



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

## **Proyecto ShoggothViz<sup>1</sup>**

### *Metodología para la visualización de la evolución de código fuente*

**Desarrollado por:** Joan Sebastian Lopez Riaño

**Institución:** Universidad Nacional de Colombia – Sede Bogotá

**Programa:** Maestría en Ingeniería de Sistemas y Computación – Modalidad en profundización

**Fecha:** Marzo de 2017

## **METODOLOGÍA Y PRESUPUESTO**

### **1. Metodología**

#### **1.1 Estrategia General**

Para la efectiva implementación del proyecto se optará por una metodología exploratoria, es decir, que se partirá de una comprensión de la actualidad en el tema de visualización de *Software*, la implementación de metáforas y como se ha tratado el aspecto de la naturaleza evolutiva del código fuente.

Para llevar a cabo lo definido en el objetivo principal (diseñar una metodología para la visualización de la evolución del código fuente) se definirán un conjunto de posibles propuestas de metáforas visuales las cuales serán validadas mediante la construcción de una aplicación de software por cada una. Finalmente se espera aplicar la metodología propuesta a algunos casos de estudio y a partir de los productos obtenidos poder elaborar y exponer las conclusiones del trabajo realizado.

Los cuatro objetivos específicos se repartirán en cuatro fases de evolución del proyecto, las cuales se pueden resumir como : Levantamiento del estado del arte en el tema de visualizaciones enfocadas en la evolución del *software*, definición de la metáfora visual y sus características, implementación de la metodología y finalmente validación de la misma mediante algunos casos de estudio.

#### **1.2 Fases del proyecto**

##### **Fase I: Construcción del estado del arte**

Representa el esfuerzo inicial por comprender el estado del arte en el tema de visualizaciones de software, la evolución del y la implementación de metáforas visuales para la representación de las distintas características a analizar, clasificándolas según su relevancia y/o utilidad.

##### **Fase II: Definición de la metáfora visual**

Después de llegar a unas conclusiones respecto al estado actual del campo de estudio se planea explorar algunas propuestas para la metáfora final a seleccionar, escogiendo aquella que cuente con las bases más sólidas que justifiquen su usabilidad y pertinencia. De igual manera se definirá el conjunto de características a extraer tanto de los artefactos de código fuente.

---

1. Es un nombre de proyecto provisional. Al definirse la metáfora visual a implementar se definirá un nuevo nombre para el proyecto.

### Fase III: Implementación de la metodología

Definida la metáfora visual y las características del código fuente que se desea representar, se procederá a la implementación de dicha metodología mediante la construcción de una aplicación que tenga como objetivo generar las visualizaciones sobre los artefactos de código de un proyecto *open source* como caso de estudio.

Para la realización del prototipo se utilizará la metodología de desarrollo SCRUM<sup>2</sup> simplificada, la cual, aplicada a un equipo unipersonal con *sprints* o iteraciones semanales permitirá agilizar la generación cada semana de nuevas versiones que se acerquen más a la visualización deseado.

### Fase IV: Validación de la metodología

Para esta fase final se planea realizar un análisis exploratorio que tenga como objetivo sustentar la justificación de la metáfora visual seleccionada al estudiar distintas visualizaciones generadas por medio de la metodología propuesta aplicada a algunos artefactos de código fuentes que serán tratados como casos de estudio.

### 1.3 Actividades por fase

A continuación se presentan las actividades y productos por cada objetivo específico:

Fase I: Construcción del estado del arte	
OBJETIVO	<i>Investigar el estado del arte de trabajos relacionados metodologías para la visualización de Software usando metáforas visuales y que analicen a su vez la evolución del código fuente</i>
ACTIVIDADES	<ol style="list-style-type: none"><li>1. Definir un marco teórico mínimo para las temáticas de visualización de software.</li><li>2. Realizar un filtrado inicial de referencias recientes en el campo de la visualización de software</li><li>3. Realizar un segundo filtrado de referencias para seleccionar los proyectos relacionados con metáforas del código fuente</li><li>4. Elaborar las conclusiones del estado del arte</li></ol>
PRODUCTOS	<ol style="list-style-type: none"><li>1. Reporte técnico del marco teórico y estado del arte del proyecto</li></ol>

Fase II: Definición de la metáfora visual	
OBJETIVO	<i>Definir la metáfora visual a implementar y el conjunto de características a extraer de los artefactos de código fuente.</i>
ACTIVIDADES	<ol style="list-style-type: none"><li>1. Escoger y clasificar un conjunto inicial de características relacionadas con la evolución del <i>software</i> de acuerdo a las conclusiones del estado del arte.</li><li>2. Proponer y desarrollar varias propuestas de metáforas visuales para la representación de las características anteriormente seleccionadas.</li><li>3. Diseñar ilustraciones de ejemplo sobre las visualizaciones propuestas</li></ol>

<sup>2</sup> “...es un modelo de referencia que definen un conjunto de prácticas y roles, y que pueden tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto” - (Schwaber & Beedle, 2001)

	aplicadas a características extraídas de algunos artefactos de código fuente.
PRODUCTOS	<ol style="list-style-type: none"> <li>1. Reporte técnico de la descripción de la metáfora a implementar y de las otras posibles propuestas descartadas.</li> <li>2. Reporte técnico de la propuesta de la metáfora principal, utilidad y aplicaciones.</li> </ol>

