

Desarrollo de aplicación en Scrum con control de versiones y pruebas unitarias.

Práctica de la asignatura Calidad, Medición y Estimación de Procesos y Productos Software

Manuel de la Villa

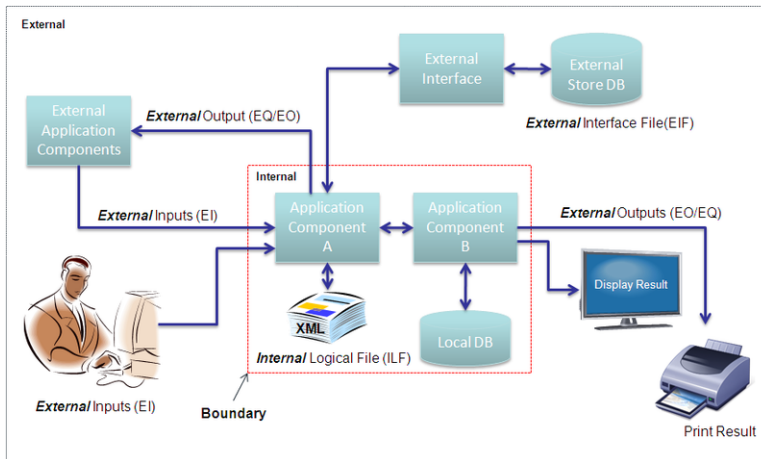
Depto. de Tecnologías de la
Información

manuel.villa@dti.uhu.es

Resumen: El objetivo de la presente práctica será el diseño y desarrollo de un proyecto siguiendo la metodología ágil SCRUM, usando un gestor de versiones (como Git) que dé soporte al desarrollo colaborativo e incluyendo la elaboración de pruebas unitarias usando la librería JUnit.

Palabras clave: SCRUM, gestión de versiones, Git, Github, JUnit, TDD.

1. Introducción



Visión: En el momento de definición de un proyecto, cuando aún los requisitos no están detallados ni la tecnología es conocida, se requiere una estimación inicial de tamaño y costes. Para estimar el tamaño de un proyecto una de las técnicas más usadas es el uso de la métrica Punto-Función.

La métrica Punto-Función se centra en el análisis descendente del desarrollo a abordar, analizando la complejidad de las diversas funcionalidades o transacciones que se han de realizar. Cada transacción, en función del número de ficheros y datos de estos ficheros involucrados, tendrá asociada una complejidad. Una vez calculada la complejidad de todas las transacciones y de los propios ficheros, hemos de realizar un ajuste que nos permita tener en cuenta esos requisitos no funcionales, ya sean de carácter técnico u organizacional, que pueden complicar aún más nuestro proyecto.

Una vez realizado el ajuste, obtenemos una medición de la funcionalidad en una unidad, el PF (Punto-Función) que, a través de la colección de valores recopilados de proyectos por la ISBSG, puede regresionarse y aportarnos unos valores iniciales de esfuerzo y duración.

La práctica va a consistir en elaborar una herramienta que dé soporte a todo el proceso de medición de la funcionalidad para obtener una estimación de tamaño y esfuerzo aplicando el método de Albrecht así como los cálculos aplicando las tablas de la ISBSG para calcular esfuerzo y duración.

Requisitos operacionales obligatorios:

- La práctica ha de realizarse de manera **obligatoria** por equipos de 2-3 personas.
- Deberá usarse una plataforma de **gestión de versiones** tipo Git para el desarrollo colaborativo del proyecto.
- Las clases involucradas en el cálculo directo del tamaño de la aplicación deberán incluir un conjunto de **pruebas unitarias** con la librería JUnit.
- Se valorará positivamente la formalización de la **integración continua** de software usando Jenkins.
- Se valorará el uso del patrón MVC y de formularios Swing.

Requisitos funcionales (han de implementarse todos):

- El usuario debe poder identificar los **ficheros lógicos** (tanto internos como externos) involucrados en el proyecto, así como los datos elementales que los componen.
- El usuario debe poder registrar cada transacción, identificando si se trata de una **entrada externa, una salida externa o una consulta externa**.
- En cada transacción, registrará cuantos ficheros y cuantos datos elementales participan, para poder así calcular la complejidad implícita.

El formulario 'Añadir elemento' presenta los siguientes campos y elementos:

- denominación:** Un campo de texto rectangular.
- tipo de transacción:** Un menú desplegable con la opción 'Entrada externa' seleccionada.
- número ficheros / tipo registros:** Un campo de texto rectangular.
- número de datos elementales:** Un campo de texto rectangular.
- Botones:** Dos botones rectangulares, 'Grabar' y 'Cancelar', situados a la derecha de los campos de número.

