**[Nombre del proyecto]**

AlimNova®

9 Marzo de 2010

Versión 1.0

[Logos]

LAURA ARIAS

ANDREA FAJARDO

NESTOR DIAZGRANADOS

GERMAN MORALES

DAVID SUAREZ

WILLIAM RAMÍREZ

**PAGINA DE FIRMAS**

El presente documento es aprobado por las personas referenciadas a continuación:

**Firma Cliente:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Miguel Eduardo Torres

**Firma Director de Proyecto:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Laura Arias

**Firma Director de Desarrollo:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

German Morales

**Firma Director de Calidad y Manejo de Riesgos:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

David Suarez

**Firma Administrador de Configuración y Documentación:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

William Ramirez

**Firma Arquitecto:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Nestor Diazgranados

**Firma Analista de Requerimientos :**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Andrea Fajardo

**HISTORIAL DE CAMBIOS**

*En esta sección se presenta una tabla que describe la evolución y los cambios que se le realizan al documento desde que se inicia hasta que se haya llegado a la versión base que es entregada al cliente.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Sección del documento modificada** | **Descripción de cambios (corta)** | **Responsable (S)** |
| *Indica la versión del documento, que depende según la forma de administración de configuraciones seleccionada.* | *Se incluye la fecha en la que fue realizado el cambio del documento.* | *Permite especificar las secciones del documento que fueron modificadas.* | *Es un pequeño resumen de los cambios más relevantes que fueron realizados en la versión* | *Indica las personas del grupo de trabajo que son responsables del o los cambios realizados en el documento.* |

Tabla 1: Historial cambios

**PREFACIO**

*Según el diccionario de la Real Academia Española [6] el prefacio es una especie de prólogo o introducción de un libro, por lo tanto, este debe ser una especie de abrebocas del documento SPMP, debe estar centrado en su alcance, el por qué es importante y para quien se realiza, es necesario resaltar que el prefacio gira en torno al documento SPMP y no se centra en el proyecto por el cual se está realizando.*

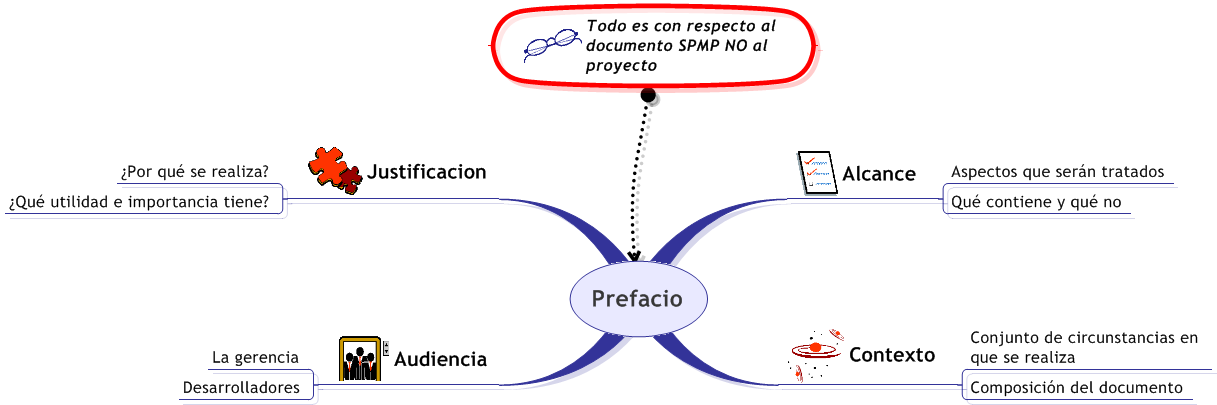


Ilustración 1: Prefacio

**TABLA DE CONTENIDO**

[LISTA DE FIGURAS 7](#_Toc175389657)

[LISTA DE TABLAS 8](#_Toc175389658)

[1. VISION GENERAL DEL PROYECTO 9](#_Toc175389659)

[1.1 RESUMEN DEL PROYECTO 9](#_Toc175389660)

[1.1.1 Propósito, Alcance y Objetivos 9](#_Toc175389661)

