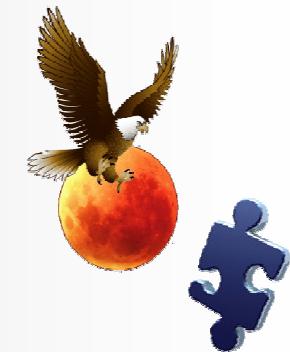




Universidad de Burgos
ESCUELA POLITÉCNICA SUPERIOR
Sistemas Informáticos
Ingeniería Informática

CLASIFICACIÓN DE REFACTORIZACIONES

DYNAMIC REFACTORING PLUGIN 3.0



TUTOR

RAÚL MARTICORENA SÁNCHEZ

ALUMNOS

MÍRYAM GÓMEZ SAN MARTÍN
ÍÑIGO MEDIAVILLA SAIZ

Burgos, Julio 2011

PROYECTOS ANTERIORES

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

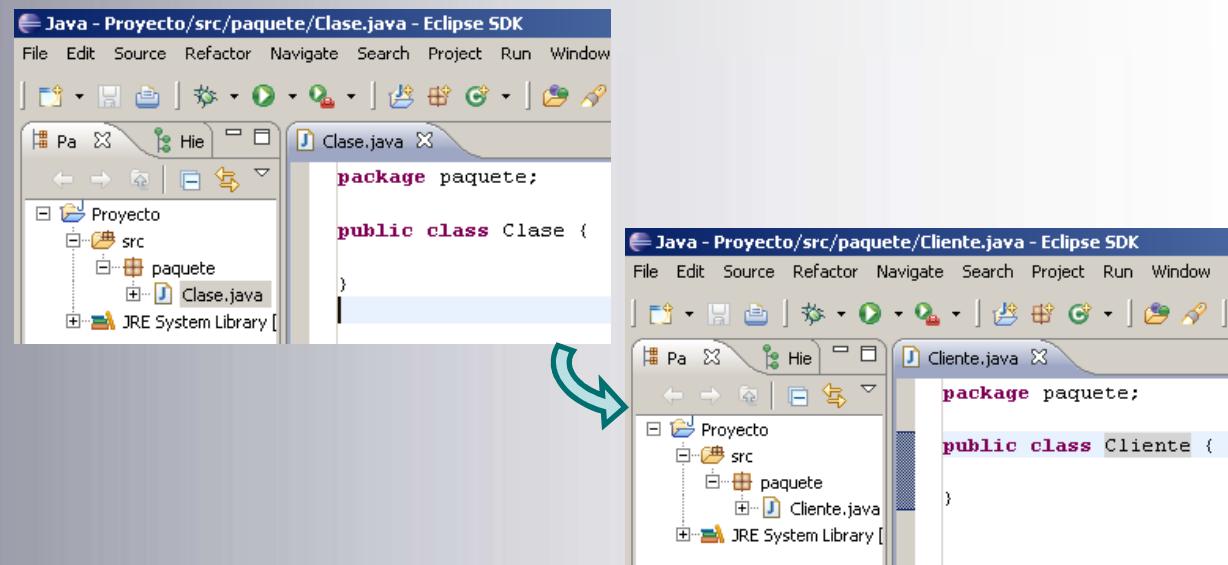
COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

Herramienta que permitía **definir refactorizaciones.**



CONCEPTOS TEÓRICOS

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

REFACTORIZACIÓN

Serie de pequeñas transformaciones, que **cambia la estructura interna** del programa **sin cambiar su comportamiento externo.**



PRECONDICIONES

ACCIONES

POSTCONDICIONES

OBJETIVOS DEL PROYECTO

OBJETIVOS FUNCIONALES



Facilitar
construcción



Clasificar
refactorizaciones



Visualizar
catálogo
refactorizaciones





Facilitar construcción

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

DynamicRefactoringWizard

Description - Input Configuration

Inputs

- moon.core.Model(1)
- moon.core.classdef.ClassDef(1)
- moon.core.classdef.ClassDef(2)
- javamoon.expr.constant.JavaS

Parameters

Name: Main

From:

Method:

JavaStringConstant

Description: String constant for

Refactorings it belongs: MoveField, MoveMethod

Refactorings where it's Main:

JavaDoc

Back Next Finish Cancel





Clasificar

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

Classifications Editor

Classifications

Available classifications.

Scope

Fowler

Bad Smells



Selected Classification

Selected classification data.

Name: Scope Multi:

Description:

{Description}

Categories

Existing categories.

Name ^

Attribute

Class



Dynamic Refactoring Wizard

Dynamic refactoring creation (Step 1 of 7)

Description

Name

Description

Image

 ...

