

UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
MÁSTER EN TECNOLOGÍAS DE LA INFORMACIÓN



Estimación Basada en Casos de Uso
UCP – Use Case Points

ESTIMACIÓN DE PROYECTOS DE SOFTWARE

Presentado por:

- Olga María Recalde Maidana
- Alejandrina María Aranda López King

Matrícula: A00488

Matrícula: A00467

A: Ana María Morano

Madrid, 13 Enero del 2010

ÍNDICE

1. INTRODUCCIÓN	2
2. CONCEPTOS BÁSICOS	3
2.1. RUP	3
2.2. ACTORES	4
2.3. CASOS DE USO	6
3. PUNTOS DE CASOS DE USO (UCP – USE CASE POINTS)	7
3.1. CÁLCULO DE LOS PUNTOS DE CASO DE USO SIN AJUSTAR	8
3.1.1. <i>Factor de Peso de los Actores sin ajustar</i>	8
3.1.2. <i>Factor de Peso de los Casos de Uso sin ajustar</i>	9
3.2. CÁLCULO DE PUNTOS DE CASOS DE USO AJUSTADOS	10
3.2.1. <i>Factor de Complejidad Técnica (TCF)</i>	10
3.2.2. <i>Factores de Ambiente (EF)</i>	16
3.3. ESTIMACIÓN DEL ESFUERZO	18
4. EJEMPLO	19
4.1. MODELO DE CASOS DE USO	19
4.2. ESTIMACIÓN BASADA EN UCP	21
4.2.1. <i>Factor de Peso de los Actores sin ajustar</i>	21
4.2.2. <i>Factor de Peso de los Casos de Uso sin ajustar</i>	22
4.2.3. <i>Cálculo de los Puntos de Casos de Uso sin ajustar</i>	23
4.2.3. <i>Cálculo de los Puntos de Casos de Uso sin ajustar</i>	23
4.2.4. <i>Cálculo del Factor de Complejidad Técnico</i>	24
4.2.5. <i>Cálculo del Factor de Ambiente (EF)</i>	25
4.2.6. <i>Cálculo de los Puntos de Casos de Uso (UCP)</i>	25
4.2.7. <i>Cálculo del Esfuerzo</i>	26
5. CASOS DE USO Y PUNTOS DE FUNCIÓN	27
6. CONCLUSIÓN	29
7. REFERENCIAS	30

1. INTRODUCCIÓN

La especificación de los requisitos mediante Casos de Uso ha probado ser uno de los métodos más efectivos para capturar la funcionalidad de un sistema. Este hecho se puede apreciar en algunas metodologías actuales ampliamente difundidas, como el Proceso Unificado de Rational (Rational Unified Process - RUP) o Métrica Versión 3 (Ministerio de Administraciones Públicas de España), en las cuales se propone especificar la funcionalidad de los sistemas mediante la utilización de Casos de Uso.

Si bien los Casos de Uso permiten especificar la funcionalidad de un sistema bajo análisis, no permiten por sí mismos efectuar una estimación del tamaño que tendrá el sistema o del esfuerzo que tomaría implementarlo.

Para la estimación del tamaño de un sistema a partir de sus requisitos, una de las técnicas más difundidas es el **Análisis de Puntos de Función**. Ésta técnica permite cuantificar el tamaño de un sistema en unidades independientes del lenguaje de programación, las metodologías, plataformas y/o tecnologías utilizadas, denominadas Puntos de Función.

El método de Puntos de Caso de Uso (UCP - Use Case Point), está basado en los tradicionales Puntos de Función. Es un método originado de la tesis de máster de Gustav Karner (Karner, 1993), desarrollada mientras trabajaba en Objectory AB, bajo supervisión de Ivar Jacobson (creador de los casos de uso). La técnica ha sido usada por la empresa Rational (posteriormente adquirida por IBM) durante varios años y con buenos resultados. Además la técnica se ha documentado en varias publicaciones (Carroll, 2005; Clemmons, 2006; Karner, 1993; Nageswaran, 2007). [GAGA09]

2. CONCEPTOS BÁSICOS

Antes de entrar en detalle sobre la estimación basada en Puntos de Casos de Uso, se dedicará este apartado a presentar los conceptos básicos relacionados a Casos de Uso y algunos elementos que interactúan con ello. Específicamente, se dará una pequeña descripción sobre qué es la metodología RUP y los conceptos principales sobre actor y casos de uso.

2.1. RUP

Las siglas RUP en inglés significan *Rational Unified Process* (Proceso Unificado de Rational) es un proceso de ingeniería de software que proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización que se dedica al desarrollo. Su meta es asegurar la producción del software de alta calidad que resuelve las necesidades de los usuarios dentro de un presupuesto y tiempo establecidos.

RUP divide el proceso en cuatro fases (Inicio, Elaboración, Construcción, Transición), dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en los distintas actividades. En la siguiente figura se muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP. [DSIC]

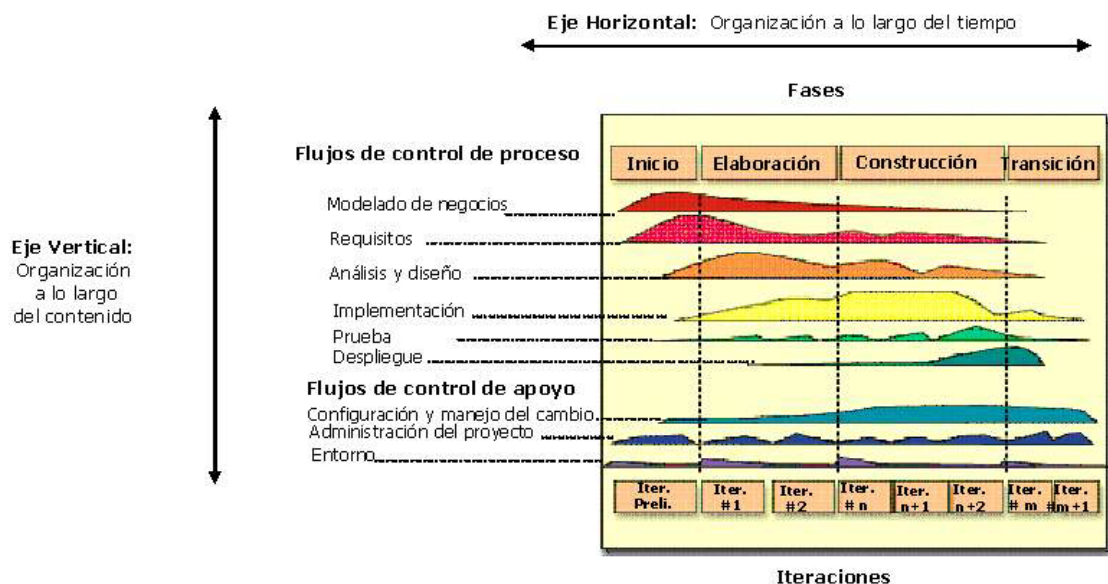


Figura 1 - Esfuerzo en actividades según fase del proyecto

RUP tiene tres características principales [DSIC], [RUP02]:

- **Proceso Dirigido por Casos de Uso:** utiliza los casos de uso para el desarrollo de las disciplinas con los artefactos, roles y actividades necesarias. Los casos de uso son la base para la implementación de las fases y disciplinas del RUP.

- **Proceso Centrado en la Arquitectura:** arquitectura de un sistema es la organización o estructura de sus partes más relevantes. Una arquitectura ejecutable es una implementación parcial del sistema, construida para demostrar algunas funciones y propiedades. RUP establece refinamientos sucesivos de una arquitectura ejecutable, construida como un prototipo evolutivo.
- **Proceso Iterativo e Incremental:** la estrategia que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos. Permitiendo que el equilibrio entre casos de uso y arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto. El ciclo de vida iterativo se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes.