[1.1.2 Suposiciones y Restricciones 10](#_Toc175389662)

[1.1.3 Entregables del Proyecto 11](#_Toc175389663)

[1.1.4 Resumen de Calendarización y Presupuesto 11](#_Toc175389664)

[1.2 EVOLUCIÓN DEL PLAN 12](#_Toc175389665)

[2. REFERENCIA 13](#_Toc175389666)

[3. DEFINICIONES Y ACRONIMOS 15](#_Toc175389667)

[4. ORGANIZACIÓN DEL PROYECTO 18](#_Toc175389668)

[4.1 Interfaces Externas 18](#_Toc175389669)

[4.2 Estructura Interna 18](#_Toc175389670)

[4.3 Roles y Responsabilidades 20](#_Toc175389671)

[5. PLAN DE PROCESOS DE GESTIÓN 26](#_Toc175389672)

[5.1 PLAN DE ARRANQUE 26](#_Toc175389673)

[5.1.1 Plan de Estimación 26](#_Toc175389674)

[5.1.2 Plan de Personal 27](#_Toc175389675)

[5.1.3 Plan de Entrenamiento de Personal 29](#_Toc175389676)

[5.2 PLAN DE TRABAJO 31](#_Toc175389677)

[5.2.1 Actividades de Trabajo 31](#_Toc175389678)

[5.2.2 Cronograma 33](#_Toc175389679)

[5.2.3 Asignación De Recursos 33](#_Toc175389680)

[5.2.4 Asignación De Presupuesto 34](#_Toc175389681)

[5.3 PLAN DE CONTROL 34](#_Toc175389682)

[5.3.1 Plan de Control de requerimientos 34](#_Toc175389683)

[5.3.2 Plan de Control de cronograma 35](#_Toc175389684)

[5.3.3 Plan de Control de Presupuesto 37](#_Toc175389685)

[5.3.4 Plan de Control de Calidad 37](#_Toc175389686)

[5.3.5 Plan de Reportes 38](#_Toc175389687)

[5.3.6 Plan de Recolección de Métricas 39](#_Toc175389688)

[5.4 PLAN DE ADMINISTRACIÓN DE RIEGOS 40](#_Toc175389689)

[5.5 PLAN DE CIERRE 42](#_Toc175389690)

[6. PLAN DE PROCESOS TÉCNICOS 43](#_Toc175389691)

[6.1 MODELO DE CICLO DE VIDA DEL PROCESO 43](#_Toc175389692)

[6.2 Métodos, Herramientas y Técnicas 46](#_Toc175389693)

[6.3 Plan de Infraestructura 50](#_Toc175389694)

[6.4 Plan de Aceptación del Producto 50](#_Toc175389695)

[7. PLAN DE PROCESOS DE SOPORTE 51](#_Toc175389696)

[7.1 PLAN DE ADMINISTRACIÓN DE LA CONFIGURACIÓN 51](#_Toc175389697)

[7.2 PLAN DE VERIFICACIÓN Y VALIDACIÓN 56](#_Toc175389698)

[7.3 PLAN DE DOCUMENTACIÓN 59](#_Toc175389699)

[7.4 PLAN DE ASEGURAMIENTO DE LA CALIDAD 61](#_Toc175389700)

[7.5 REVISIONES Y AUDITORIAS 66](#_Toc175389701)

[7.6 PLAN DE RESOLUCIÓN DE PROBLEMAS 67](#_Toc175389702)

[7.7 PLAN DE ADMINISTRACIÓN DE SUBCONTRATOS 68](#_Toc175389703)

[7.8 PLAN DE MEJORAS DEL PROCESO 68](#_Toc175389704)

[8. ANEXOS 70](#_Toc175389705)

[REFERENCIAS DE LA PLANTILLA **Error! Bookmark not defined.**](#_Toc175389706)

