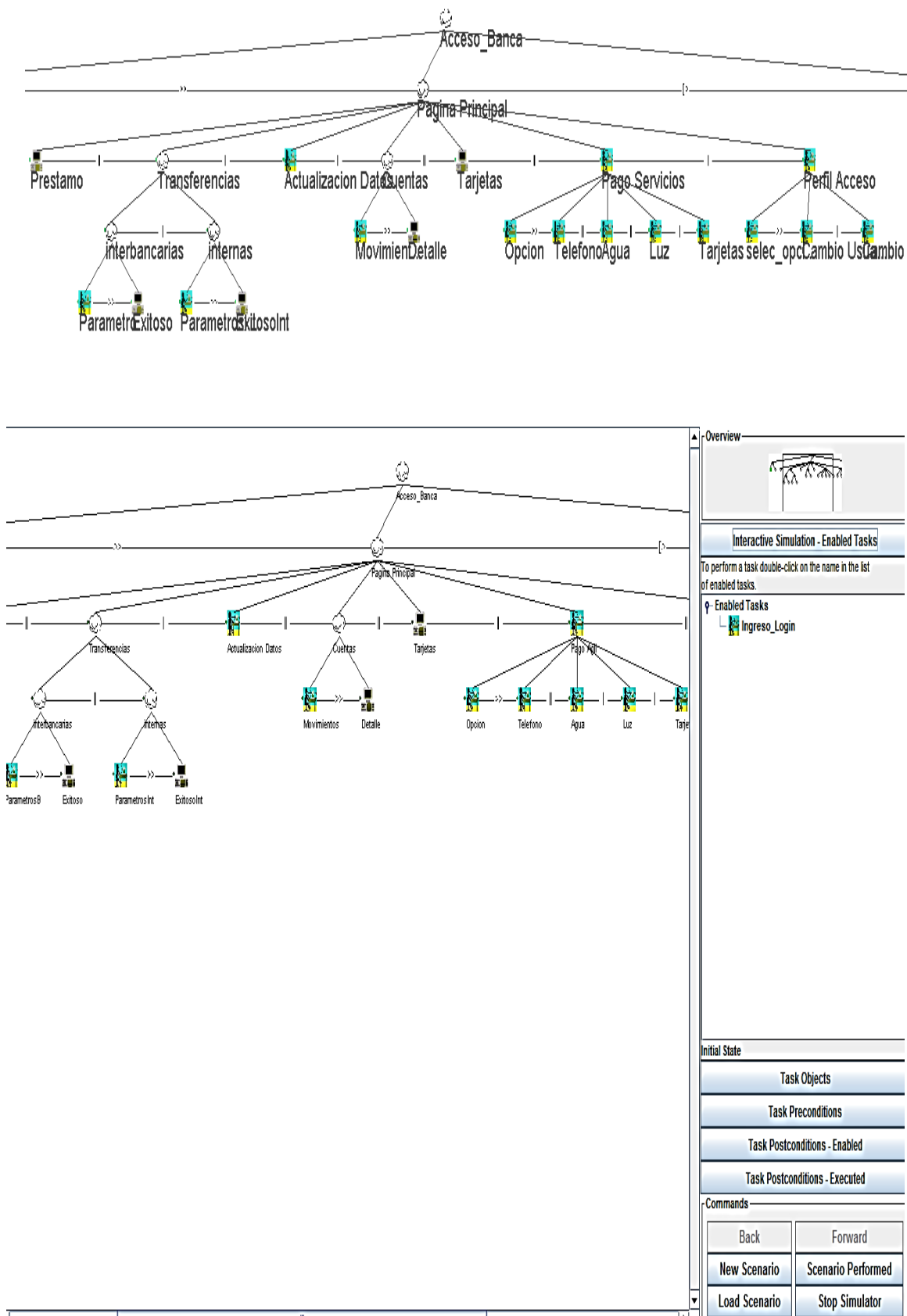


Figura 32. Simulador del Modelo

Se puede observar que la simulación es de gran ayuda ya que permite comprobar la secuencialidad de las tareas en base a las pre condiciones y post condiciones, que permite darse él cuenta el desempeño general ya del modelo.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se Realizó el análisis descriptivo de la notación ConcurTaskTrees.
- Se Analizó la Herramienta CTTE para aplicar la notación CTT.
- Se Creó un prototipo utilizando la herramienta CTTE.
- El modelado de tareas puede desempeñar un papel importante en el diseño de la interacción. Los procesos propuestos por varios autores como (Pressman, 2001),(Ferré, Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo),(Seffah A, 2003), concuerdan que el proceso de desarrollo de interacción inicia con el análisis de los usuarios y en el análisis de las tareas de usuario y que en conjunto con los requerimientos de usabilidad son los principales insumos para el proceso de diseño de prototipos de bajo y alto nivel. Finalmente el proceso de evaluación contrasta el producto obtenido con los resultados del proceso de análisis.
- ConcurTaskTrees es un lenguaje de modelado con una notación flexible y una semántica expresiva. Cuenta con un número reducido de elementos que lo hace sencillo de utilizar. Sin embargo presenta una gran potencialidad para representar toda la funcionalidad del sistema.
- CTTE usa el formato XML para la especificación del modelo de tareas, dando como ventaja su compatibilidad y estandarización con otras herramientas de modelado como lo es MARIAE.
- CTTE es una herramienta creada por el HIIS Laboratory de Italia. Es una herramienta relativamente nueva, sin embargo, en la actualidad cuenta con más de 3000 descargas en su página oficial.