2.2. Actores [RUP02]

Para comprender completamente la finalidad del sistema, es necesario saber para quién se desarrolla el sistema, es decir, quien lo usará. Tipos diferentes de usuarios son representados como actores.

Un actor es algo que intercambia datos con el sistema. Puede ser un usuario, un hardware externo u otro sistema.

La diferencia entre un actor y un usuario individual del sistema es que un actor representa una clase específica de usuario, en vez de ser un usuario real. Varios usuarios pueden desempeñar el mismo papel, lo que significa que puede ser el mismo actor. En este caso, cada usuario constituye una instancia de un actor.

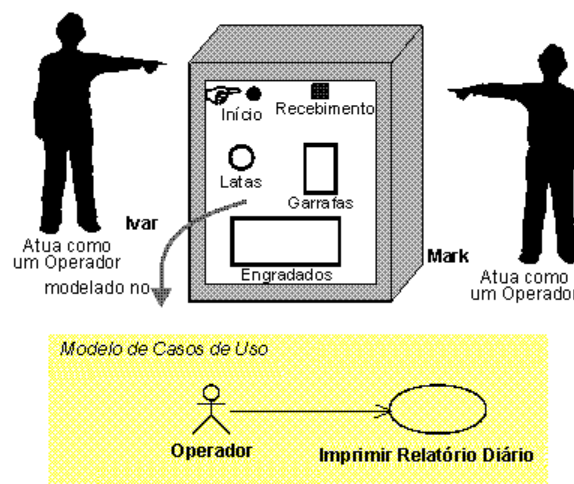


Figura 2 - Conceito de ator

Ivar y Mark son operadores de una máquina de reciclaje. Cuando están usando la máquina, cada uno es representado como una instancia del actor Operador.

Sin embargo, en algunas situaciones, solamente una persona desempeña el papel. Por ejemplo, puede existir solamente un individuo cuyo papel o rol sea de administrador de sistemas para un sistema relativamente pequeño.

El mismo usuario puede actuar también, como varios actores (es decir, la misma persona puede asumir diferentes papeles).

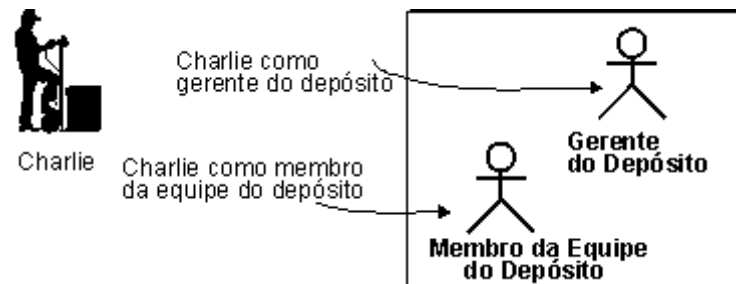


Figura 3 - Actor con varios roles

Charlie usa el sistema para Administrar un depósito principalmente como Gerente del depósito, pero algunas veces también utiliza el sistema como un miembro cualquiera del Equipo de Depósito.

- **¿Cómo identificar un actor?**

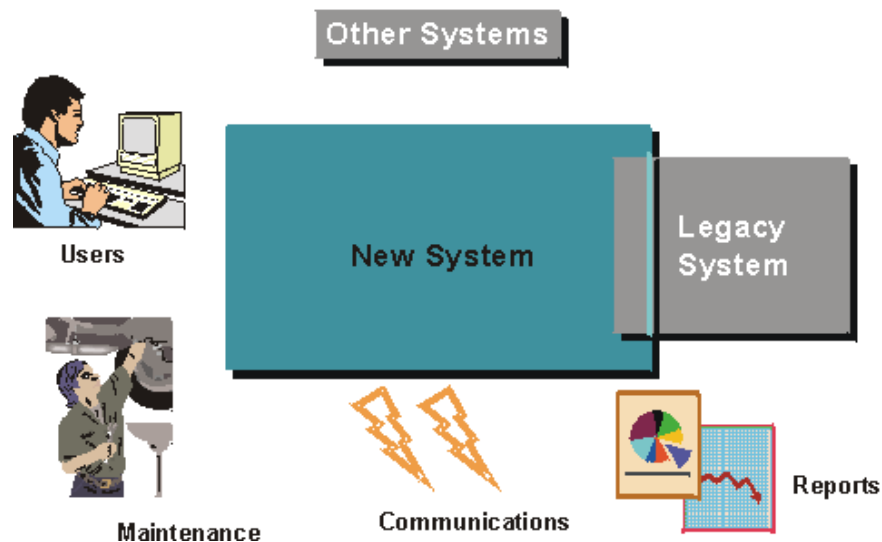


Figura 4 - Actores para un sistema

Se puede iniciar pensando qué individuos usarán el sistema. Es bueno hacerse las siguientes preguntas:

- ¿Quién va a proveer, usar o remover informaciones?
- ¿Quién usará esa funcionalidad?
- ¿Quién está interesado en un determinado requisito?
- ¿En qué parte de la organización el sistema es usado?
- ¿Quién va a dar soporte y mantenimiento al sistema?
- ¿Cuáles son los recursos externos al sistema?
- ¿Qué otros sistemas necesitarán interactuar con éste?

Existen varios aspectos diferentes al ambiente de un sistema, que se podrá representar como actores separados:

- Usuarios que ejecuten las principales funciones del sistema.
- Usuarios que ejecuten funciones secundarias al sistema, como la administración del sistema.
- Hardware externo que el sistema usa.
- Otros sistemas que interactúan con el sistema.

2.3. Casos de Uso

Los casos de uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Se define un caso de uso como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Los Casos de Uso representan los requisitos funcionales del sistema. En RUP los casos de uso no son sólo una herramienta para especificar los requisitos del sistema, sino que también guían su diseño, implementación y prueba. [DSIC]

▪ ¿Cómo identificar casos de uso?

Para poder identificar los casos de uso, uno puede plantearse alguna de las siguientes preguntas [RUP02]:

- ¿Para cada actor identificado, cuáles son las tareas en las cuales el sistema formaría parte?
- ¿El actor necesita estar informado sobre ciertas ocurrencias en el sistema?
- ¿El actor necesita informar el sistema sobre cambios externos repentinos?
- ¿El sistema provee al negocio el comportamiento correcto?
- ¿Todas las características pueden ser realizadas por los casos de usos identificados?
- ¿Qué tipos de casos de uso, supondrá y mantendrá el sistema?
- ¿Qué informaciones deben ser modificadas o creadas en el sistema?

3. PUNTOS DE CASOS DE USO (UCP – USE CASE POINTS)

El método de casos de uso permite documentar los requisitos de un sistema en términos de actores y casos de uso, proporcionando uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico. Debido a la importancia de esta técnica en el análisis orientado a objetos para capturar y describir los requisitos funcionales de un sistema, surgió la necesidad de desarrollar un método de cálculo del tamaño funcional específicamente diseñado para esta técnica [CUAD08]. El método “Puntos de Casos de Uso”, como se mencionó en la introducción lo obtuvo Gustva Karner en 1993 y posteriormente fue refinado por varios autores.

Por tanto, la **estimación mediante el análisis de Puntos de Casos de Uso**, consiste en un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de “pesos” a un cierto número de factores que lo afectan, para que luego a partir de esos factores se pueda contabilizar el tiempo total estimado para el proyecto en cuestión. [PERA09]

En la siguiente figura, se pueden observar los principales pasos del método Estimación Puntos de Casos de Uso.

