Estimación del Módulo de Seguimiento

1. Puntos de función

Este método mide el tamaño del software de acuerdo a las funcionalidades entregadas al cliente independientemente de la tecnología utilizada para la construcción de la aplicación, es útil en cualquiera de las fases de vida del software, desde el diseño inicial hasta la implementación y mantenimiento. Consiste en asignar una cantidad de puntos al desarrollo de una aplicación según la complejidad de los datos que maneja y de los procesos que realiza sobre ellos. La propuesta original identifica 5 tipos de funciones básicas: inputs, outputs, consultas, archivo lógico interno, e interfaces. La manera en que se operan estas funciones se define en los siguientes puntos.

Para este método se tendrá en consideración que 1 PF (Punto de Función) tiene un valor asignado de 2 Horas/Hombre. Esto es debido a que en relación con proyectos de software realizados en semestres anteriores, hemos decidido que 2 horas es un tiempo promedio en el que tardamos para implementar una funcionalidad de dificultad media/baja.

* 1. Funciones transaccionales
     1. Inputs: Entrada externa

Proceso para mantener uno o más archivos lógicos internos. Cuenta las pantallas o formularios usados para captura.

* Formato evaluación informe técnico – financiero (SEG10)
* Formulario para Registrar informe de visita técnica (SEG11)
* Formulario para el registro y seguimiento de beneficiarios de Becas de Doctorado Nacional (SEG12)
* Formulario de seguimiento a Becarios seleccionados por Colciencias (Becas Doctorado en el Exterior) (SEG12)
* Formulario de registro y Seguimiento a jóvenes investigadores (SEG12)
* Ingresar comentarios para validar requisitos mínimos de Becarios (SEG13)
* Adjuntar informe para condonación (SEG14)
* Adjuntar informe de evaluación de condonación (SEG16)

**Total =** 8

* + 1. Outputs: Salida externa

Proceso para presentar información al usuario que requiere operaciones adicionales al de solo recuperar datos. Cuenta pantallas o reportes que la aplicación produce.

* Propuestas que se encuentran en proceso de seguimiento y que han sido asignadas a un evaluador (SEG10)
* Registros de Becarios que están relacionados a un Usuario Colciencias y a una Convocatoria determinada (SEG12)

**Total =** 2

* + 1. Consultas

Procesos para presentar información leída de uno o más grupos de datos. No requieren procesamiento adicional.

* Listar propuestas en procesos de seguimiento y que han sido asignadas al evaluador (SEG10)
* Actualizar propuestas en proceso de seguimiento y asignadas a un evaluador (SEG10)
* Consultar informe de seguimiento asociado a una propuesta (SEG10)
* Listar los contratos asignados a un Usuario Colciencias (SEG11)
* Listar informes de visitas realizadas y que se encuentren finalizadas (SEG11)
* Listar convocatorias de formación vigentes (SEG12)
* Listar registros de becarios relacionados al usuario y a la convocatoria (SEG12)
* Listar las convocatorias de formación vigentes (SEG13)
* Listar registros de becarios por convocatoria (SEG13)
* Listar convocatorias relacionadas con un becario (SEG13)
* Listar convocatorias de formación en las que participa un becario (SEG14)
* Listar los informes ingresados asociados a una convocatoria de un becario (SEG14)
* Listar convocatorias de formación (SEG15)
* Listar becarios de convocatoria que han ingresado el informe final (SEG15)
* Listar por evaluador los informes que le han sido asignados (SEG16)

**Total =** 15

* 1. funciones de datos
     1. Archivo lógico interno

Numero de almacenamiento de datos mantenido a través de alguna transacción.

* Convocatoria
* Proyecto
* Notificación
* Anexos
* VisitaTecnica
* Evaluador
* Convenio a operador del contrato
* Becario
* Seguimiento Academico-Financiero
* Informe
* Evaluación

**Total** = 11

* + 1. Interfaces

Archivo de interfaz externa. Grupos de datos relacionados y referenciados que son mantenidos por otro sistema.