Motivation

Keywords

Categories

- BadSmells
- Fowler
- Scope
- Attribute
- Class
- CodeFragment

Back

Next

Finish

Cancel





Visualizar

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS

RefactoringCatalogBrowser

Classification: RefactoringScope

Available refactorings:

- Class
- Attribute
 - Add attribute
 - Preconditions
 - MethodExists
 - Actions
 - Posconditions
 - Rename attribute

Show Filtered Clear all conditions

S^	Condition	Clear
1	category:Scope@Class	<input checked="" type="checkbox"/>
2	keyword:attribute	<input checked="" type="checkbox"/>

Overview Inputs Mechanism

Rename Method

Description: Permite añadir un nuevo parametro a la signatura de un metodo.

Motivation: Verifica que no exista ya otro parametro en el metodo con el mismo nombre y que no existan variables locales a ese metodo con nombre

Categories: Scope_Method

Keywords: Favoritos Rename



OBJETIVOS DEL PROYECTO

OBJETIVOS NO FUNCIONALES

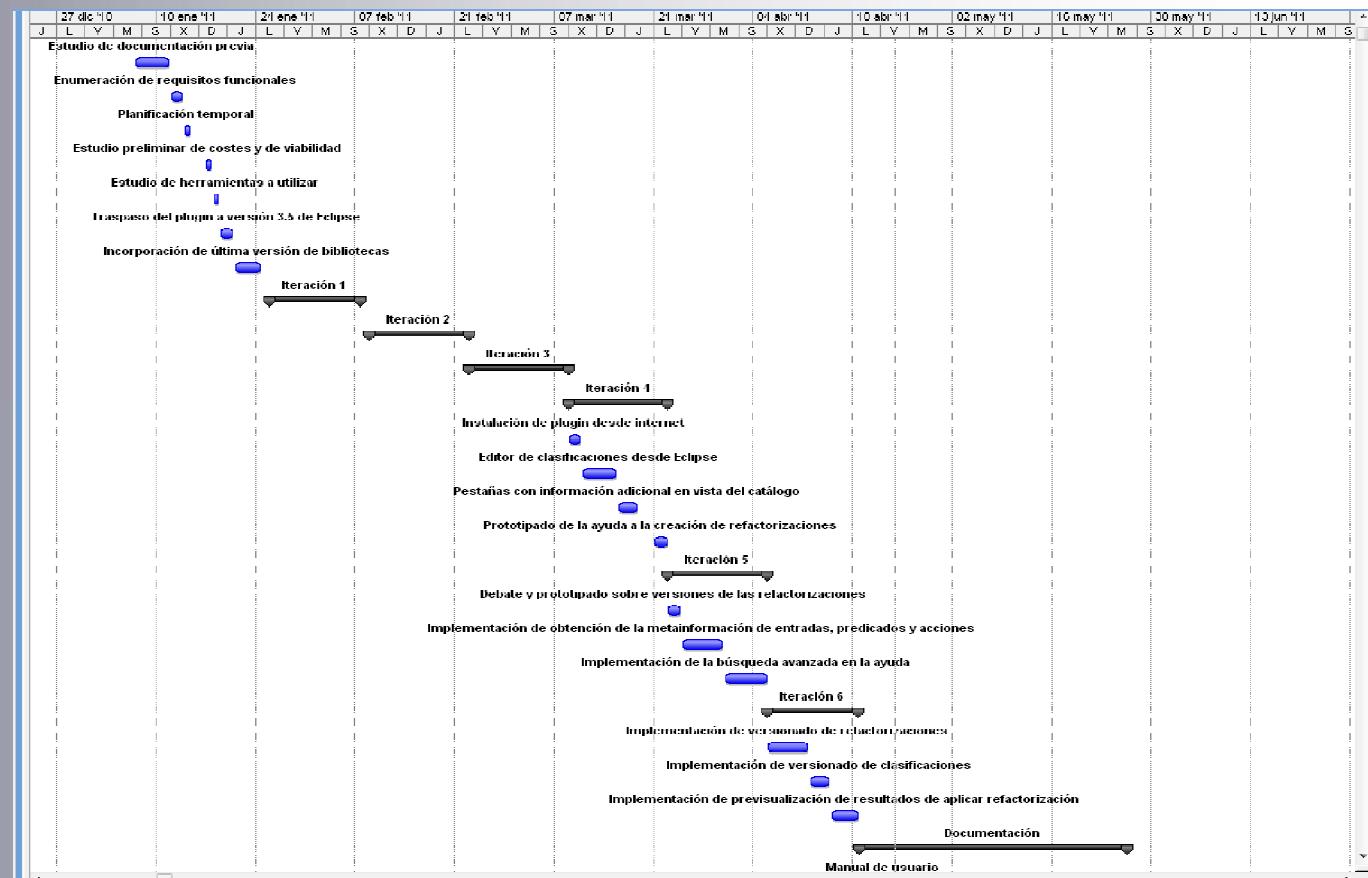


PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS



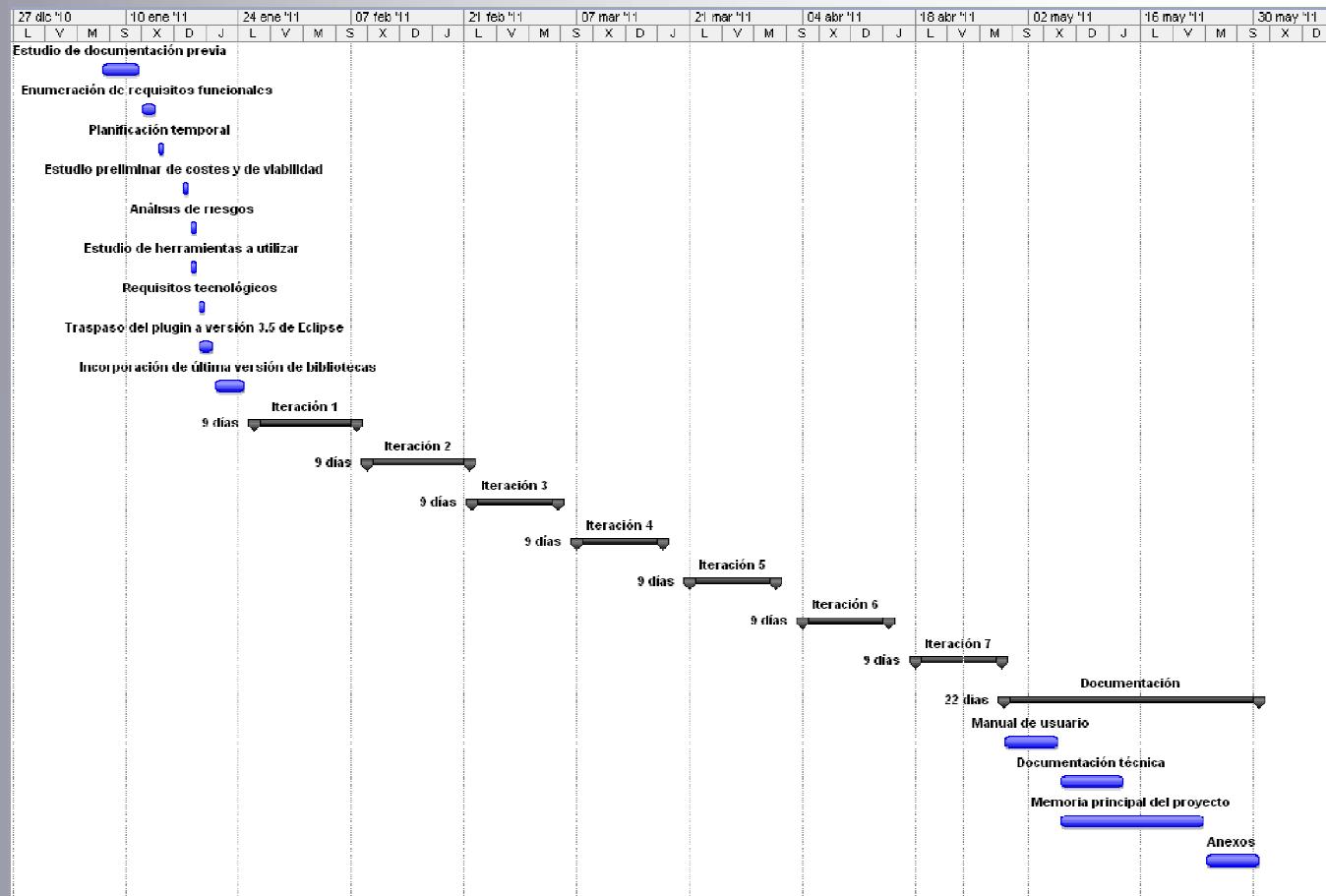
PLANIFICACIÓN TEMPORAL (1)

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS



PLANIFICACIÓN TEMPORAL (2)

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS



ESTUDIO VIABILIDAD

VIABILIDAD ECONÓMICA

Concepto	Cantidad
Coste personal	17.700,00 €
Coste Seguridad Social	5.292,30 €
Coste hardware	200,00 €
Coste software	201,33 €
Otros gastos	2.000,00 €
Coste Total	25.393,63 €



ESTUDIO VIABILIDAD

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

- Análisis Coste/Beneficio

Precio aplicación: 120 €/licencia

Para recuperar la inversión deberán venderse :
25.393,63€ de coste total / 120 € por licencia = **212 licencias**

Tabla comparativa entre el número de licencias que deberán venderse para amortizar el coste estimado.