Ilustración 1. Prototipo formulario añadir elemento (<http://framebox.org/AabVd>)

- Habrá una sección del programa en la que se mostrará el **cuadro resumen** que calcula el total de Puntos Funcion no Ajustados:

DESCRIPCIÓN	SENCILLA	MEDIA	COMPLEJA	TOTAL P.F.
Nº de Entradas Externas	x 3	x 4	x 6	
Nº de Salidas Externas	x 4	x 5	x 7	
Nº Grupos Lógicos de Datos Internos	x 7	x 10	x 15	
Nº de Grupos Lógicos de Datos de Interfaz	x 5	x 7	x 10	
Nº de Consultas Externas	x 3	x 4	x 6	
TOTAL PUNTOS FUNCIÓN NO AJUSTADOS (PFNA)				

- El usuario deberá poder realizar el ajuste de PFNA atendiendo a las **características generales del sistema**. Consistirá en el cálculo de un *Factor de Ajuste* en base a la cuantificación de ciertos coeficientes vinculados con las características deseadas del sistema (comunicación de datos, rendimiento, facilidades de instalación, de operación, frecuencia de transacciones, etc.).
- A cada una de estas características el usuario le podrá asignar un *factor de peso* (un valor entre 0 y 5) que indica la importancia de la característica para el sistema bajo análisis. El significado del valor asignado a cada característica es el siguiente:

- 0 No presente o sin influencia
- 1 Influencia incidental
- 2 Influencia moderada
- 3 Influencia media
- 4 Influencia significativa
- 5 Fuerte influencia

- Las **características** a tener en cuenta son:
 - Comunicación de datos: Cuántas facilidades de comunicación hay disponibles para ayudar en el intercambio de información con la aplicación o el sistema?
 - Procesamiento distribuido de datos: Cómo se manejan los datos y las funciones de procesamiento distribuido
 - Rendimiento: Existen requerimientos de velocidad o tiempo de respuesta?
 - Configuraciones fuertemente utilizadas: Cómo de intensivas se utilizan las plataformas hardware donde se ejecuta el sistema
 - Frecuencia de transacciones: Con qué frecuencia se ejecutan las transacciones? Diariamente, semanalmente,...
 - Entrada de datos on- line: Qué porcentaje de la información se ingresa on-line?
 - Eficiencia del usuario final: Aplicación diseñada para maximizar la eficiencia del usuario final
 - Actualizaciones Online: Cuántos Archivos Lógicos Internos se actualizan por una transacción on-line?
 - Procesamiento complejo: Hay procesamientos lógicos o matemáticos intensivos en la aplicación?
 - Reusabilidad: La aplicación se desarrolla para suplir una o muchas de las necesidades de los usuarios?
 - Facilidad de instalación: Cómo de difícil es la instalación y la conversión al nuevo sistema?
 - Facilidad de operación: Cómo de efectivos y/o automatizados deben ser los procedimientos de arranque, parada, backup y restore
 - Instalación en distintos lugares: La aplicación fue concebida para su instalación en múltiples sitios y organizaciones?
 - Facilidad de cambio: La aplicación fue concebida para facilitar los cambios sobre la misma?

Ajuste PF con características generales del sistema	
Comunicación de datos	1
Funciones distribuidas	1
Prestaciones	1
Gran uso de la configuración	1
Velocidad de las transacciones	1
Entrada de datos en línea	1
Diseño para la eficiencia del usuario final	1
Actualización de datos en línea	1
Complejidad del proceso lógico interno	1
Reusabilidad del código	1
Facilidad de instalación	1
Facilidad de operación	1
Localizaciones múltiples	1
Facilidad de cambios	1
Total: <input type="text"/>	
<input type="button" value="Grabar"/> <input type="button" value="Cancelar"/>	

Ilustración 2. Prototipo formulario características generales del sistema

- El programa calculará el Factor de Ajuste como $FA = (SVA \times 0.01) + 0.65$, donde SVA es la suma de los valores de 0 a 5 de los 14 atributos o características generales anteriores.

- Finalmente, los Puntos de Función Ajustados se obtienen como el producto de los Puntos de Función sin ajustar por el Factor de Ajuste: **FP = FPNA x FA**
- Para obtener una mejor calificación, se pueden realizar las siguientes funcionalidades: Una vez calculado el tamaño en PF de la aplicación, el usuario podrá calcular **Esfuerzo y Duración aplicando las tablas ISBSG**. El usuario podrá identificar el **tipo de desarrollo** (nuevo o mantenimiento), la **plataforma** (Mainframe, Mid-range, PC o Combinación), el **lenguaje** a emplear (3GL, 4GL o generación de código automática) y a partir de ellas calcularse:

- **Esfuerzo**= $C \cdot PF^E$ medido en horas-hombre, donde

	Características	C	E
1	MF	49,02	0,736
2	MR	78,88	0,646
3	PC	48,90	0,661
4	Multi	16,01	0,865
5	3GL	54,65	0,717
6	4GL	29,50	0,758
7	GenAp	68,11	0,660
8	Mantenimiento	52,58	0,683
9	Nuevo	39,05	0,731
10	MF-3GL	65,37	0,705
11	MF-4GL	52,09	0,640
12	MF-GenAp	65,68	0,692
13	MR-3GL	126,3	0,565
14	MR-4GL	62,35	0,694
15	PC-3GL	60,46	0,648
16	PC-4GL	36,48	0,694
17	Multi-3GL	19,82	0,666
18	Multi-4GL	6,49	0,983
19	MF-3GL-Mantenimiento	83,27	0,650

- **Duración**= $C \cdot PF^E$ medido en meses (1 persona), donde

	Características	C	E
1	PC	0,503	0,409
2	Multi	0,679	0,341
3	4GL	0,578	0,393
4	Nuevo	0,739	0,359
5	PC-4GL	0,348	0,471
6	Multi-4GL	0,366	0,451
7	PC-4GL-Nuevo	0,250	0,515
8	Multi-4GL-Nuevo	0,240	0,518

La actividad deberá acompañarse de una memoria en la que se documenten las decisiones de diseño (requisitos, modelado estructural, etc...).

La práctica tiene como fecha límite de entrega el 7 de febrero de 2017.