* Convenio-Contrato
* Proyecto(Formulario para cada postulante)
* Proyecto de Inversión

**Total** = 3

* 1. Factor de complejidad

Para el factor de complejidad se tienen en cuenta 14 aspectos presentados en la tabla a continuación. Cada uno de ellos está relacionado con un nivel de influencia encontrado en una escala entre 0 y 5.

|  |  |
| --- | --- |
| **Valor del factor** | **Influencia en el sistema** |
| 0 | Ninguna |
| 1 | Insignificante |
| 2 | Moderada |
| 3 | Media |
| 4 | Significativa |
| 5 | Fuerte |

|  |  |
| --- | --- |
| **Aspectos** | |
| Comunicaciones | 5 |
| Objetivos de desempeño (sobrecarga) | 3 |
| Tasa de transacciones | 4 |
| Eficiencia para usuario | 4 |
| Proceso complejo | 3 |
| Facilidad de instalación | 0 |
| Varios sitios | 0 |
| Funciones distribuidas | 0 |
| Configuración | 1 |
| Entrada de datos online | 3 |
| Actualización en línea | 3 |
| Reuso | 1 |
| Facilidad de operación | 1 |
| Facilidad de mantención | 2 |
| Total | **30** |

El factor de complejidad está dado por:

Por lo que tenemos:

* 1. Puntos de función sin ajustar

Los resultados dados para los 5 tipos de funciones se expresan en la siguiente tabla, donde a cada tipo de función básica se le define un peso en una escala de 0 a 10 basados en datos históricos de los integrantes del equipo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Numero** | **Tipo** | **Peso** | **Total** | |
| 8 | Inputs (EI) | 5 | 40 | |
| 2 | Outputs(EO) | 3 | 6 | |
| 15 | Consultas(EQ) | 5 | 75 | |
| 11 | Archivos(ILF) | 7 | 77 | |
| 3 | Interfaz(EIF) | 4 | 12 | |
| PFSA | | | | **210** |

Se tienen entonces 210 PFSA (Puntos de Función Sin Ajustar).

* 1. Puntos de función ajustados

Una vez obtenidos los puntos de función sin ajustar y el factor de complejidad podemos hallar los puntos de función ajustados aplicando la siguiente formula:

Donde reemplazando los valores anteriores tenemos que para este proyecto se requieren 199,5 PF (Puntos de Función):

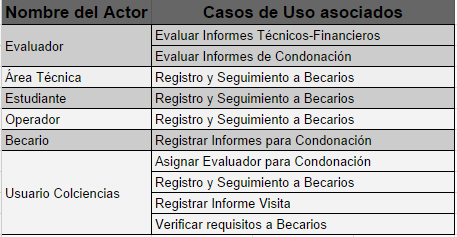
Asumiendo un esfuerzo temporal de 2 horas por cada punto de función, se necesita un total de **399 horas** para culminar el producto.

Estimando una intensidad de 6 horas diarias de trabajo de lunes a viernes (30 horas semanales) se requerirían 13 semanas y 2 horas de tiempo para finalizar el aplicativo.

Si cada hora representa un costo de $6.200, el sistema integrado de información tendría un costo de $2.473.800.

1. Puntos de casos de uso

Este método de estimación de proyectos de software fue desarrollado en 1993 por Gustav Karner de Rational Software y está basado en una metodología orientada a objetos, dándole el nombre de "estimación de esfuerzos con casos de uso". Surgió como una mejora al método de puntos de [función](http://www.monografias.com/trabajos7/mafu/mafu.shtml) pero basando las estimaciones en el [modelo](http://www.monografias.com/trabajos/adolmodin/adolmodin.shtml) de casos de uso, [producto](http://www.monografias.com/trabajos12/elproduc/elproduc.shtml) del [análisis](http://www.monografias.com/trabajos11/metods/metods.shtml" \l "ANALIT) de requerimientos. Según su autor, la funcionalidad vista por el usuario (modelo de casos de uso) es la base para estimar el tamaño del software.[1]



* 1. Cálculo de puntos de casos de uso sin ajustar (uucp)

Para la estimación el primer paso que se lleva a cabo es el cálculo de los Puntos de Casos de Uso sin ajustar. Este valor se calcula a partir de la siguiente ecuación:

, 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.