# LISTA DE FIGURAS

Ilustración 1: Prefacio 4

Ilustración 2: Propósito, alcance y objetivos 9

Ilustración 3: Objetivos SMART 10

Ilustración 4: Suposiciones 10

Ilustración 5: Restricciones 11

Ilustración 6: Entregables del proyecto 11

Ilustración 7: Artefactos gráficos **Error! Bookmark not defined.**

Ilustración 8: Interfaces externas 18

Ilustración 9: Estructura interna 20

Ilustración 10: Organigrama 20

Ilustración 11: Plan de estimación 26

Ilustración 12: Plan de personal 27

Ilustración 13: Plan de entrenamiento de personal 29

Ilustración 14: Plan de trabajo 31

Ilustración 15: Proceso, actividad y tarea 31

Ilustración 16: Descripción actividades y tareas 33

Ilustración 17: Control de cronograma 36

Ilustración 18: Control de presupuesto 37

Ilustración 19: Parámetros de aceptación 39

Ilustración 20: Plan de administración de riesgos 40

Ilustración 21: Plan de administración de riesgos 41

Ilustración 22: Plan de cierre 42

Ilustración 23: Plan de cierre 43

Ilustración 24: Plan de procesos técnicos 43

Ilustración 25: Ciclo de vida del proyecto 44

Ilustración 26: Métodos, Herramientas y Técnicas 47

Ilustración 27: Plan de Infraestructura 50

Ilustración 28: Plan de aceptación del producto 50

Ilustración 29: Plan de procesos de soporte 51

Ilustración 30: Definiciones básica del plan de procesos de soporte 51

Ilustración 31: Numeración de versiones 52

Ilustración 32: Contenido plan de documentación 61

Ilustración 33: Lista de chequeo para el aseguramiento de la calidad 62

Ilustración 34: Proceso de pruebas 64

Ilustración 35: Plan de mejoras de proceso 68

LISTA DE TABLAS

[Tabla 1: Historial cambios 3](#_Toc175389742)

[Tabla 2: Resumen de calendarización y presupuesto I 12](#_Toc175389743)

[Tabla 3: Resumen de calendarización y presupuesto II 12](#_Toc175389744)

[Tabla 4: Control de evolución del plan 13](#_Toc175389745)

[Tabla 5: Referencias 14](#_Toc175389746)

[Tabla 6: Herramientas para manejo de referencias 15](#_Toc175389747)

[Tabla 7: Acrónimos 17](#_Toc175389748)

[Tabla 8: Ejemplo de responsabilidades por roles **Error! Bookmark not defined.**](#_Toc175389749)

[Tabla 9: Tabla de roles, habilidades e integrantes 29](#_Toc175389750)

[Tabla 10: Planeación actividad de entrenamiento 30](#_Toc175389751)

[Tabla 11: Procesos y actividades 32](#_Toc175389752)

[Tabla 12: Actividades y tareas 32](#_Toc175389753)

[Tabla 13: Presupuesto 34](#_Toc175389754)

[Tabla 14: Control de calidad 38](#_Toc175389755)

[Tabla 15: Reportes 38](#_Toc175389756)

[Tabla 16: Recolección de métricas 39](#_Toc175389757)

[Tabla 17: Criterios de análisis de riesgos 41](#_Toc175389758)

[Tabla 18: Matriz de Riesgos 42](#_Toc175389759)

[Tabla 19: Peticiones de cambio 53](#_Toc175389760)

[Tabla 20: Seguimiento de cambios 53](#_Toc175389761)

1. VISION GENERAL DEL PROYECTO

## RESUMEN DEL PROYECTO

### Propósito, Alcance y Objetivos

*En esta sección se describen el propósito, alcance y objetivos del proyecto que se planea realizar. En los diagramas presentados a continuación se muestra la definición para cada uno de los términos utilizados [6].*

Ilustración 2: Propósito, alcance y objetivos

*Se recomienda para la realización de los objetivos tener en cuenta las características para objetivos SMART, que se describen en la figura 3 [12].*

Ilustración 3: Objetivos SMART

### Suposiciones y Restricciones

* Suposiciones: *Se describe cuales van a ser las precondiciones que se deben cumplir para que se lleve a cabo tanto en el proyecto como en el desarrollo de software de forma exitosa [4]. Algunos temas que se ponen como ejemplo son:*