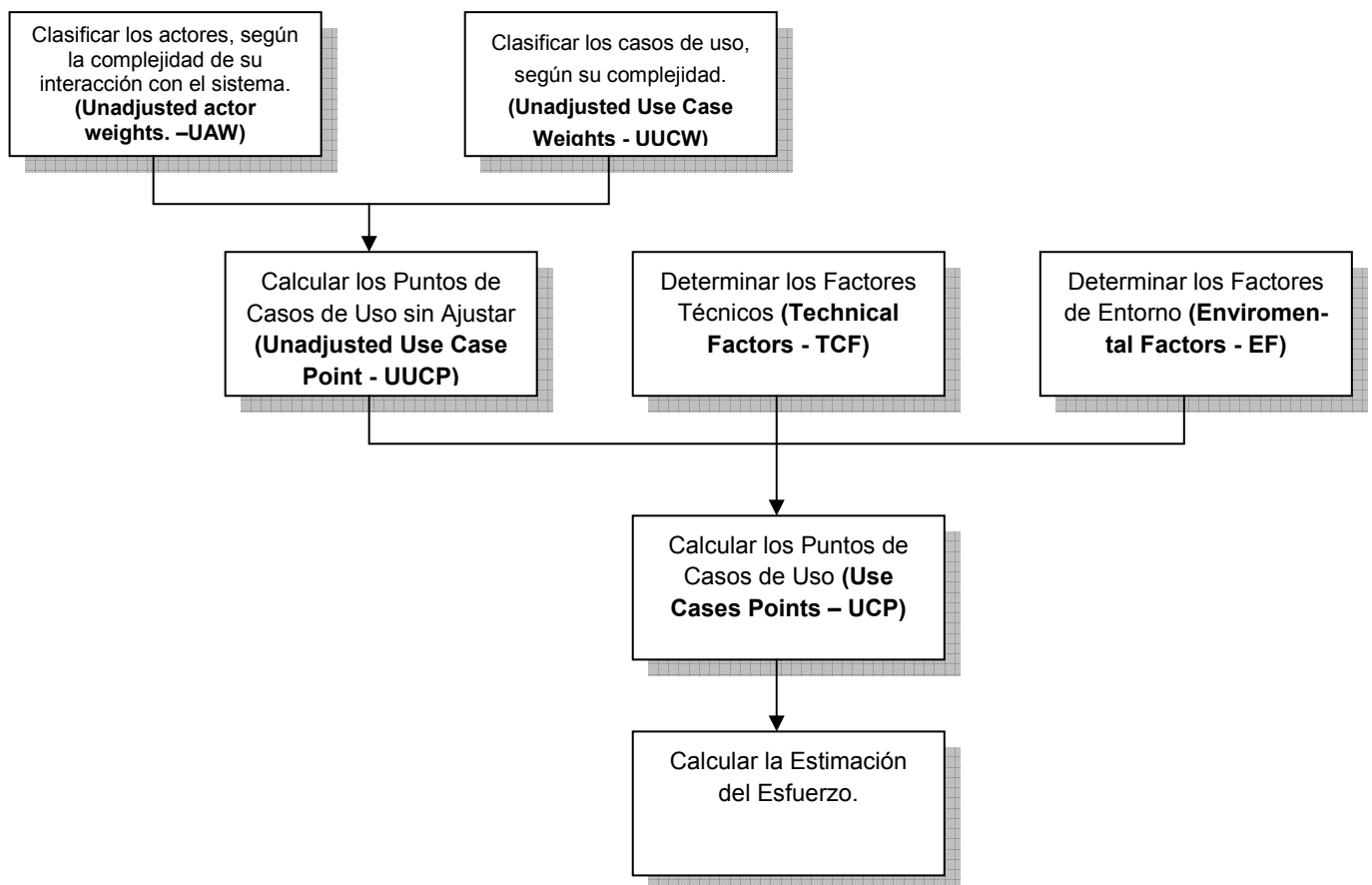


Figura 5 - Pasos - Estimación basada en Puntos de Casos de Uso

3.1. Cálculo de los Puntos de Caso de Uso sin ajustar

El primer paso para la estimación consiste en el cálculo de los Puntos de Casos de Uso sin ajustar. Este valor, se calcula a partir de la siguiente ecuación:

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

Donde,

- **UUCP:** Puntos de Casos de Uso sin ajustar
- **UAW:** Factor de Peso de los Actores sin ajustar
- **UUCW:** Factor de Peso de los Casos de Uso sin ajustar

A continuación se describen cómo obtener los factores de peso relacionados a los actores y casos de uso respectivamente, para que posteriormente se pueda obtener los Puntos de Casos de Uso sin ajustar.

3.1.1. Factor de Peso de los Actores sin ajustar

Para obtener el valor del factor de peso de los actores, se debe calcular la cantidad de actores existentes en el sistema y la complejidad de cada uno de ellos.

La complejidad de los actores se establece teniendo el **tipo de actor**, pudiendo ser éste: simple, medio y complejo; así como también teniendo en cuenta la forma que el mismo **interactúa con el sistema**, asignando el peso correspondiente asociado a dicha interacción.

Tipo de Actor	Descripción	Peso
Simple	Otro sistema externo, interactúa con el sistema a desarrollar mediante una interfaz de programación definida y conocida, (API, Application Programming Interface)	1
Medio	Otro sistema externo que interactúa a través de protocolo (conjunto de reglas que especifican el intercambio de datos u órdenes durante la comunicación entre las entidades que forman parte de la red), Ej: TCP/IP.	2
Complejo	Un usuario físico que interactúa a través de una interfaz gráfica de usuario.	3

Tabla 1 - Clasificación de Actores [GAGA09] [CUAD08]

Por tanto, esta clasificación da como resultado el número total de UAW (Unadjusted Actor Weight), contando el número de actores existentes en cada tipo, multiplicando cada resultado por el factor de ajuste o peso correspondiente y realizando la suma de los resultados.

$$UAW = \sum (Cantidad\ TipoActor_i * Peso_i)$$

3.1.2. Factor de Peso de los Casos de Uso sin ajustar

Para obtener el valor del factor de Peso de los casos de uso, se calcula la cantidad de casos de uso presentes en el sistema y la complejidad de cada uno de ellos.

La complejidad de los casos de uso se establece teniendo en cuenta el **tipo de caso de uso**, pudiendo ser éste: simple, medio y complejo; así como también teniendo en cuenta el número de transacciones efectuadas en el mismo. Se entiende transacción [GAGA09] por el conjunto de actividades atómicas, que se pueden ejecutar ya sean todas de forma completa o ninguna.

La siguiente tabla, muestra los criterios a tener en cuenta para clasificar los casos de uso:

Tipo de Caso de Uso	Descripción	Peso
Simple	El caso de Uso contiene menos de 3 transacciones	5
Medio	El caso de Uso contiene de 4 a 7 transacciones	10
Complejo	El caso de uso tiene más de 7 transacciones	15

Tabla 2 - Clasificación de los Casos de Uso

La clasificación de los casos de uso permite conocer el valor del UUCW (Unadjusted Use Case Weights) de manera análoga al caso anterior, mediante el conteo del número de casos de uso en cada tipo, multiplicando éstos por el factor de ajuste correspondiente y realizando la suma de los resultados.

$$UUCW = \sum (TipoCasoDeUso_i * Peso_i)$$

3.2. Cálculo de Puntos de Casos de Uso ajustados

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar ese valor mediante factores de ajuste, tanto técnicos como de ambiente, haciendo uso de la siguiente ecuación:

$$UCP = UUCP * TCF * EF$$

Donde,

- **UCP:** Puntos de Casos de Uso ajustados
- **UUCP:** Puntos de Casos de Uso sin ajustar
- **TCF:** Factor de complejidad técnica
- **EF:** Factor de ambiente

Se debe tener en cuenta que a través del cálculo de esta expresión obtenemos una estimación del tamaño y no del esfuerzo [GAGA09]. El cálculo de la estimación del esfuerzo se detallará en el punto 3.3.