Precio (€)	Nº licencias
100	≈ 254
120	≈ 212
140	≈ 182
180	≈ 141



ESTUDIO VIABILIDAD

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

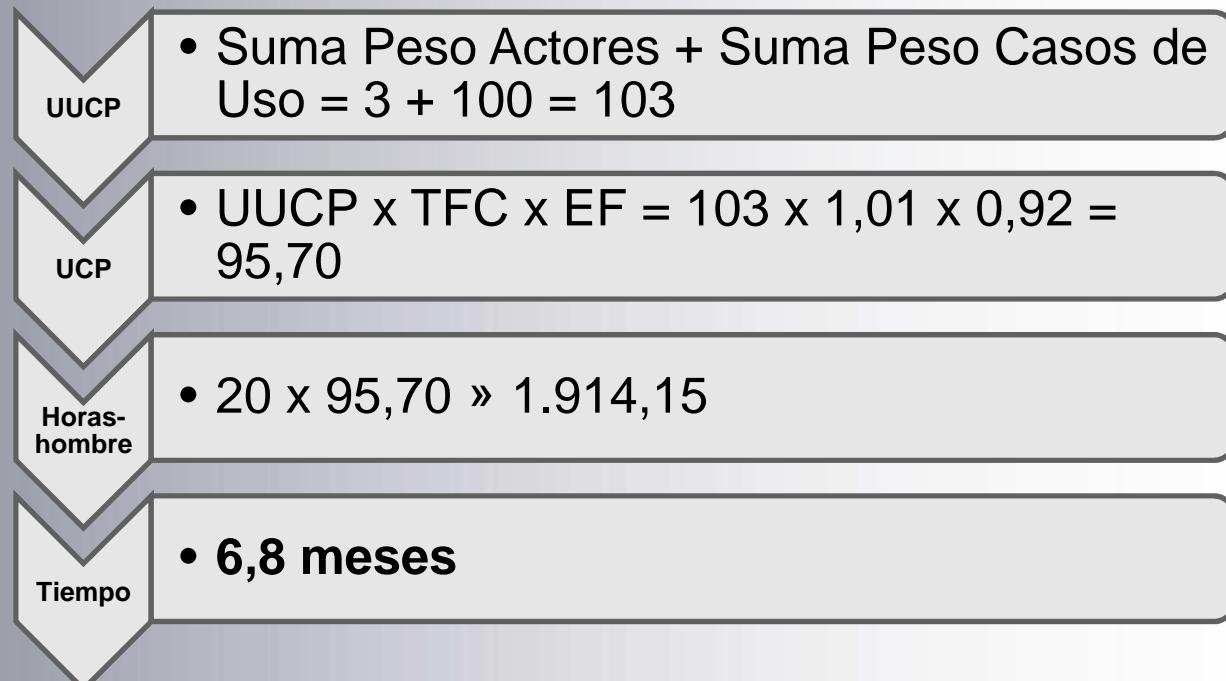
COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

VIABILIDAD TEMPORAL



ASPECTOS RELEVANTES

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

CAMBIOS EN LA CARGA DE CLASES

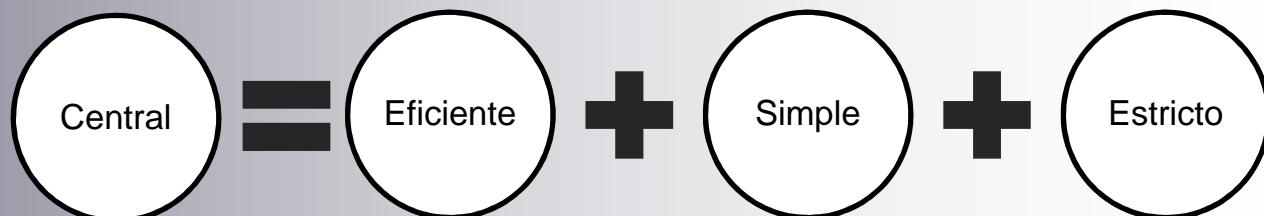
- Permitió refactorizaciones sobre cualquier clase.
- Descubrió un error que disparaba el consumo de memoria.
- Supuso la sustitución completa del sistema de deshacer.



