

## Procesos para el desarrollo de Software libre

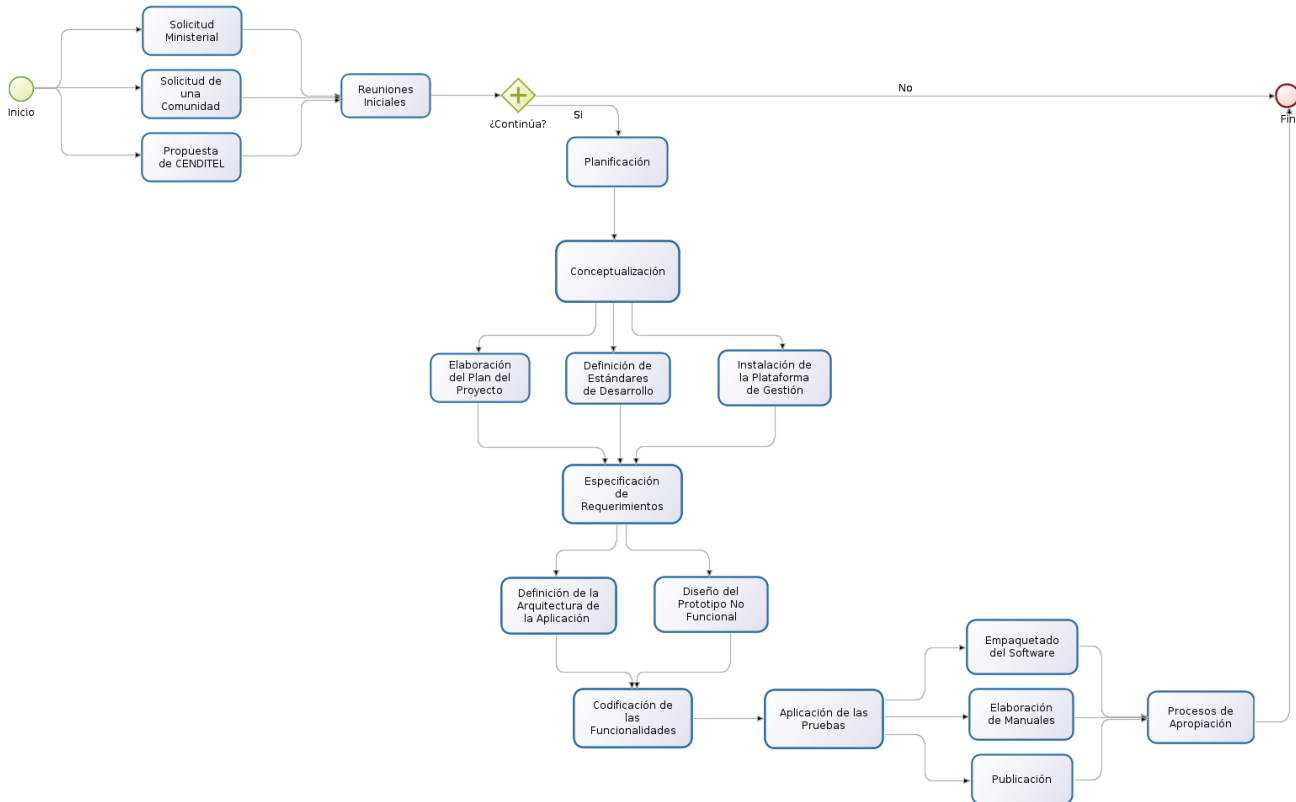


Figura 1. Flujo de trabajo de la metodología de desarrollo de software libre

Acción específica	Desarrollar Sistema de seguimiento y control a proyectos de Alta Ingeniería
Nombre del producto	Sistema de seguimiento y control a proyectos de Alta Ingeniería (SEPAI)
Fecha de entrega del producto	30 de Septiembre de 2016
Equipo Responsable	Leonel P. Hernandez Macchiarulo, Gabriela Villasana y un vacante(Analista de sistemas)
Cara visible	Leonel Paolo Hernandez Macchiarulo
Director:	Santiago Roca

### Actividades:

N°	Nombre	
1	Conceptualización	En este proceso se recopila y analiza información concerniente a los procesos que se requieren automatizar en una aplicación de software, con el objetivo de comprender el dominio de la aplicación a desarrollar, así como los problemas o necesidades de los usuarios en relación a dichos procesos, todo ello con la finalidad de plantear una propuesta de desarrollo de software acorde a las necesidades de los usuarios
	Descripción	Identificar procesos a automatizar. (2%) Identificar problemáticas y necesidades en torno a estos procesos.(1%)