* + 1. Determinación del factor de peso de los actores sin ajustar (UAW).

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el módulo de Seguimiento y la complejidad de cada uno de ellos. La complejidad de los actores se establece, teniendo en cuenta en primer lugar, si se trata de una persona o de otro sistema, y en segundo lugar, la forma en que el actor interactúa con el sistema.

**Factores de Peso de los Actores**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tipo de actor | Descripción | Factor  de peso | Número de actores | | *Resultado* |
| Simple | Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación(API, Aplication Programming Interface) | 1 | 0 | | *0* |
| Promedio | Otro sistema que interactúa con el sistema a desarrollar mediante un [protocolo](http://www.monografias.com/trabajos12/mncerem/mncerem.shtml) o una interfaz basada en [texto](http://www.monografias.com/trabajos13/libapren/libapren.shtml). | 2 | 0 | | *0* |
| Complejo | Una persona que interactúa con el sistema mediante una interfaz gráfica. | 3 | 6 | | *18* |
|  | | | | *Total* | *18* |

**UAW = 18**

* + 1. Determinación del factor de peso en los casos de uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de 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 la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómicas.

**Factores de peso de los casos de uso**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tipo de caso de uso | Descripción | Factor  de peso | | Número de Casos de Uso | | *Resultado* | |
| Simple | 1-3 Transacciones | 5 | | 0 | | *0* | |
| Promedio | 4-7 Transacciones | 10 | | 4 | | *40* | |
| Complejo | Mayor de 8 Transacciones. | 15 | | 3 | | *45* | |
|  | | | *Total* | | *85* | |  |

**UUCW = 85**

Calculando:

**UUCP = 18 + 85**

**UUCP = 103**

* 1. Cálculo de puntos de casos de uso ajustado

Seguidamente de calcular los puntos de casos de uso sin ajustar, se debe ajustar este valor mediante la siguiente ecuación:

, 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

* + 1. Determinación del factor de complejidad técnica (TCF)

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los valores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. Los pesos están dados en una escala de 0 a 10,

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Número de factor | Descripción | Peso | Valor | Factor | *Comentario* |
| T1 | Sistema Distribuido | 2 | 0 | 0 | *El sistema es Web, pero no se tendrán en cuenta funciones distribuídas* |
| T2 | Tiempo de respuesta | 2 | 2 | 4 | *El tiempo de respuesta respalda los* [objetivos](http://www.monografias.com/trabajos16/objetivos-educacion/objetivos-educacion.shtml) *que se persiguen con el proyecto realizado, por lo que es el adecuado.* |
| T3 | Eficiencia por el usuario | 2 | 4 | 8 | *Algunos roles necesitan estar relacionados con el sistema para su mejor funcionamiento.* |
| T4 | Proceso interno complejo | 1 | 3 | 3 | *El sistema no posee cálculos complejos, aunque proporciona una serie de datos lógicos que necesitan un nivel medio de* [conocimiento](http://www.monografias.com/trabajos/epistemologia2/epistemologia2.shtml) *para lograr su correcta comprensión.* |
| T5 | Reusabilidad | 1 | 1 | 1 | *No es objetivo esencial hacer reusabilidad del* [código](http://www.monografias.com/trabajos12/eticaplic/eticaplic.shtml)*, a pesar de que este será orientado a objetos y podrá ser usado por* [sistemas](http://www.monografias.com/trabajos11/teosis/teosis.shtml) *similares.* |
| T6 | Facilidad de instalación | 0 | 0 | 0 | *Por ser un sistema Web no existe complejidad en la instalación.* |
| T7 | Facilidad de uso | 2 | 5 | 10 | *El sistema debe ser fácil de usar, aunque se encuentra dirigido a personas ajenas al centro además.* |
| T8 | Portabilidad | 0 | 0 | 0 | *El sistema no tiene como requisito ser diseñado para que sea usado en situaciones similares en otras* [empresas](http://www.monografias.com/trabajos11/empre/empre.shtml)*.* |
| T9 | Facilidad de cambio | 2 | 5 | 10 | *El sistema encuentra estructurada para que los cambios realizados afecten lo menos posible las funcionalidades del sistema.* |
| T10 | Concurrencia | 2 | 4 | 8 | *La concurrencia es tratada con suma importancia.* |
| T11 | Objetivos especiales de seguridad | 2 | 4 | 8 | *La* [seguridad](http://www.monografias.com/trabajos/seguinfo/seguinfo.shtml) *del sistema es un tema bastante controlado, ya que el sistema sólo permite que un usuario realice las funcionalidades correspondientes a su rol dentro del sitio.* |
| T12 | Acceso directo a terceras partes | 0 | 0 | 0 | *La aplicación solo es accesible a quien tenga un rol de usuario asignado.* |
| T13 | Facilidades especiales de [entrenamiento](http://www.monografias.com/trabajos14/mocom/mocom.shtml) a usuarios finales | 1 | 1 | 1 | *No se hace necesario el entrenamiento de los usuarios finales, debido a la facilidad de uso que presenta el sistema, pero se debe incluir un* [manual](http://www.monografias.com/trabajos13/mapro/mapro.shtml) *de usuario para garantizar la correcta usabilidad de dicho sistema.* |
|  |  |  | *Total Factor* | *53* |  |