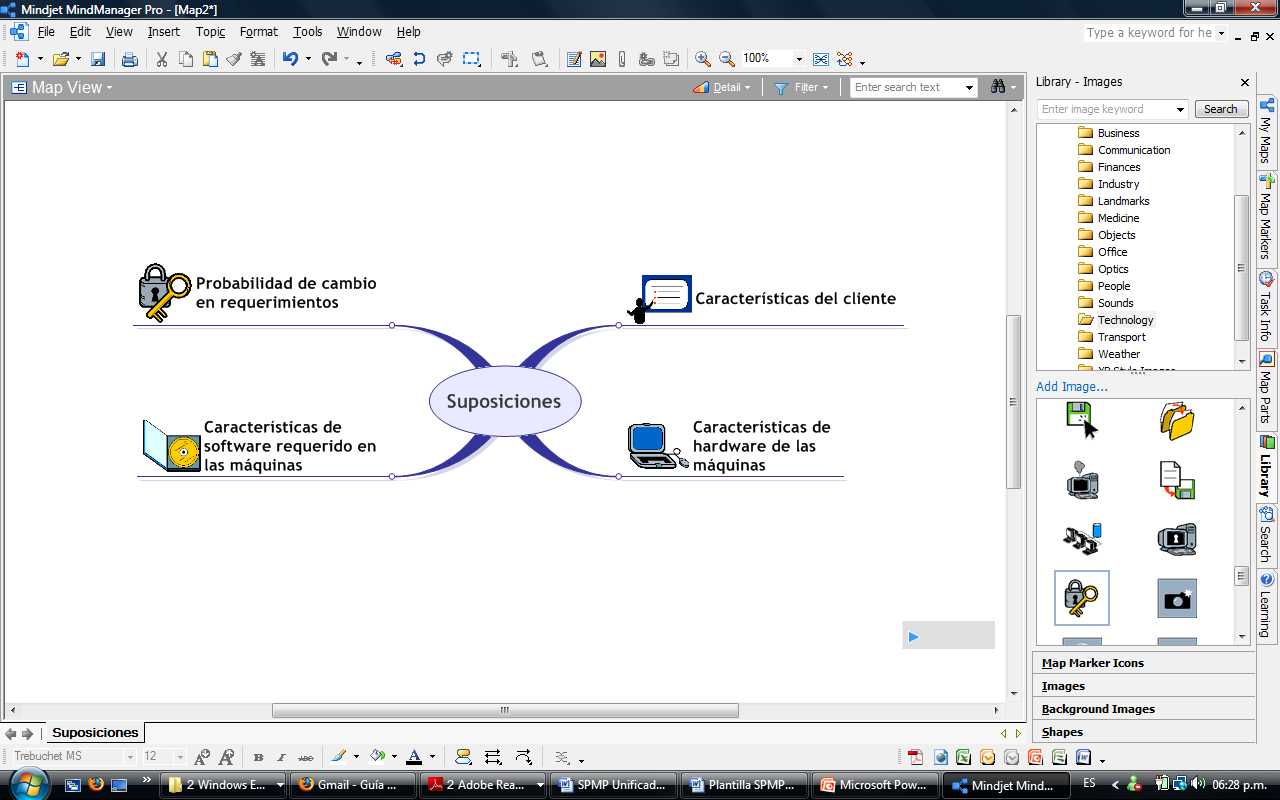
**

Ilustración 4: Suposiciones

* Restricciones: *Se deben identificar las limitaciones y condiciones en el desarrollo de software [4], lo cual puede incluir:*