- Se logró crear un modelo de banca virtual con las tareas comunes prestadas por las bancas más importantes del Ecuador.

5.2 RECOMENDACIONES

- Iniciar el proceso de diseño de interacción en etapas tempranas del proceso de desarrollo de software y relacionar a los diseñadores de interacción con los diseñadores de software, con la finalidad de mejorar la usabilidad del producto.
- Utilizar CTT como lenguaje de modelado para el análisis de tareas ya que es un lenguaje con pocos elementos pero que presenta una gran potencialidad para representar toda la interacción del sistema. En la mayoría de casos el modelo de tareas con CTT permite visualizar el modelo de navegación del sistema.
- Plantar investigaciones que permitan utilizar el código XML que genera CTTE y aplicarlo en otro tipo de herramientas que me permitan agregarle funcionalidad y generar prototipos funcionales de alto nivel de forma automática. Los prototipos generados pueden ser utilizados para elicitación de requerimientos, así como para la evaluación de la aplicación.
- El modelado de Tareas se debe llevar a la par del proceso de desarrollo de sistemas y en etapas tempranas. Como lo recomienda Ferré, se debe coordinar con partes del proceso de desarrollo de software para que no se dupliquen esfuerzos y que los desarrolladores tengan desde el inicio del proceso la idea clara del modelo de interacción que deben implementar.
- Se ha identificado algunas falencias que posee la herramienta CTTE como el simulador de tareas de la herramienta CTTE, que permite solo observar la interacción entre una tarea seleccionada y otra, pero no permite calcular el tiempo que toma realizar dicha tarea, dejando vacíos

a la hora de querer obtener mediciones del aprendizaje por parte del usuario.

- Publicar los resultados del presente trabajo de investigación para que la comunidad científica pueda mejorar el lenguaje de modelado y la herramienta CTTE.

BIBLIOGRAFÍA

(s.f.). *ISO 9241*.

Avilez, J. F. (2012). Análisis y Modelado de Tareas en los nuevos Paradigmas de Interacción Hombre-Máquina.

Bailey, G. (1993). Iterative Methodology and Designer Training in HumanComputer Interface Design. *In Human Factors in Computing Systems*.

Bolognesi, T., & Brinksma, E. (1987). *Introducción a la especificación ISO Idioma LOTOS*.

Booch, G., Rumbaugh, J., & Jacobson, I. (1999). El Lenguaje Unificado de Modelado. *Addison Wesley*.

Committee, I. S. (2001). *Guide to the Software Engineering Body of Knowledge*. IEEE Computer Society.

Consiglio Nazionale. (2010). *HIIS Laboratory*. Obtenido de <http://giove.cnuce.cnr.it/ctte.html>

Constantine, & Lockwood. (1999). *Software For Use: A Practical Guide to the Models and Methods of Usage-Centered Design*.

Correa, A. L. (s.f.). *Comunicabilidad, paradigma de la Interacción Humano-Computador*. Obtenido de <http://www.nosolousabilidad.com/articulos/comunicabilidad.htm>

D. Hix, H. H. (1993). Developing User Interfaces: Ensuring Usability Through Product and.

Ferré, X. (s.f.). Incorporación Técnicas de Interacción Persona-Ordenador al Proceso Desarrollo.

Ferré, X. (s.f.). Principios Básicos de Usabilidad para Ingenieros Software.

- Ferré, X., Juristo, N., H, W., & Constantine, L. (2001). Marco de integración de la usabilidad en el proceso de desarrollo software. *Usability Basics for Software* 8.
- Garrido, R. (2008). *campuscurico.atalca.c*. Obtenido de http://campuscurico.atalca.cl/~rgarrido/cursos/iuc_2008-2/doc/ppt_disenio.pdf
- Gea Miguel, L. G. (2002). El diseño.
- Iazza., Y. H. (s.f.). Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información.
- Lopez, V. (2005). Interfaces de Usuario adaptativas basadas en modelos y agentes de Software.
- Luis, C. A. (21 de 02 de 2010). *Comunicabilidad, paradigma de la Interacción Humano-Computador*. Obtenido de <http://www.nosolousabilidad.com/articulos/comunicabilidad.htm>
- Martínez, A. M. (2007). reTaskXML: Especificación de modelos de tareas a partir de especificaciones de interfaces de usuario .
- Molina, A. I., Redondo, M. A., & Ortega, M. (2007). Una revisión de notaciones para el modelado conceptual de sistemas interactivos para el soporte del trabajo en grupo.
- Myers, B., & Rosson. (1992). Survey on user interface programming. pág. 195-202.
- Negroponte, N. (1994). *Being Digital*. Nueva York: Vintage books.
- Newell, A. (1985). The prospects for psychological science in human-computer interaction. *Human-Computer Interaction*, 209-242.
- Nielsen, J. (1993). *Usability Engineering*. Academic Press.
- Paternò, F. (1999). Model-Based Design and Evaluation of Interactive Applications. *Springer*.

- Paternò, F. (2007). Review, Report and Refine Usability Evaluation Methods. págs. 49-51.
- Pressman, R. (2001). *Ingeniería del software: un enfoque práctico*.
- Seffah A, A. A. (2003). *Empowering Software Engineers in human Centered Design*.
- Simarro, B. A. (2007). coCTT: Modelado de tareas en un entorno colaborativo.
- Sommerville. (2005). *Ingeniería del Software*.
- Thimbleby, H. (1990). User interface design.
- Wegner, P. (1997). *Why interaction is more powerful than algorithms*.
Communications of the ACM.