El factor de complejidad técnica se calcula mediante la siguiente ecuación:

TCF = 0.6 + 0.01\*53

**TCF = 1.13**

* + 1. Determinación del factor ambiente (EF)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente.

**Factores de ambiente.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Número del factor | Descripción | Peso | Valor | Factor | *Comentario* | |
| E1 | Familiaridad con el modelo del proyecto usado. | 1.5 | 3 | 4.5 | *Se está familiarizado con el modelo del proyecto, pero la experiencia en el modelado es media.* | |
| E2 | Experiencia en la aplicación | 0.5 | 4 | 2 | *Se necesita de un equipo capacitado y de conocimientos suficientes para garantizar su correcto funcionamiento.* | |
| E3 | Experiencia OO. | 1 | 4 | 4 | *Se considera cierto grado de experiencia en la* [programación orientada a objetos](http://www.monografias.com/trabajos/objetos/objetos.shtml) *(OO), debido a que esta es la que se ha estudiado y trabajado.* | |
| E4 | Capacidad del analista [líder](http://www.monografias.com/trabajos15/liderazgo/liderazgo.shtml). | 0.5 | 3 | 1.5 | *No existe analista líder, los analistas que integran el equipo de trabajo poseen capacidad media.* | |
| E5 | Motivación. | 1 | 5 | 5 | *Alta* | |
| E6 | Estabilidad de los requerimientos. | 2 | 2 | 4 | *El sistema no se encuentra sujeto a cambios, el mismo brinda las funcionalidades esenciales que dan cumplimiento a los objetivos que iniciarán su realización.* | |
| E7 | Personal media jornada. | 0 | 0 | 0 | *Se trabajará a tiempo completo.* | |
| E8 | Dificultad en [lenguaje](http://www.monografias.com/trabajos35/concepto-de-lenguaje/concepto-de-lenguaje.shtml) de programación. | 2 | 2 | 4 | *El lenguaje* *empleado fue Ruby y este ofrece grandes facilidades y ventajas, contando con una curva de aprendizaje suave.* | |
|  | | *Total* | *25* |  | |  |

El factor de ambiente se calcula mediante la siguiente ecuación:

EF = 1.4 – 0.03\*25

**EF = 0.65**

Cálculo de los puntos de caso de uso ajustados:

UCP = UUCP \* TCF \* EF

UCP = 103 \* 1.13 \* 0.65

**UCP = 75.65**

* 1. Cálculo del esfuerzo

El esfuerzo en horas-hombre viene dado por:

, donde

E = esfuerzo estimado en horas/hombre

UCP = puntos de casos de uso ajustados

CF = factor de conversión

Asumiendo un factor de conversión de 5 horas por cada punto de casos de uso ajustados, se necesita un total de **321 horas** para culminar el producto. Este valor es dado de esta manera debido a que un caso de uso es algo mucho más elaborado que una funcionalidad básica, por tanto la intensidad horario debe ser mayor al valor planteado en el método anterior.