3.2.1. Factor de Complejidad Técnica (TCF)

El coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 al 5, donde:

- Un valor de 0 significa que el factor es irrelevante.
- Un valor de 3 significa que el factor es promedio.
- Un valor de 5 significa que el factor es esencial.

Dichos valores varían de acuerdo con la complejidad del proyecto.

Una vez que todos los factores técnicos tienen asignados el valor de influencia, se procede al cálculo de los resultados de cada factor, es decir, se realiza una multiplicación entre la influencia (valor asignado) y su peso asociado. Conforme se puede visualizar en la siguiente tabla:

Factor	Descripción	Peso	Influencia	Resultado
F1	Sistema Distribuido	2	n1	2*n1
F2	Tiempo de Respuesta y Desempeño	1	n2	1*n2
F3	Eficiencia respecto al usuario final	1	n3	1*n3
F4	Procesamiento interno complejo	1	n4	1*n4
F5	Código reutilizable en otras aplicaciones	1	n5	1*n5
F6	Facilidad en la instalación	0,5	n6	0,5*n6

Factor	Descripción	Peso	Influencia	Resultado
F7	Usabilidad (Fácil de usar)	0,5	n7	0,5*n7
F8	Portabilidad	2	n8	2*n8
F9	Facilidad en mantener	1	n9	1*n9
F10	Acesos simultáneos (concurrentes)	1	N10	1*n10
F11	Incluye objetivos especiales de seguridad	1	N11	1*n11
F12	Provee acceso directo a terceros	1	N12	1*n12
F13	Se requiere facilidades especiales de entrenamiento a usuarios	1	N13	1*n13

Tabla 3 – Cálculo de los Factores Técnicos

Karner, G. (1993) y Schneider, G.; Winters, J. P. (2001, p. 154),

Por tanto, el factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$TCF = 0.6 + 0.01 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

A continuación, el significado de cada uno de los factores, mencionados en la tabla anterior, los mismos fueron descritos por [DOSS09]:

F1 - Sistema Distribuido

La arquitectura puede ser centralizada como también distribuida.

- **0:** La aplicación no auxilia en la transferencia de datos o procesos entre CPUs.
- **1:** La Aplicación prepara datos para que el usuario final los procese en otra CPU.
Ejemplo: planillas electrónicas o gestores de bases de datos.
- **2:** Los datos son preparados para ser transferidos y procesados en otra CPU.
- **3:** Procesamiento distribuido y transferencia de datos online en una dirección.
- **4:** Procesamiento distribuido y transferencia de datos online en ambas direcciones.
- **5:** Las funciones de procesamiento son ejecutadas dinámicamente en la CPU más apropiada.

F2 - Tiempo de Respuesta y Desempeño

Identifica los objetivos de Desempeño de la aplicación, establecidos y aprobados por el usuario.

- **0:** Ninguna exigencia especial de desempeño fue establecida por el usuario.
- **1:** Requisitos de desempeño fueron establecidos y revisados por el usuario, pero ninguna acción especial fue necesaria.

- **2:** El tiempo de respuesta es crítico durante horas pico. El intervalo de tiempo límite (deadline) del procesamiento es siempre para el próximo día útil. Ninguna consideración especial para la utilización de la CPU fue requerida.
- **3:** El tiempo de respuesta es crítico durante todo el tiempo de utilización de la aplicación. Los requisitos de plazo de procesamiento con otros sistemas son limitados. No fue necesario ningún procedimiento especial para la utilización de la CPU.
- **4:** Los requisitos de desempeño establecidos por el usuario son rigurosos y bastante para requerir tareas de análisis de desempeño en la fase de análisis y proyecto de la aplicación.
- **5:** Además de lo descrito en el ítem 4, herramientas de análisis de desempeño fueron usadas en las fases de proyecto, desarrollo y/o implementación con el fin de proporcionar el desempeño establecido por el usuario.

F3 - Eficiencia respecto al Usuario Final

Las funciones proporcionadas, enfatizan una aplicación para la eficiencia del usuario final. Ejemplo: menú, documentación de ayuda online, movimiento automático del cursor, movimiento de la pantalla (scrolling) vertical y horizontal, teclas de función predefinidas, facilidad navegación entre pantallas, número mínimo de pantallas para ejecutar una función de negocio, impresiones remotas, soporte multilingüe (más de dos idiomas), entre otras.

- **0:** La aplicación no presenta ninguno de los ítems mencionados anteriormente.
- **1:** Presenta 1 a 3 de los ítems mencionados arriba.
- **2:** Presenta 4 a 5 de los ítems mencionados arriba.
- **3:** Presenta 6 o más de los ítems mencionados arriba, pero no hay ningún requisito relacionado a la eficiencia.
- **4:** Presenta 6 o más de los ítems mencionados arriba, y los requisitos establecidos para la eficiencia son riguroso y suficientes para que la fase de proyecto de la aplicación incluya factores para minimizar la digitación, maximizar los defaults, utilizar templates, entre otros.
- **5:** Presenta 6 o más de los ítems mencionados anteriormente, y los requisitos establecidos para la eficiencia del usuario son rigurosos o suficientes para que sea necesario el uso de herramientas y procesos especiales para demostrar que los objetivos de eficiencia fueron alcanzados.

F4 - Procesamiento Interno Complejo

La complejidad de procesamiento tiene influencia en el dimensionamiento del sistema, y, por tanto, debe ser cuantificado el grado de influencia con base en las siguientes categorías:

- Procesamiento especial de auditoría y/o procesamiento especial de seguridad.
- Procesamiento lógico extensivo.

- Procesamiento matemático extensivo.
- Gran cantidad de procesamiento de ejecución, resultante de transacciones incompletas que necesitan de reprocesamiento. Por ejemplo: transacciones incompletas de ATM (cajeros automáticos) causados por interrupciones de comunicación, valores de datos ausentes o validaciones de errores.
- Procesamiento complejo para manipular múltiples posibilidades de entrada/salida. Por ejemplo: múltiples medios e independencia de equipos.
 - **0:** No presenta ningún ítem mencionado anteriormente.
 - **1:** Presenta uno de los ítems de arriba.
 - **2:** Presenta dos de los ítems de arriba.
 - **3:** Presenta tres de los ítems de arriba.
 - **4:** Presenta cuatro de los ítems de arriba.
 - **5:** Presenta todos los ítems mencionados en el párrafo anterior.

F5 - Código Reutilizable en otras Aplicaciones

La aplicación y su código fuente, fueron específicamente proyectados, desarrollados y soportados para que sean reutilizables en otras aplicaciones.

- **0:** No presenta código reutilizable.
- **1:** El código reutilizable es usado solamente dentro de la aplicación.
- **2:** Menos del 10% de la aplicación fue hecha, teniendo en cuenta su utilización en otras aplicaciones.
- **3:** 10% o más de la aplicación fue hecha, teniendo en cuenta su utilización en otras aplicaciones.
- **4:** La aplicación fue proyectada y documentada para facilitar la reutilización de código y la aplicación es personalizada por el usuario a nivel de código fuente.
- **5:** La aplicación fue proyectada y documentada para facilitar la reutilización del código fuente.

F6 - Facilidad en la Instalación

Indica el nivel de preparación de procedimientos y herramientas para la instalación del sistema.