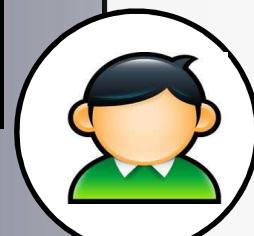
ASPECTOS RELEVANTES

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS

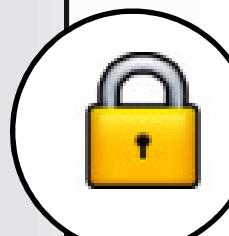
CATÁLOGOS ÚNICOS



VERSIONADO



- Editable
- Copias



- No Editable
- Actualizable

ASPECTOS RELEVANTES

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

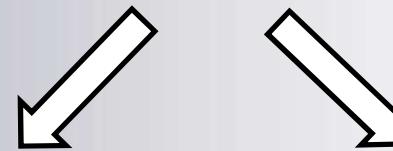
AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

BÚSQUEDA CON LUCENE

+ 280 entradas, 215 predicados

Necesidad : búsqueda **flexible**



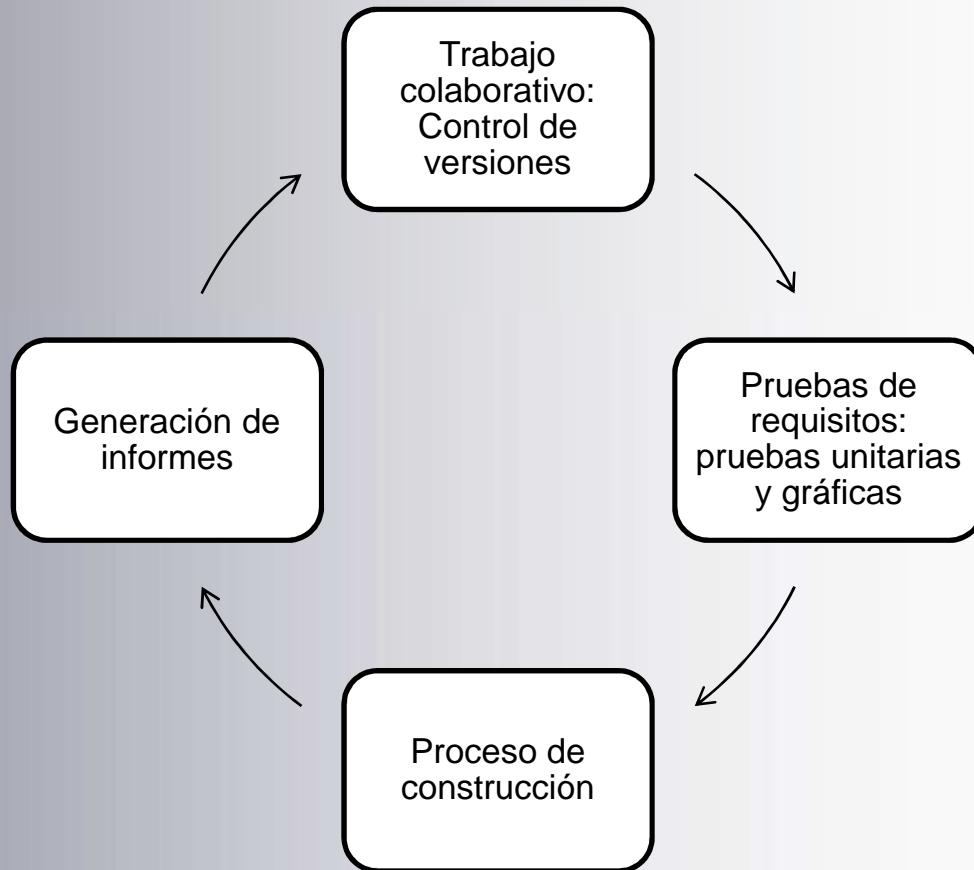
Extracción
javadoc



ASPECTOS RELEVANTES

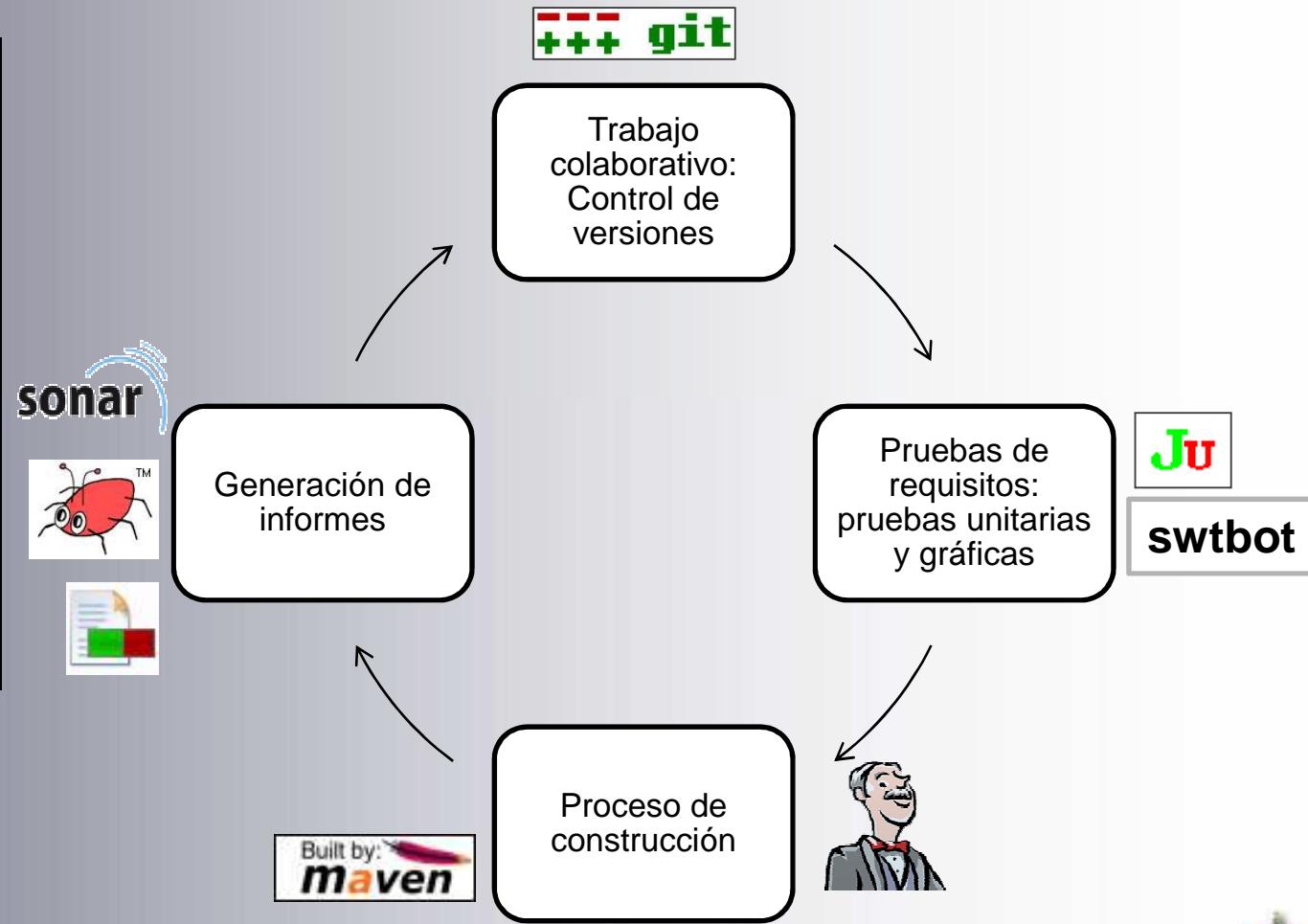
PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS

MODELO DE DESARROLLO



TÉCNICAS Y HERRAMIENTAS

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS



TÉCNICAS Y HERRAMIENTAS

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS

- Prototipos



- Diseño



- Desarrollo



- Documentación



- Tareas



DEMOSTRACIÓN



PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS



CONCLUSIONES

OBJETIVOS FUNCIONALES

- Navegación y visualización de refactorizaciones:
 - sistema de clasificación de refactorizaciones.
 - vista catálogo refactorizaciones.
- Mejorar asistente creación/edición refactorizaciones:
 - resumen elemento.
 - nuevo sistema de búsqueda de elementos.
- Mejora proceso instalación plugin.



CONCLUSIONES

OBJETIVOS TÉCNICOS

- Mejora del proceso de desarrollo:
 - uso de control de tareas y bugs, FogBugz.
 - utilización de control de versiones Git y repositorio web Github.
 - proceso de construcción del proceso mediante ejecución de un solo comando, gracias a Maven.
 - generación de informes de cobertura de código, métricas y defectos de código, mediante Sonar.
 - Utilización de un servidor de integración continua, Hudson.