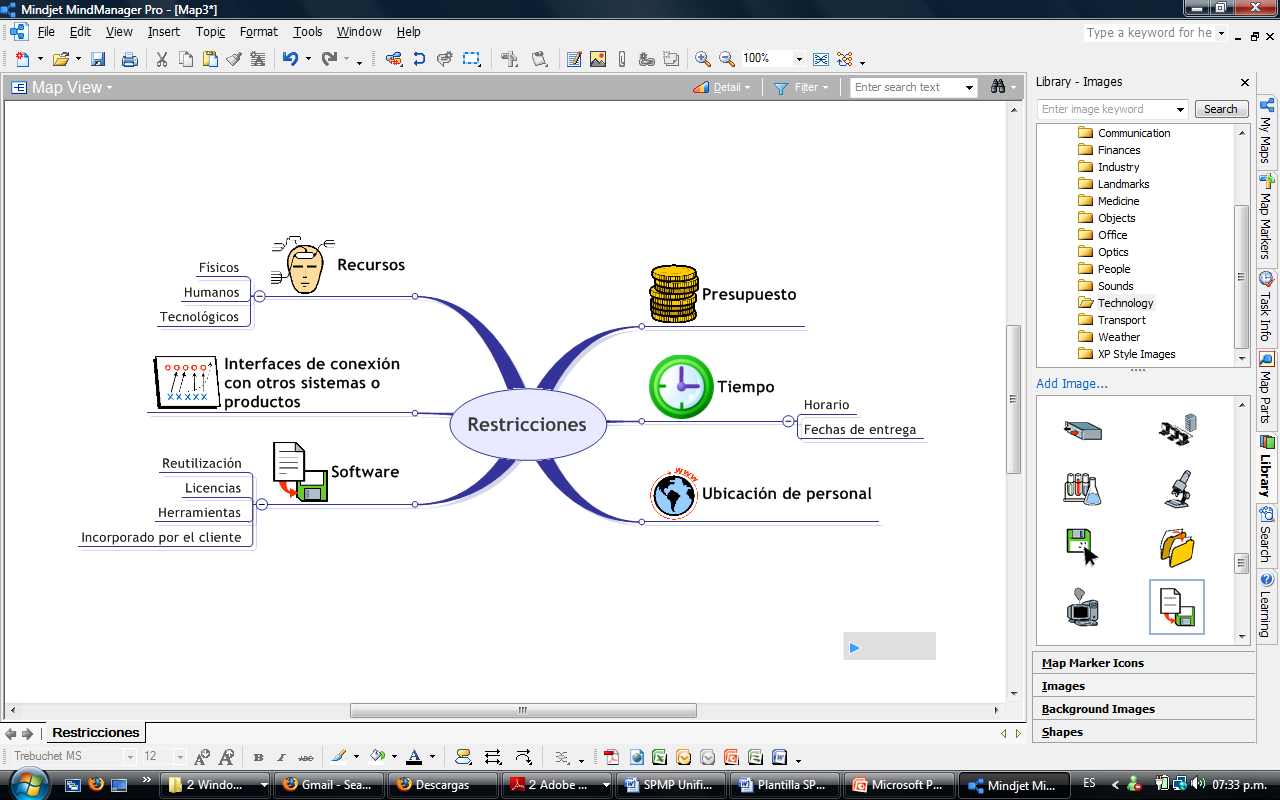


Ilustración 5: Restricciones

### Entregables del Proyecto

*Se debe hacer una lista de los productos de trabajo que serán entregados al cliente, esto incluye documentos, la aplicación, manuales, entre otros, que son generados a lo largo del desarrollo del proyecto especificando en cada uno:*

Ilustración 6: Entregables del proyecto

### Resumen de Calendarización y Presupuesto

*Se deben enunciar de manera breve las principales actividades que se planean llevar a cabo a lo largo del desarrollo del producto de software, las cuales se recomienda agrupar en entregas o hitos específicos; también se debe estimar el presupuesto para cada una de estas actividades y el tiempo en el que se pretende realizar o cumplirlas.*

*Como ejemplo se presenta a continuación una tabla que resume la calendarización y presupuesto de un proyecto agrupado en n entregas:*

|  |  |  |  |
| --- | --- | --- | --- |
| **Entregas** | **Actividades** | **Duración (en horas, minutos...)** | **Presupuesto estimado (en pesos, dólares...)** |
| **Entrega o hito 1** | **Actividad 1** |  |  |
| **Actividad 2** |  |  |
| **…** |  |  |
| **Actividad m** |  |  |
| **Entrega o hito 2** | **Actividad 1** |  |  |
| **Actividad 2** |  |  |
| **…** |  |  |
| **Actividad m** |  |  |
| **…** | **Actividad 1** |  |  |
| **Actividad 2** |  |  |
| **…** |  |  |
| **Actividad m** |  |  |
| **Entrega o hito n** | **Actividad 1** |  |  |
| **Actividad 2** |  |  |
| **…** |  |  |
| **Actividad m** |  |  |
|  | **TOTAL** |  |  |