- **0:** Ninguna consideración se ha tenido en cuenta por el usuario y ningún procedimiento especial fue requerido para la instalación.
- **1:** Ninguna consideración especial se ha tenido en cuenta por el usuario, pero un procedimiento especial fue requerido para la instalación.
- **2:** Requisitos de instalación fueron establecidos por el usuario.
- **3:** Requisitos de instalación fueron fijados por el usuario y scripts de instalación fueron preparados y probados.

- **4:** Además de lo descrito en el ítem 2, herramientas automatizadas de instalación fueron preparadas y probadas.
- **5:** A demás de los descrito en el ítem 3, herramientas automatizadas de instalación fueron preparadas y probadas.

F7 - Usabilidad

Procedimientos efectivos de instalación, backup y recuperación fueron desarrollados y probados. La aplicación minimiza la necesidad de actividades manuales.

- **0:** Ninguna consideración especial sobre la facilidad operacional, además de los procedimientos normales de backup, fue tomada en cuenta por el usuario.
- **1:** Procedimientos eficientes de inicialización, backup y recuperación fueron preparados, pero la intervención del operador es necesaria.
- **2:** Procedimientos eficientes de inicialización, backup y recuperación fueron preparados, pero ninguna intervención del operador es necesaria.
- **3:** La aplicación minimiza la operación de montaje de cintas magnéticas.
- **4:** La aplicación minimiza la necesidad de manejo de formularios.
- **5:** La aplicación fue proyectada de manera que ningún operador intervenga en el funcionamiento normal.

F8 - Portabilidad

La aplicación fue específicamente proyectada, desarrollada y soportada para ser instalada en múltiples plataformas (Windows, Unix, Linux...).

- **0:** El usuario no ha solicitado considerar la necesidad de instalar la aplicación en más de una plataforma.
- **1:** La necesidad de instalación en múltiples plataformas fue llevada en consideración en el proyecto y la aplicación fue proyectada para operar solamente en ambientes idénticos de hardware y software.
- **2:** La necesidad de instalación en múltiples plataformas fue llevada en consideración en el proyecto y la aplicación fue proyectada para operar solamente en ambientes similares de hardware y software.
- **3:** La necesidad de instalación en múltiples plataformas fue llevada en consideración en el proyecto y la aplicación fue proyectada para operar en ambientes diferentes.
- **4:** Un plan de documentación y mantenimiento fue elaborado y probado para soportar la aplicación en múltiples plataformas y la aplicación atiende los ítems 1 y 2.
- **5:** Un plan de documentación y mantenimiento fue elaborado y probado para soportar la aplicación en múltiples plataformas si la aplicación atiende el ítem 3.

F9 - Fácil de Mantener

La aplicación fue específicamente proyectada, desarrollada para soportar el mantenimiento, garantizando la facilidad de cambios.

- **0:** No fue considerado por el usuario, ningún requisito especial para proyectar la aplicación que garantice minimizar o facilitar los cambios.
- **1:** Recursos de consultas/informes flexibles son proporcionados, de manera que sea capaz de manipular solicitudes simples de consulta (Query/requests). Ejemplo: Lógica de and or aplicada solamente aun Archivo Lógico Interno (contar como un ítem).
- **2:** Recursos de consultas/informes flexibles son proporcionados, de manera que sea capaz de manipular solicitudes de consulta (Query/requests) de complejidad media. Ejemplo: Lógica de and/or aplicada a más de un Archivo Lógico Interno (contar como dos ítems).
- **3:** Recursos de consultas/informes flexibles son proporcionados, de manera que sea capaz de manipular solicitudes consulta (Query/requests) complejas. Ejemplo: Combinaciones de lógica de and/or aplicadas a uno o más de un Archivo Lógico Interno (contar como tres ítems).
- **4:** Datos de control son mantenidos en tablas que son actualizadas por el usuario a través de procesos online e iterativos, pero las alteraciones sólo son actualizadas por el usuario a través de procesos online e iterativos, pero las modificaciones sólo son efectivas en el próximo día útil.
- **5:** Datos de control son mantenidos en tablas que pueden ser actualizadas por el usuario a través de procesos online e iterativos, y las modificaciones son efectivizadas inmediatamente (contar como dos ítems).

F10 - Accesos Simultáneos (Concurrentes)

Indica el volumen de acceso simultáneo a la aplicación.

- **0:** No es esperado acceso simultáneo.
- **1:** Son esperados accesos simultáneos esporádicamente.
- **2:** Accesos simultáneos son esperados.
- **3:** Accesos simultáneos son esperados diariamente.
- **4:** Muchos accesos simultáneos fueron fijados por el usuario para la aplicación, lo que fuerza la ejecución de tareas de análisis de desempeño en la fase de proyecto de la aplicación.
- **5:** Requiere el uso de herramientas que controlen el acceso en las fases de desarrollo e implementación, además de las consideraciones anteriores.

F11 - Características Especiales de Seguridad

Indica el nivel de seguridad exigido por la aplicación.

- **0:** Ningún requisito por parte del usuario fue solicitado para considerar la necesidad de control de seguridad en la aplicación.
- **1:** Fue considerada en el proyecto del sistema, la necesidad de control de seguridad.
- **2:** Fue considerada en el proyecto del sistema, la necesidad de control de seguridad y la aplicación fue proyectada para que solamente usuarios autorizados puedan acceder.
- **3:** Fue considerada en el proyecto del sistema, la necesidad de control de seguridad y la aplicación fue proyectada para que solamente usuarios autorizados puedan acceder. El acceso será controlado y auditado.
- **5:** Un plan de seguridad fue elaborado y probado para soportar el control de acceso a la aplicación y una auditoría.

F13 - Entrenamiento a los Usuarios

Indica el nivel de facilidad de proveer capacitación a los usuarios para utilizar el software.

- **0:** Ningún pedido por parte del usuario para considerar la necesidad de entrenamiento especial.
- **1:** La necesidad de capacitación especial, fue llevada en consideración en el proyecto del sistema.
- **2:** La necesidad de capacitación fue llevada en consideración en el proyecto del sistema y una aplicación fue proyectada para que los usuarios puedan acceder con facilidad.
- **3:** La necesidad de capacitación especial, fue llevada en consideración en el proyecto del sistema y una aplicación fue proyectada para que los usuarios pueda utilizar en diversos niveles
- **4-5:** Un plan de capacitación fue elaborado y probado para facilitar el uso de la aplicación.

3.2.2. Factores de Ambiente (EF)

Además de tener en cuenta los factores técnicos para el ajuste de los UUCP (Puntos de Casos de Uso no ajustados), se contabilizan los factores de ambiente. De manera similar al cálculo de los TCF, a cada factor de ambiente definido en la siguiente tabla, se la asignan valores entre el 0 y el 5.

Una vez que todos los factores de ambiente tienen asignado el valor de la influencia, se procede al cálculo de los resultados de cada factor, es decir, se realiza una multiplicación entre la influencia del factor y su peso asociado.

En la siguiente tabla se presenta un resumen del procedimiento del cálculo de los factores de ambiente, siendo Ei los factores concretos.

Factor	Descripción	Peso	Influencia	Resultado
E1	Familiarizado con el proceso de desarrollo (RUP)	1,5	n1	$R1=1,5*n1$
E2	Experiencia en la aplicación	0,5	n2	$R2=0,5*n2$
E3	Experiencia en orientación a objetos	1	n3	$R3=1*n3$
E4	Capacidades de análisis	0,5	n4	$R4=0,5*n4$
E5	Motivación	1	n5	$R5=1*n5$
E6	Requisitos estables	2	n6	$R6=2*n6$
E7	Trabajadores a tiempo parcial	-1	n7	$R7=-1*n7$
E8	Lenguaje complejo	-1	n8	$R8=-1*n8$

Tabla 4 - Cálculo los Factores de Ambiente

- Para los factores E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.
- Para el factor E6, 0 significa requisitos extremadamente inestables, 3 estabilidad media y 5 requisitos estables sin posibilidad de cambios.
- Para el factor E7, 0 significa que no hay personal tiempo parcial (es decir todos son de tiempo completo), 3 significa mitad y mitad (es decir, la empresa cuenta con personal que trabaja, medio período y otros tiempo completo), y 5 significa que todo el personal trabaja medio período (nadie período completo).
- Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil.