CONCLUSIONES

- Mejora de calidad del código:
 - inclusión test gráficos, SWTBot.
 - mejoras siguientes:

Concepto	Antes	Después	Porcentaje
Juego de test	173	297	 70%
Cobertura test	38%	59%	 21%
Conformidad a los estándares del código*	57%	78%	 20%

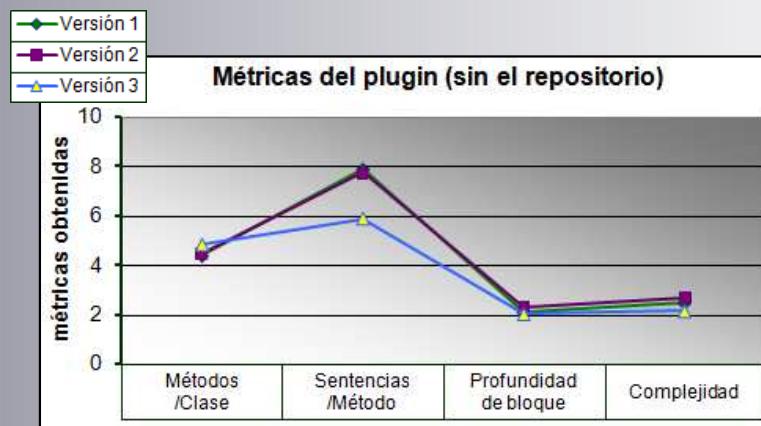
* basado en defectos por línea de código, se incrementa cuanto más conforme es.

- métricas:
 - incremento del tamaño del proyecto, en la parte del plugin, 46%.
 - el diseño se mantiene respecto a la versión anterior.
 - mejora en la profundidad media del bloque, la cual se encontraba por encima del límite recomendado en la versión anterior del proyecto.
 - descenso por debajo del mínimo, en la parte operativa, del número medio de sentencias por método.
 - significativos los valores de complejidad media.



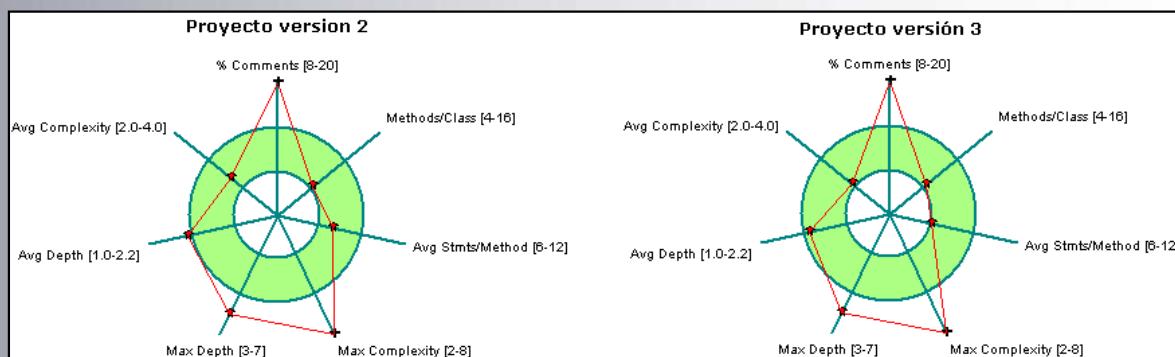
CLASIFICACIÓN DE REFACTORIZACIONES DYNAMIC REFACTORING PLUGIN 3.0

- PROYECTOS ANTERIORES**
- CONCEPTOS TEÓRICOS**
- OBJETIVOS DEL PROYECTO**
- PLANIFICACIÓN TEMPORAL**
- ESTUDIO VIABILIDAD**
- ASPECTOS RELEVANTES**
- TÉCNICAS Y HERRAMIENTAS**
- DEMOSTRACIÓN**
- CONCLUSIONES**
- COMPARACIÓN VERSIONES**
- LÍNEAS FUTURAS**
- AGRADECIMIENTOS**
- RUEGOS Y PREGUNTAS**



Intervalos UBU			
Min	Max	Valor	Cobertura UBU
2.3	46.3	42.2	<div style="width: 100%; background-color: green;"></div>
3.66	35.27	4.84	<div style="width: 100%; background-color: green;"></div>
3.72	64.87	5.89	<div style="width: 100%; background-color: green;"></div>
-	-	10.2	<div style="width: 100%; background-color: green;"></div>
1.33	4.59	2.02	<div style="width: 100%; background-color: green;"></div>
2	4	2.17	<div style="width: 100%; background-color: green;"></div>

M-Num	Nombre	Esperado	Plugin		
			Versión 1	Versión 2	Versión 3
M-1	Líneas de código	-	-	26.057	37.794
M-2	Nº clases	-	-	182	297
M-3	Nº sentencias	-	-	8.931	13.044
M-4	Nº medio de métodos por clase	[4-16]	4,39	4,46	4,84
M-5	Nº medio de sentencias por método	[6-12]	7,89	7,74	5,89
M-6	Profundidad media del bloque	[1-2,2]	2,09	2,29	2,02
M-7	Complejidad media	[2-4]	2,48	2,67	2,14