Tabla 2: Resumen de calendarización y presupuesto I

|  |  |  |
| --- | --- | --- |
|  | **Fecha de inicio** | **Fecha fin** |
| **Entrega o hito 1** |  |  |
|
| **Entrega o hito 2** |  |  |
|
| **…** |  |  |
|
| **Entrega o hito n** |  |  |
|

Tabla 3: Resumen de calendarización y presupuesto II

## EVOLUCIÓN DEL PLAN

*Se deben planificar las actividades de cronograma y fuera de este, que se van a llevar a cabo para gestionar la evolución y actualización del SPMP. Se deben especificar actividades como la realización de versiones, de revisiones y de correcciones, cada una de las cuales debe tener una descripción, persona(s) responsable(s), y periodo de tiempo en que se implementara.*

*A continuación se presenta un modelo de tabla para llevar a cabo el control de la evolución del plan.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Actividad** | **Descripción** | **Fecha o Periodo** | **Responsable** |
| *Aquí se incluye el nombre de la actividad* | *Definición de la actividad, indicando cómo se desarrollará* | *Periodo de tiempo en el que se pretende realizar la actividad* | *Indica la o las personas del equipo de desarrollo encargadas de verificar y validar la actividad* |

Tabla 4: Control de evolución del plan

# REFERENCIA

1. TORRES, Miguel Eduardo, DIAZ, Luis Carlos, ALDANA Juan Carlos y. Sección Materias/Ingeniería de Software/Proyecto – Material de clase – Diapositivas, Métricas. 2009 Disponible en: <http://sophia.javeriana.edu.co/~metorres/Materias/IngSoftware/Diapostivas/ProcesoSW_Metricas.pdf>
2. Ximena Higuera Moriones, Andrés Teyes Aguilar, Nicolás Gaitán,Manuel Valencia, Jairo Ipial, Christian Lemus. Ancient Risk .Incasoft. 13 Octubre 2009.
3. <http://classes.cec.wustl.edu/~cse528/SPMPReport05.pdf>
4. Miguel Eduardo Torres. Sección Materias/Ingenieria de Software/Material de Clase – Diapositivas/Puntos Funcionales [En línea]. Disponible en [http://sophia.javeriana.edu.co/~metorres](http://sophia.javeriana.edu.co/%7Emetorres)
5. Software Project Management Plan (SPMP) template.

Online <http://users.csc.calpoly.edu/~jdalbey/205/Mgmt/SPMP.html>.