Cuando se han calculado los resultados de cada uno de los factores, se aplica la ecuación descrita a continuación:

$$EF = 1.4 - 0.03 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

3.3. Estimación del Esfuerzo

Karner originalmente sugirió que cada Punto de Casos de Uso requiere **20 horas-hombre**. Posteriormente, surgieron otras propuestas como la de **Barnerje** que propone un rango entre **15 y 30 horas**, o la de **Scheider y Winters**, que sugieren otros refinamientos de los factores de ambiente, que proponen una granularidad algo más fina, basándose en los siguientes criterios (Factor de Conversión) [PERA09]:

- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por debajo del valor medio (3), para los factores E1 a E6.
- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por encima del valor medio (3), para los factores E7 y E8.
- Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre.
- Si el total es 3 o 4, se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 28 horas-hombre.
- Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

Entonces se tiene [GAGA09]:

- 20 horas-hombre por UCP si el valor es ≤ 2
- 28 horas-hombre por UCP si el valor es ≤ 4
- 36 horas-hombre por UCP si el valor es ≥ 5 , en este caso se debería replantear el proyecto.

El esfuerzo en horas-hombre se calcula aplicando la siguiente ecuación:

$$E = UCP * CF$$

Donde,

- **E**: esfuerzo estimado en horas-hombre
- **UCP**: Puntos de Casos de Uso ajustados
- **CF**: factor de conversión (Productividad)

Cabe destacar, que el valor del esfuerzo estimado, no cubre todas las fases del ciclo de vida del proyecto, sino que proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso. La fase de codificación representa un 40% del esfuerzo total del proyecto.

En este sentido, para obtener el esfuerzo total del proyecto, se puede realizar un nuevo ajuste que consiste en sumar a la estimación de esfuerzo obtenida por UCP, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo del software, que se pueden distribuir de la siguiente forma: análisis 10%, diseño 20%, codificación 40%, pruebas 15% y sobrecarga 15%. [GAGA09]

Con éste criterio, y tomando como entrada la estimación de tiempo calculada a partir de los Puntos de Casos de Uso, se pueden calcular las demás estimaciones para obtener la duración total del proyecto.

4. EJEMPLO

El siguiente ejemplo, muestra un caso práctico, en el que se aplican y explican los pasos mencionados en los puntos anteriores, necesarios para obtener los Puntos de Casos de Uso.

4.1. Modelo de Casos de Uso

Los diagramas de casos de uso no sólo son útiles en la identificación de requisitos y comportamiento del sistema, sino que también tienen utilidad en la planificación.

El sistema que ha sido estimado, ya desarrollado por un grupo de alumnos hace unos años, tiene por objetivo mantener y consultar los datos relacionados con los alumnos y profesores de una autoescuela. Gestionar los clientes, deudas y pagos, clases prácticas, consultas estadísticas. Y por medio de una aplicación web, permitir a los alumnos la realización de tests, consultas de manuales del conductor, modificar sus datos personales, obtener informaciones de la autoescuela, entre otras.

Los autores que han identificado y que interactúan con el sistema son los siguientes:

- **Internauta:** Este actor representa a las personas que acceden a la página web de la autoescuela buscando información.
- **Alumno:** Este actor representa a los alumnos ya validados en la autoescuela, que acceden a la página web para usar alguno de los servicios que esta ofrece.
- **Solicitante profesor:** Este actor representa a las personas que acceden a la página web para enviar a la autoescuela todos los datos de su curriculum con el fin de que sean tenidos en cuenta en posibles futuras contrataciones.
- **Empleado:** Este usuario es el encargado de realizar la mayoría de las tareas que se ejecutan a diario desde la aplicación, tales como la gestión de clientes, de profesores, de clases prácticas, etc.

- **Administrador:** Es el encargado de realizar tareas puntuales concernientes al funcionamiento global de la autoescuela, como pueden ser la gestión de precios, la realización del cambio de moneda y la configuración de los tests.

Los casos de usos identificados, se listan más adelante. El detalle de los mismos se podrán encontrar en el siguiente sitio web [GOAL01], en este documento a modo de ejemplo de tomará uno “Alta de Clientes”.

RF – 1	Alta de Cliente.	
Objetivos asociados	OBJ-1: Gestión de clientes	
Descripción	El sistema deberá permitir al empleado en cualquier momento dar de alta a un nuevo cliente según se explica en el siguiente caso de uso:	
Precondición	Ninguna.	
Secuencia normal	Paso	Acción
	1	Este caso de uso comienza cuando un empleado elige la opción de “Alta de cliente”.
	2	Si el empleado desea consultar los precios actuales, se realiza el caso de uso RF-45 (Consultar Precios Actuales).
	3	El sistema solicita los datos del nuevo cliente: nombre, apellidos, <i>dni</i> , sexo, fecha de nacimiento, dirección, localidad, provincia, código postal, teléfono, <i>e-mail</i> .
	4	El sistema permite al empleado elegir el centro de Formación en el que se encuentra.
	5	El empleado proporciona los datos requeridos al sistema y elige aceptar alta.
	6	Si no existe ese cliente, el sistema lo registra como nuevo cliente.
	7	El sistema registra la matricula de ese cliente, sin validarla
	8	El sistema muestra un mensaje de alta realizado correctamente
Postcondición	El cliente queda registrado.	
Excepciones	Paso	Acción
	2-5	Si el empleado cancela la operación, este caso de uso aborta.
	6-8	Si el sistema detecta que el cliente esta registrado como alumno actualmente o ya aprobó el carnet, se informa de esta situación al empleado y a continuación este caso de uso aborta.
Importancia	Vital.	
Comentarios	Un cliente puede tener varias matriculas siempre que se dé de baja sin haber aprobado.	

4.2. Estimación basada en UCP

Aplicando el método “Puntos de Casos de Uso - UCP”, se llegará a una estimación de horas hombre que deberán ser empleadas para la realización del proyecto.

Los pasos que se seguirán para obtener la estimación basada en UCP serán los siguientes.

1. Identificar los actores que interactúan con el sistema y calcular el factor de peso de los mismos.
2. Identificar los casos de uso y calcular el factor de peso de los mismos.
3. Calcular los puntos de Casos de Uso sin ajustar.
4. Calcular el factor técnico de complejidad.
5. Calcular el factor de ambiente
6. Calcular los puntos de Casos de Uso ajustados
7. Calcular la estimación del Esfuerzo

4.2.1. Factor de Peso de los Actores sin ajustar

En primer lugar se calcula el peso de los **actores**, que será 1, 2 ó 3 dependiendo de si se trata de actores simples, promedios o complejos.






Actor	Tipo Actor	Descripción	Peso
 solicitante profesor	Promedio	Utiliza una interfaz simple en modo texto.	2
 internauta	Promedio	Utiliza una interfaz simple en modo texto.	2
 alumno	Complejo	Interactúa con una GUI.	3
 empleado	Complejo	Interactúa con una GUI.	3
 administrador	Complejo	Interactúa con una GUI.	3

Tabla 5 – Identificación de los Actores

Aplicando la siguiente fórmula:

$$UAW = \sum (\text{Peso}_i * \text{Cantidad TipoActor}_i)$$

Tenemos que el factor del Peso sin ajustar es:

$$UAW = (2 * 2) + (3 * 3) = 4 + 9 = 13$$

$$\underline{UAW = 13}$$

4.2.2. Factor de Peso de los Casos de Uso sin ajustar

A continuación se determina el factor de Peso de los **Casos de Uso** en función de si se trata de casos de uso simples, promedios o complejos. Sólo se consideran los casos de uso que interactúan con actores.