COMPARACIÓN DE VERSIONES

PROYECTOS ANTERIORES
CONCEPTOS TEÓRICOS
OBJETIVOS DEL PROYECTO
PLANIFICACIÓN TEMPORAL
ESTUDIO VIABILIDAD
ASPECTOS RELEVANTES
TÉCNICAS Y HERRAMIENTAS
DEMOSTRACIÓN
CONCLUSIONES
COMPARACIÓN VERSIONES
LÍNEAS FUTURAS
AGRADECIMIENTOS
RUEGOS Y PREGUNTAS

	DynamicRefactoringPlugin		
	Versión 1.0	Versión 2.0	Versión 3.0
Identificación refactorización usuario/plugin	✗	✗	✓
Exportar refactorizaciones	✗	✓	✓ ★
Vista de refactorizaciones disponibles	✗	✓	✓ ★
Creación/Edición/Borrado refactorizaciones	✓	✓	✓ ★
Copia refactorizaciones	✗	✗	✓
Ayuda para entradas y elementos	✗	✓	✓ ★
Búsqueda flexible de entradas y elementos	✗	✗	✓
Interfaz mejorada	✗	✓	✓
Categorización de refactorizaciones	✗	✗	✓
Vista clasificación catálogo de refactorizaciones	✗	✗	✓
Editor de clasificaciones	✗	✗	✓

★ funcionalidad mejorada destacable



LÍNEAS DE TRABAJO FUTURAS

- Desarrollo de un repositorio *online* de refactorizaciones.
- Actualización automática de las vistas ante cambios en la definición de refactorizaciones o/y clasificaciones.
- Internacionalización de la ayuda y de la búsqueda de entradas, acciones y predicados.
- Perfeccionamiento del mecanismo de búsqueda basada en Lucene. Por ejemplo, incorporando sinónimos de términos.
- Incorporación del ámbito de paquete al plugin. Muy útil para la refactorización AddOverrideAnnotation.
- Exportación/importación de clasificaciones.
- Previsualización de las transformaciones de código resultantes de aplicar una refactorización.
- Realizar un estudio de *usabilidad* del plugin, en colaboración con los profesores y alumnos de diferentes niveles.



AGRADECIMIENTOS

- Tutor: Raúl Marticorena Sánchez.
- Profesores de Ingeniería en Informática.
- Familiares, amigos y compañeros.

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS



RUEGOS Y PREGUNTAS

PROYECTOS ANTERIORES

CONCEPTOS TEÓRICOS

OBJETIVOS DEL PROYECTO

PLANIFICACIÓN TEMPORAL

ESTUDIO VIABILIDAD

ASPECTOS RELEVANTES

TÉCNICAS Y HERRAMIENTAS

DEMOSTRACIÓN

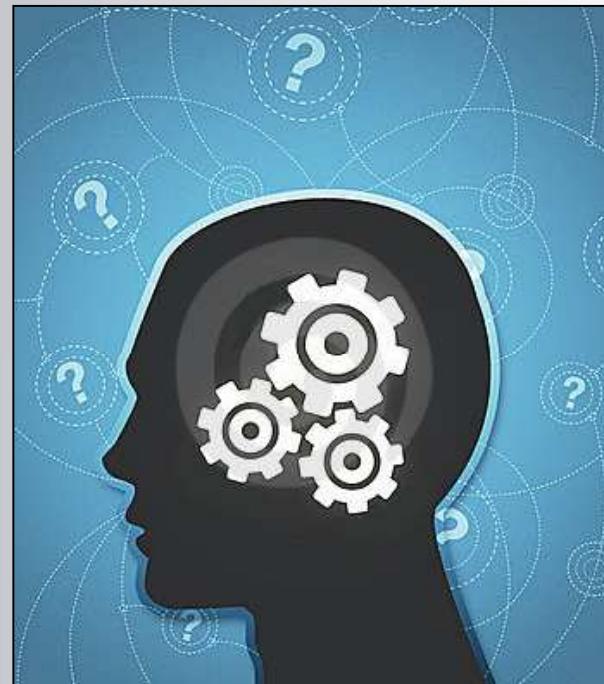
CONCLUSIONES

COMPARACIÓN VERSIONES

LÍNEAS FUTURAS

AGRADECIMIENTOS

RUEGOS Y PREGUNTAS



Gracias por su atención