E = 75.65 \* 5

**E = 378 horas/hombre**

Estimando una intensidad de 6 horas diarias de trabajo de lunes a viernes (30 horas semanales) se requerirían 12 semanas y 18 horas de tiempo para finalizar el aplicativo.

Si cada hora representa un costo de $6.200, el sistema integrado de información tendría un costo de $2.343.600.

Se debe tener en cuenta que éste método proporciona una estimación del esfuerzo en horas/hombre contemplando solo el desarrollo de la funcionalidad especificada en los casos de uso.

1. T-shirt Sizing

T-shirt sizing es una técnica de estimación ágil que fue pensada dentro de las metodologías agiles para categorizar historias de usuario en tallas o medidas de camisetas. Cada una de estas medidas tiene asignado un valor en términos de “Story Points” basados en la experiencia del equipo de desarrollo.

La técnica consiste en que el equipo debe agrupar las historias de usuario en su correspondiente medida de camiseta de acuerdo a su complejidad, sin embargo, en este ejemplo se hará una adaptación para usar los casos de uso correspondientes al presente proyecto. También se hará una adaptación particular a este método con respecto a los Story Points de cada medida con el fin de asignar rangos y no valores fijos para así poder contemplar un aumento o reducción en el esfuerzo de desarrollo.

|  |  |
| --- | --- |
| **T-Shirt Size** | **Point Size** |
| S | 1-3 |
| M | 4-6 |
| L | 7-9 |
| XL | 10-12 |
| XXL | 13-15 |

* 1. Aplicación de la técnica

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **T-Shirt Size** | **Casos de uso** | **Cantidad** | **Point Size** | **Total** | |
| S |  | 0 | 1-3 | 0 | |
| M | * SEG15 - Asignar evaluador para condonación * SEG16 - Evaluar informes de condonación | 2 | 4-6 | 8-12 | |
| L | * SEG10 - Evaluar informes Técnicos-Financieros * SEG14 - Registrar informes para condonación | 2 | 7-9 | 14-18 | |
| XL | * SEG13 - Verificar requisitos a becarios | 1 | 10-12 | 10-12 | |
| XXL | * SEG11 - Registrar informe visita * SEG12 - Registro y seguimiento a becarios | 2 | 13-15 | 26-30 | |
| **Story Point Totales** | | | | | 58-72 |

Con base en la tabla anterior podemos encontrar que los SP (Story Point) totales para el módulo de seguimiento se encuentran entre un rango de 58 a 72, tomando **65 SP** como valor promedio.

Por último, se define que la relación entre Story Points y horas es:

Este valor ha sido asumido debido a que es el esfuerzo que como equipo de trabajo hemos identificado en proyectos anteriores y es necesario dedicar mínimamente en un día para alcanzar las metas planteadas en un tiempo prudente.

Así pues, el esfuerzo estimado en horas hombre es:

Estimando una intensidad de 6 horas diarias de trabajo de lunes a viernes (30 horas semanales) se requerirían 13 semanas de tiempo para finalizar el aplicativo.

Si cada hora representa un costo de $6.200, el sistema integrado de información tendría un costo de $2.418.000.

1. Pregunta

**¿Tendría sentido que la aplicación de tres técnicas de estimación diferentes sobre el proyecto final produzcan resultados diferentes, teniendo en cuenta que los requisitos y las restricciones del proyecto son las mismas para todas las técnicas?**

Es poco probable lograr resultados idénticos con la aplicación de los tres métodos; no obstante, deben tener una variación poco significativa ya que se están teniendo en cuenta los mismos requisitos y restricciones. Esto se puede comprobar con los resultados obtenidos en la presente estimación, que tienen los siguientes rangos:

**Costo:** [$2.343.600 – $2.473.800] Variación del 5,55% con respecto al valor inicial

**Cantidad Horas Hombre:** [378 - 399] Variación del 5,55% con respecto al valor inicial

1. Referencias

[1]. Valero Orea, Sergio .ESTIMACIÓN DE PROYECTOS DE SOFTWARE CON PUNTOS DE CASOS DE USO.