Una transacción es cada una de las etapas o pasos de un caso de uso necesarios para alcanzar el objetivo previsto.

Caso de uso	Tipo Caso de Uso	Número Transacciones	Peso
Alta de Clientes	Medio	7	10
Modificar Datos Personales del Cliente	Medio	5	10
Baja de Alumno	Medio	7	10
Baja de Clientes no validados	Medio	5	10
Validar Cliente	Complejo	9	15
Consultar Datos Clientes	Medio	4	10
Consultar Apuntes	Simple	3	5
Buscar Clientes	Medio	4	10
Alta de Apuntes	Medio	7	10
Alta Clases Prácticas	Simple	3	5
Consultar Clases Prácticas	Simple	3	5
Baja Clases Prácticas	Simple	3	5
Asignar Profesor de Prácticas	Medio	5	10
Alta Examen	Complejo	9	15
Consultar Exámenes	Simple	3	5
Baja Examen	Simple	3	5
Alta Precios	Complejo	8	15
Consultar Precios	Simple	3	5
Alta Profesor	Medio	5	10
Consultar Datos de Profesores	Medio	4	10
Consultar Curriculums	Simple	2	5
Baja de Profesor	Medio	7	10
Buscar Profesor	Medio	4	10

Caso de uso	Tipo Caso de Uso	Número Transacciones	Peso
Modificar Datos Personales del Profesor	Medio	5	10
Generar Listado	Simple	3	5
Consultar Evolución Cliente	Simple	2	5
Cambio de Moneda	Simple	3	5
Configuración de Tests	Simple	3	5
Alta de Cliente Vía Web	Medio	7	10
Consulta de Condiciones Legales	Simple	3	5
Modificar Datos Personales de Clientes vía Web	Medio	5	10
Identificación del Cliente	Simple	3	5
Consultar apuntes vía web	Simple	2	5
Bajas de clientes vía web	Medio	7	10
Consultar estadísticas	Medio	4	10
Realizar Tests	Complejo	9	15
Envío de Currículum	Medio	5	10
Consulta de precios vía web	Simple	1	5
Enviar preferencias de clases prácticas vía web	Medio	4	10

Tabla 6. Identificación de los Casos de Uso y cálculo del Peso de los casos de uso

Aplicando la siguiente formula:

$$UUCW = \sum (Peso_i * TipoCasoDeUso_i)$$

Tenemos que el factor de peso de los casos de uso sin ajustar es:

$$UUCW = (16*5) + (19*10) + (4*15)$$

$$UUCW = 80 + 190 + 60$$

$$UAW = 330$$

4.2.3. Cálculo de los Puntos de Casos de Uso sin ajustar

Tomando como entrada los valores UAW = 13 y UUCW = 245; aplicando la siguiente fórmula:

$$UUCP = UAW + UUCW$$

Se tiene que el valor de los UCCP es:

$$UUCP = 13 + 330$$

$$UUCP = 343$$

4.2.4. Cálculo del Factor Complejidad Técnico

	Factor	F _i	Peso	F _i * Peso	Comentarios
T1	Sistema distribuido	2	0	0	Es un sistema centralizado
T2	Tiempos de respuesta críticos	1	1	1	El tiempo de respuesta es limitado, por las entradas provistas por el usuario
T3	Eficiencia del usuario	1	1	1	Escasas restricciones de eficiencia
T4	Procesos internos complejos	1	3	3	Existe algunos cálculos complejos internos
T5	Código reutilizable	1	0	0	No se requiere que el código sea reutilizable
T6	Fácil de instalar	0.5	4	2	El sistema es bastante fácil de instalar.
T7	Fácil de utilizar	0.5	5	2.5	El sistema es de fácil uso
T8	Portable	2	5	10	Es portable, y puede correr en diferentes plataformas.
T9	Fácil de modificar	1	1	1	Es fácil de mantener
T10	Concurrencia	1	5	5	El sistema permite que varios usuarios accedan simultáneamente.
T11	Características de seguridad	1	3	3	Seguridad normal
T12	Acceso a software de otras compañías	1	1	1	No tiene acceso permitido a terceros
T13	Incluye facilidades de aprendizaje para el usuario	1	1	1	El sistema es fácil de usar.

Tabla 7. Peso de los factores en el cálculo del TFC

Aplicando la siguiente formula:

$$TCF = 0.6 + 0.01 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

Tenemos que el factor de complejidad técnico es:

$$TCF = 0.6 + (0.01 \times 30,5)$$

$$TCF = 0.905$$

4.2.5. Cálculo del Factor de Ambiente (EF)

Trata de medir cómo de familiarizado está el equipo de desarrollo con el tipo de problema de proyecto a realizar.

	Factor	F _i	Peso	F _i * Peso
E1	Familiarizado con RUP	1.5	3	4.5
E2	Experiencia en este tipo de aplicaciones	0.5	3	1.5
E3	Experiencia en Orientación a Objeto	1	3	3
E4	Capacidad de liderazgo del analista	0.5	5	2.5
E5	Motivación	1	5	5
E6	Requisitos estables	2	3	6
E7	Trabajadores a tiempo parcial	-1	0	0
E8	Lenguaje de programación difícil de utilizar	-1	3	-3

Tabla 8. Peso de los factores en el cálculo del EF.

Aplicando la siguiente formula:

$$EF = 1.4 - 0.03 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

Tenemos que el factor de ambiente es:

$$EF = 1.4 - (0.03 * 19.5)$$

$$\underline{\underline{EF = 0.985}}$$

4.2.6. Cálculo de los Puntos de Casos de Uso (UCP)

Finalmente los puntos de casos de uso se calculan aplicando la siguiente fórmula, teniendo como entrada los valores calculados anteriormente.

$$UCP = UUCP * TCF * EF$$

$$UCP = 343 * 0.905 * 0.985$$

$$\underline{\underline{UCP = 305.76}}$$

4.2.7. Cálculo del Esfuerzo

Por último, si el número de factores en la puntuación del EF que no llegan al nivel promedio de 3, es decir, es menor o igual que 3, como es el caso del ejemplo, el número de horas-hombre necesarias se calcula:

$$\text{HorasHombre} = 20 * \text{UCP}$$

$$\text{UCP} = 20 * 305,76$$

$$\text{UCP} = 6115 \text{ horas}$$

Teniendo en cuenta que dos desarrolladores trabajarán 10 horas al día y 20 días al mes, la duración del proyecto se estima en:

$$\text{Tiempo} = \frac{6115 \text{ HorasHombre}}{2 \text{ Hombres}} \times \frac{1 \text{ Día}}{10 \text{ Horas}} \times \frac{1 \text{ Mes}}{20 \text{ Días}}$$

$$\text{Tiempo} = 15 \text{ meses}$$

Por tanto el tiempo estimado obtenido por el método basado en Puntos de Casos de Uso, es de 1 año y tres meses.

Se puede tener en cuenta las siguientes suposiciones:

Es importante destacar que este tiempo no ha sido dedicado totalmente al diseño del sistema sino que gran parte del mismo se ha empleado en la toma de contacto con las nuevas herramientas y tecnologías en las primeras fases del proyecto.