		Elaborar diagramas de procesos a automatizar.(3%) Validar los diagramas de procesos y de actividades con el equipo.(1%) Correcciones a partir de lo acordado en la reunión.(3%)
	Fecha de entrega:	29 de Abril de 2016
	Ponderación de la actividad (porcentaje):	10%
	Persona responsable:	Gabriela Villasana, Kleivymar Montilla
2	Elaboración del Plan del proyecto:	El plan del proyecto debe contener: 1) Priorización y orden de dependencia de las funcionalidades del software; 2) Estudio de riesgos; 3) Priorización de desarrollo de las funcionalidades, 4) Cronograma de desarrollo del proyecto. En el cronograma de desarrollo del proyecto se debe especificar el número de iteraciones a realizar, indicando las funcionalidades a desarrollar y las fechas de inicio y fin de cada una.
	Descripción	Definir y priorizar las funcionalidades del software. (1%) Definir el orden de dependencia entre las funcionalidades del software.(2%) Realizar estudio sobre los riesgos de desarrollo del software. (2%) Elaboración del plan por iteración de desarrollo. (4%)
	Fecha de entrega:	13 de Mayo de 2016
	Ponderación de la actividad (porcentaje):	10%
	Persona responsable:	Leonel Hernandez
3	Definición de Estándares de desarrollo:	Constituyen las normas o patrones de referencia que se deben implementar en el desarrollo de aplicaciones de software. Entre los estándares de desarrollo más comunes se encuentran: normas de codificación, normas y esquemas de seguridad, estándares de interfaz u/s, entre otros.
	Descripción	Definir estándares de desarrollo (1%)
	Fecha de entrega:	13 de Mayo de 2016
	Ponderación de la actividad (porcentaje):	1%
	Persona responsable:	Leonel Hernandez
4	Instalación de la Plataforma de gestión del proyecto:	Es importante utilizar una herramienta que apoye la gestión de la práctica de desarrollo, a través de la cual se puedan realizar actividades que faciliten dicha práctica, el desarrollo colaborativo y la apropiación del software. Entre las actividades a las que se hace referencia se encuentran: la publicación del software y su respectiva documentación, la gestión de errores, la asignación y seguimiento de tareas en el Equipo de Desarrollo, el control de versiones del código y su documentación, entre otras.
	Descripción	
	Fecha de entrega:	13 De Mayo de 2016

	Ponderación de la actividad (porcentaje):	0%
	Persona responsable:	Leonel Hernandez
5	Especificación de requerimientos:	Consiste en describir de manera detallada cada una de las funcionalidades o requerimientos que debe cumplir la aplicación a desarrollar. En esta especificación se debe indicar el flujo de actividades que implica el llevar a cabo cada funcionalidad del sistema. Para la especificación de los requerimientos de la aplicación se utilizarán diagramas de casos de uso con su respectiva descripción textual.
	Descripción	Elaborar los diagramas de casos de uso. (2%) Describir los casos de uso. (3%) Discutir con el equipo de desarrollo la descripción de los casos de uso. (2%) Correcciones a partir de lo acordado en las reuniones. (3%)
	Fecha de entrega:	27 de Mayo de 2016
	Ponderación de la actividad (porcentaje):	10%
	Persona responsable:	Gabriela Villasana , Kleivymar Montilla
6	Especificación de arquitectura de la aplicación:	La arquitectura del software integra sus componentes y la interacción (comunicación) entre éstos, teniendo en cuenta las funcionalidades y las limitaciones tecnológicas existentes. Puede ser representada a través de diferentes diagramas o vistas arquitectónicas.
	Descripción	Especificar los datos a intercambiar con otras aplicaciones de software. (5%) Estudio del lenguaje de programación a utilizar para el desarrollo del software. (5%)
	Fecha de entrega:	1 de Julio de 2016
	Ponderación de la actividad (porcentaje):	10%
	Persona responsable:	Leonel Hernandez
7	Diseño del Prototipo no funcional:	El prototipo no-funcional de la interfaz u/s está compuesto por el diagrama jerárquico de pantallas, en el cual se muestran las diferentes interfaces a través de las cuales el usuario podrá acceder a las funcionalidades que ofrece la aplicación. El diseño de este prototipo solo muestra las formas y colores que presentarán las pantallas de la interfaz de la aplicación, sin incluir funcionalidad. Este tipo de prototipo se utiliza para validar con el usuario el estilo de las pantallas de interfaz que presentará la aplicación.
	Descripción	Diseño del prototipo no funcional de la interfaz de usuario. (3%) Diseñar las pantallas no funcionales correspondientes a la interfaz gráfica del software. (2%) Validar el prototipo no funcional de la interfaz con el equipo. (1%) Correcciones a partir de lo acordado en las reuniones. (2%) Correcciones a partir de lo acordado en reunión con usuarios. (2%)
	Fecha de entrega:	1 de Julio de 2016
	Ponderación de la actividad (porcentaje):	10%
	Persona responsable:	Gabriela Villasana, Kleivymar Montilla
8	Codificación de las funcionalidades:	En esta fase se codifican las funcionalidades de la aplicación de software correspondientes a la iteración actual, se construye la interfaz de usuario y la base de datos. De esta manera, en esta fase, con cada iteración de desarrollo, se obtiene una