Fase III: Implementación de la metodología	
OBJETIVO	<i>Desarrollar una aplicación de software que implemente la metodología propuesta y que tomando como entrada un conjunto de artefactos de código fuente produzca una visualización que represente distintas características a analizar de dichos elementos</i>
ACTIVIDADES	<ol style="list-style-type: none"> <li>1. Documentar los casos de uso y diagramas de flujo de la aplicación a desarrollar.</li> <li>2. Escoger la tecnología a implementar en la propuesta de visualización y conexiones con los repositorios de código fuente.</li> <li>3. Diseñar la arquitectura y el modelado de información para la generación de las visualizaciones.</li> <li>4. Implementar el diseño (desarrollo de la aplicación).</li> <li>5. Realización de iteraciones semanales que generen productos mínimos viables para la generación de las visualizaciones deseadas.</li> <li>6. Realización de controles semanales de lo que se ha hecho y lo que no, por cada versión del prototipo liberada.</li> </ol>
PRODUCTOS	<ol style="list-style-type: none"> <li>1. Prototipo de software por cada iteración realizada.</li> </ol>

Fase IV: Validación de la metodología	
OBJETIVO	<i>Validar la metodología desarrollada mediante la aplicación de la misma a algunos proyectos de código abierto que servirán de casos de estudio</i>
ACTIVIDADES	<ol style="list-style-type: none"> <li>1. Definir un grupo delimitado artefactos de código fuente pertenecientes a proyectos <i>open source</i>.</li> <li>2. Generar por cada grupo de artefactos una visualización usando la metodología propuesta.</li> <li>3. Generar un listado de conclusiones respecto a la visualización obtenida.</li> </ol>
PRODUCTOS	<ol style="list-style-type: none"> <li>1. Reporte técnico con la descripción del proceso de validación y de conclusiones respecto a posibles usos y aplicaciones de la metodología propuesta.</li> </ol>

#### **1.4 Cronograma**

Se distribuirá la realización del proyecto a lo largo de un (1) año de trabajo completo distribuido de la siguiente manera:

##### **Fase I**

**Duración:** Dos (2) meses

**Fecha de inicio:** Enero 2018

**Fecha de finalización:** Marzo 2018

##### **Fase II**

**Duración:** Tres (3) meses

**Fecha de inicio:** Marzo 2018

**Fecha de finalización:** Junio 2018

##### **Fase III**

**Duración:** Tres (3) meses

**Fecha de inicio:** Junio 2018

**Fecha de finalización:** Septiembre 2018

##### **Fase IV**

**Duración:** Un (1) mes

**Fecha de inicio:** Noviembre 2018

**Fecha de finalización:** Diciembre 2018

	2018	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEP	OCT	NOV	DIC
FASE I	Definir un marco teórico mínimo para las temáticas de visualización de software.												
	Realizar un filtrado inicial de referencias recientes en el campo de la visualización de software												
	Realizar un segundo filtrado de referencias para seleccionar los proyectos relacionados con metáforas del código fuente												
	Elaborar las conclusiones del estado del arte												
FASE II	Escoger y clasificar un conjunto inicial de características relacionadas con la evolución del <i>software</i> de acuerdo a las conclusiones del estado del arte.												
	Proponer y desarrollar varias propuestas de metáforas visuales para la representación de las características anteriormente seleccionadas.												
	Diseñar ilustraciones de ejemplo sobre las visualizaciones propuestas aplicadas a características extraídas de algunos artefactos de código fuente.												
FASE III	Documentar los casos de uso y diagramas de flujo de la aplicación a desarrollar.												
	Escoger la tecnología a implementar en la propuesta de visualización y conexiones con los repositorios de código fuente.												
	Diseñar la arquitectura y el modelado de información para la generación de las visualizaciones.												
	Implementar el diseño (desarrollo de la aplicación).												
	mínimos viables para la generación de las visualizaciones deseadas.												
FASE IV	Definir un grupo delimitado artefactos de código fuente pertenecientes a proyectos <i>open source</i> .												
	Generar por cada grupo de artefactos una visualización usando la metodología propuesta.												
	Generar un listado de conclusiones respecto a la visualización obtenida.												

**Figura.1 : Cronograma de trabajo por cada actividad para el año 2018**

## 2. Presupuesto

La principal fuente de financiamiento para el proyecto propuesto sera mixta, es decir que estará compuesta de recursos propios del autor y de parte de los recursos que provee la Universidad Nacional de Colombia en el convenio que como estudiante de la Maestría de Ingeniería de Sistemas y Computación el autor es beneficiario.

### 2.1 Recursos físicos

- Equipo de cómputo para desarrollo, redacción y acceso a la Internet (Propio del investigador)
- Equipo de cómputo para el desarrollo de la etapa de experimentación (Propio del investigador)
- Fotocopias y material de papelería.
- Artículos y libros de referencia (Facilitados por la división de bibliotecas de la Universidad Nacional de Colombia)

### 2.2 Descripción de los costos y rubros

La siguiente descripción de gastos se aplica para un cálculo anual, es decir durante la duración de los 12 meses de ejecución del proyecto. La estimación de costos de personal resulta de un estimado en la experiencia del costo de un desarrollador junior (para el caso del investigador) y de estimados en los gastos de profesores. Otros rubros como el gasto de Internet y equipos son también estimados con base en la experiencia y tasas actuales de prestadores de servicios públicos.

CONCEPTO	FUENTE	TOTAL
GASTOS DE PERSONAL		
Salario investigador	Investigador	\$26.400.000,00
Salario Director	Universidad Nacional de Colombia	\$18.000.000,00
EQUIPOS Y SOFTWARE		
Computador de escritorio	Investigador	\$2.300.000,00
MATERIALES DE SUMINISTRO Y BIBLIOGRAFÍA		
Conexión a Internet	Investigador	\$960.000,00
Bibliografía y acceso a artículos académicos	Universidad Nacional de Colombia	\$1.000.000,00
TOTAL		\$48.660.000,00

**Figura 2. Cuadro de gastos**