Además hay que tener en cuenta que el tiempo de desarrollo se ha incrementando en gran medida por el hecho de que durante la mayor parte del tiempo se han realizado tareas conjuntas con el fin de que ambos desarrolladores adquiriesen conocimientos sobre todas las materias tratadas.

Para una empresa real el tiempo empleado en este proyecto, entendiéndolo como su primer trabajo utilizando un tipo de herramientas y tecnologías específicas, se vería totalmente compensado con la experiencia adquirida por los desarrolladores, permitiéndoles llevar a cabo en el futuro proyectos relacionados en un menor tiempo.

5. CASOS DE USO Y PUNTOS DE FUNCIÓN

Existe una relación natural entre los Puntos de Función y los Casos de Uso. Los Puntos de Función (FPA) permiten estimar el tamaño del software a partir de sus requisitos, mientras que los Casos de Uso permiten documentar los requisitos del software. Ambos tratan de ser independientes de las tecnologías utilizadas para la implementación.

Algunas **desventajas** encontradas por la Consultoría de Sistemas FATTO [FATTO] frente a los Puntos de Función son las siguientes:

1. UCP solo puede ser aplicado en proyectos de software cuya especificación esté documentada en casos de uso. La medición del FPA es independiente a la forma en que los requisitos fueran escritos.
2. No existe un padrón único para describir los casos de uso. Existen diferentes estilos de escribir los casos de uso o en su granularidad (nivel de detalle) puede llevar a resultados diferentes en la medición por UCP. La medición por FPA de los casos de uso de un sistema siempre llegará al mismo resultado independientemente del estilo en que estén escritos los casos de uso u de su granularidad, debido a que FPA usa el concepto de Proceso Elemental.
3. Debido a que el método UCP está basado en casos de uso, el mismo no puede ser utilizado antes de concluir la fase de análisis de requisitos del proyecto. La mayoría de las veces hay una necesidad de obtener una estimativa antes de que esta etapa finalice. El proceso de medición del FPA también puede ser empleado después del levantamiento de requisitos del proyecto. Aunque existen técnicas estimativas del tamaño en puntos de función que pueden ser aplicadas con éxito antes de que la fase de análisis de requisitos concluya.
4. El método UCP no contempla la medición de proyectos de mejoría (mantenimiento) de software, solamente proyectos de desarrollo. FPA contempla la medición de proyectos de desarrollo, de mejora y aplicaciones.
5. No es posible aplicar UCP en la medición de aplicaciones existentes cuya documentación no esté actualizada o ni siquiera existe. Con FPA es posible realizar la medición analizando la propia aplicación en uso.
6. La determinación de los factores de complejidad técnicos y ambientales del UCP está sujeta a un grado de subjetividad que dificulta la consistencia de la aplicación del método en diferentes organizaciones. El factor de ajuste del FPA también posee el mismo problema, aunque IFPUG¹ posee directrices específicas que ayudan a minimizar este impacto.

¹ **IFPUG**: International Function Point Users Group

Según [CLEM06], [MOTO05], se puede resaltar algunas ventajas principales del uso del método UCP:

- El proceso del cálculo de los UCP's se puede automatizar, utilizando herramientas de gestión automática de casos de uso.
- El método es versátil y extensible para una variedad de proyectos de desarrollo.
- Es fácil de aprender y rápido de aplicar.

6. CONCLUSIÓN

En el presente trabajo, se ha presentado el método de estimación Puntos de Casos de Uso, método que permite estimar el esfuerzo de un proyecto de desarrollo de software a partir de los casos de uso. En este contexto, se han presentado y descrito cada uno de los pasos necesarios para calcular dicha estimación.

1. Puntos de Casos de Uso sin Ajustar
 - a. Identificar los actores que interactúan en el sistema y calcular el factor de peso de los actores sin ajustar.
 - b. Identificar los casos de usos y calcular el factor de peso de los casos de usos
 - c. Calcular los Puntos de Casos de Uso sin ajustar
2. Puntos de Casos de Uso Ajustado
 - a. Calcular el factor de complejidad técnico
 - b. Calcular el factor de ambiente
 - c. Calcular los puntos de Casos de Usos Ajustados
3. Estimación del Esfuerzo.

La estimación por Puntos de Caso de Uso resulta muy efectiva para estimar el esfuerzo requerido en el desarrollo de los primeros Casos de Uso de un sistema, si se sigue una aproximación iterativa como el Proceso Unificado de Racional. También se pudo observar que es fácil de entender, y a su vez, fácil de aplicar, siendo recomendable disponer tanto del personal con experiencia en el análisis de proyectos, como en el entorno de trabajo.

Cabe destacar que la estimación obtenida mediante este método sólo aplica al esfuerzo requerido para la fase de codificación del proyecto, siendo necesario aplicar otros ajustes con el fin de obtener el esfuerzo de todo el ciclo de vida del proyecto.

7. REFERENCIAS

- [CLEM06] Clemmons, Roy K. (2006) *Project Estimation with Use Case Points*. Disponible en: <http://www.stsc.hill.af.mil/crosstalk/2006/02/0602Clemmons.pdf> [Consulta: Diciembre 2009]
- [CUAD08] Cuadrado Gallego, Juan, Domínguez Alda, María y otros. (2008). *Estudio Experimental de la conversión entre las unidades de medición funcional del software puntos de casos de uso e IFPUG*. Revista Española de Innovación, Calidad e Ingeniería del Software, Vol.4, No. 2, 2008. Disponible en: <http://www.ati.es/IMG/pdf/CuadradoVol4Num2.pdf> [Consulta: Diciembre 2009]
- [DOSS09] Dos Santos Chiossi, Thelma Cecília (2009). *Guía para el cálculo de TFator&EFator*. Universidad de Campinas, Brasil. Disponible en: <http://www.ic.unicamp.br/~thelma/gradu/MC876/Turma2009/Trabalho-Plano-desenvolvimento/Tfator-Efator-guia-para-calculo.pdf> [Consulta: Diciembre 2009]
- [DSIC] *Rational Unified Process (RUP)*. Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. Disponible en: <https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/Introducci%C3%B3n%20a%20RUP.doc> [Consulta: Diciembre 2009]
- [FATTO] Consultoria e Sistemas. Especializada em Análise de Pontos de Função. *Vantagens da APF sobre os Pontos por Caso de Uso*. Disponible en: <http://www.fattocs.com.br/faq.asp#P27> [Consulta: Diciembre 2009]
- [GAGA09] García, María Carmen y Garzás, Javier (2009). *Método de Estimación de Puntos de Caso de Uso*. Disponible en: <http://www.kybeleconsulting.com/index.php/articulos/86> [Consulta: Diciembre 2009]
- [GOAL01] González, Silvia Alonso Carrera, Eva (2001). *Gestión de Auto Escuela y Realización de Test en Internet*. Disponible en: <http://pisuerga.inf.ubu.es/lsi/Docencia/TFC/ITIG/Curso2000-01/GestiondeAutoEscuelayRealizaciodetestenInternet/Documentacion/Anexo2-EspecificaciondeRequisitos.doc> [Consulta: Diciembre 2009]
- [MOTO05] (2005) *METHODS & TOOLS. Global knowledge source for software development professionals*. Disponible en Internet: <http://www.methodsandtools.com/PDF/mt200503.pdf> [Consulta: Diciembre 2009]

- [PERA09] Peralta, Mario. *Estimación del Esfuerzo basada en casos de uso*. Disponible en: <http://pisuerga.inf.ubu.es/rcobos/anis/estimacion-del-esfuerzo-basada-en-casos-de-usos.pdf> [Consulta: Diciembre 2009]
- [RUP02] *RUP, en portugués*. Disponible en: <http://www.wthreex.com/rup/portugues/index.htm> [Consulta: Diciembre 2009]