		nueva versión de la aplicación, clasificada como versión beta, es decir, una versión sobre la cual se deben realizar un conjunto de pruebas como pruebas funcionales y no funcionales.
	Descripción	Construir la base de datos conforme al modelo de datos persistentes. (15%) Codificar los componentes requeridos para construir las funcionalidades. (15%) Realizar las pruebas unitarias a los componentes adaptados y/o codificados y corregir los errores encontrados. (3%) Construir la interfaz de usuario correspondiente a las funcionalidades. (5%) Documentar el código fuente. (2%)
	Fecha de entrega:	30 de Agosto de 2016
	Ponderación de la actividad (porcentaje):	40 %
	Persona responsable:	Leonel Hernandez
9	Aplicación de las pruebas (funcionales, no funcionales, seguridad y rendimiento según sea el caso):	En esta fase se elaboran y aplican pruebas funcionales y no funcionales a cada versión del software, así como pruebas de regresión y de instalación, con lo cual se facilita la detección temprana de errores y/o incompatibilidades en el código. Estas pruebas deben ser elaboradas y aplicadas por probadores de software, quienes se recomiendan deben ser personas distintas a quienes codifican la aplicación.
	Descripción	Pruebas por parte del equipo del proyecto. (2%) Correcciones a partir de las pruebas realizadas. (2%) Correcciones a partir de lo acordado en la reunión. (1%)
	Fecha de entrega:	30 de septiembre de 2016
	Ponderación de la actividad (porcentaje):	5%
	Persona responsable:	Leonel Hernandez
10	Elaboración de manuales:	Es importante contar con el manual de instalación y el manual del usuario. El primero contiene las indicaciones de cómo instalar la aplicación desarrollada. El manual de usuario representa una guía de uso de la aplicación, en la cual se indican los pasos a efectuar para utilizar los servicios (funcionalidades) que ofrece la aplicación.
	Descripción	
	Fecha de entrega:	30 de septiembre de 2016
	Ponderación de la actividad (porcentaje):	4%
	Persona responsable:	Gabriela Villasana y Kleivymar Montilla
11	Empaquetado del software:	A fin de que el software desarrollado pueda ser portable, es decir, operar en varias distribuciones, se recomienda empaquetar el software para más de una distribución. Empaquetar el software facilita en gran medida el proceso de instalación de éste.
	Descripción	



	Fecha de entrega:	30 de Septiembre de 2016
	Ponderación de la actividad (porcentaje):	0%
	Persona responsable:	Leonel Hernandez
12	Publicación:	Una vez se hayan corregido los errores reportados por los usuarios en las versiones beta se publica la versión estable del software. Con la intención de facilitar el proceso de apropiación del software, la publicación de las versiones estables debe incluir no sólo la publicación del código y los paquetes respectivos, sino también el manual de usuarios y el resto de documentación asociada a la versión publicada.
	Descripción	
	Fecha de entrega:	30 de Octubre de 2016
	Ponderación de la actividad (porcentaje):	0%
	Persona responsable:	Leonel Hernandez
13	Procesos de Apropiación:	Consiste en las actividades de encuentro y socialización con otros actores para la apropiación del resultado del proyecto. Puede incluir: talleres, cursos, reuniones, participación en eventos, entre otros.
	Descripción	
	Fecha de entrega:	30 de Noviembre de 2016
	Ponderación de la actividad (porcentaje):	0%
	Persona responsable:	Leonel Hernandez