|  |  |
| --- | --- |
| **Material** | **Descripción** |
| ***Artículos, revistas y publicaciones*** | 1. ***Apellidos e iníciales*** *de los autores, que deben ir sólo las iníciales en mayúsculas. Por ejemplo: Torres M, (seguido por comas)(signo de punto después del último autor)* 2. ***Título del trabajo*** *en su versión original sin traducir (punto).* 3. ***Publicación****, título abreviado de la revista en que este se publica (punto).* 4. ***Año de la publicación*** *que puede incluir* ***mes y día****; (punto y coma).* 5. ***Volumen y el Número****, éste  último entre paréntesis (6): - sin espaciado y seguido de dos puntos.* 6. ***Páginas****. Especificando página inicial  – final. Para abreviar se indica el final con un solo dígito. Por ejemplo: 431-7. En este caso no se escribe la palabra “pág.”  Al final se cierra con punto. [19]* |
| ***Ejemplo (Referencia):*** *[1] Henri E. Bal, Jennifer G. Steiner, Andrew S. Tanenbaum. Programming Languages for Distributed Computing Systems, ACM Computing Survey. Septiembre 1989; (Volumen 21, No3): 262-322.* |
| ***Libros y Monografías*** | 1. ***Apellidos e iníciales*** *de todos los autores (o editores, compiladores, etc.) o el nombre completo de una entidad colectiva.* 2. ***Título.*** 3. ***Número de la edición****.* 4. ***Lugar de publicación****. Es decir la ciudad.* 5. ***Entidad editorial****.* 6. ***Año****.* 7. *Solo en caso de referencia es decir cita textual se agregarán* ***los******números del volumen y páginas consultadas****,* ***serie a que pertenece y número que le corresponde en la serie****. [1xxxy]* |
| ***Ejemplo (Bibliografía):*** *Larman C. UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. 2nd ed. Aragón DF. Madrid: Pearson Educación. S.A.; 2003.*  ***Ejemplo (Referencia):*** *[1] Larman C. UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. 2nd ed. Aragón DF. Madrid: Pearson Educación. S.A.; 2003. Páginas [132,176]* |
| ***Material Electrónico*** | 1. ***Autor o autores****, escribiendo apellidos e iníciales.* 2. ***Título del trabajo****.* 3. ***Revista o publicación****, precisando entre  corchetes […] que se trata de una serie en Internet.* 4. *Nombre publicitario del host o página (nombre publicitario no dirección web) que contiene el material electrónico.* 5. ***Año de Publicación****. Precisando entre  corchetes […] que ha sido citado, es decir revisado por Ud. en determinada fecha.* 6. ***Volumen y el Número****, éste  último entre paréntesis y sin espaciado. Seguido de dos puntos y entre  corchetes […] las páginas aproximadas del documento.* 7. *Al final precisar la* ***dirección o URL*** *en la que está disponible.[1xxxy]* |
| ***Ejemplo (Bibliografía):*** *Greiner C. Programación IV. [Curso en Internet]. Argentina: Facultad de Ciencias Exactas y Naturales y Agrimensura; 2006. [Citado 2007 Feb. 16]. Disponible en:*[*http://exa.unne.edu.ar/depar/areas/informatica/programacion4/public\_html/documentos/tema3.pdf*](http://exa.unne.edu.ar/depar/areas/informatica/programacion4/public_html/documentos/tema3.pdf) |

Tabla 5: Referencias

*Existen diferentes herramientas que permiten realizar las referencias y citar la bibliografía con facilidad, a continuación se describen algunas:*

|  |  |
| --- | --- |
| ***El administrador de fuentes proporcionado por Microsoft Office 2007*** | *Esta herramienta permite realizar las referencias con los siguientes formatos o estilos: APA,* *Chicago, GB7714, GOST - Orden de nombre, GOST - Orden de título, ISO 690 - Primer elemento y fecha, ISO 690 - Referencia numérica, MLA,* *SIST02 y Turabian.*  *También permite insertar citas y bibliografías creando una especie de repositorio de los materiales utilizados, los cuales pueden ser incluidos en cualquier lugar del documento.* |
| ***JabRef Referencie Manager*** | *Este es un manejador de referencias bibliográficas que permite obtener archivos con formato BibTex. Requiere la máquina virtual de Java con versión igual o superior a la 1.4.2 para su ejecución.*  *Para mayor información dirigirse a la página* [*http://jabref.sourceforge.net/*](http://jabref.sourceforge.net/) *[11], aquí podrá descargar la aplicación y encontrar la documentación asociada.* |

Tabla 6: Herramientas para manejo de referencias

# DEFINICIONES Y ACRONIMOS

*En esta sección se pretende proveer definiciones de todos los términos y acrónimos requeridos para el óptimo entendimiento del SPMP; incluyendo tanto técnicos como del ámbito del proyecto. Es necesario agregar la definición de términos del negocio para que cualquiera de los stakeholders (cliente, desarrolladores, diseñadores y demás equipo de trabajo) pueda entender el SPMP y el proyecto que se va a realizar.*

*Se deben referenciar todas las fuentes de las cuales se haya obtenido las definiciones de los términos y acrónimos. Además es recomendable ordenar alfabéticamente las definiciones para facilitar la búsqueda y el entendimiento.*

*Por lo general entre los desarrolladores de software se genera un leguaje común como nickname, password entre otros, pero en contraste, estos términos no son comunes para los clientes o gerentes de proyecto; por lo tanto, es recomendable buscar su traducción que para este caso sería pseudónimo y contraseña, o de lo contrario introducir estos términos en esta sección.*

*A continuación se presenta un repositorio de definiciones y acrónimos utilizados en la mayoría de documentos de desarrollo de software (En caso de no existir ninguna palabra que haga parte de un conjunto asociado a una letra es recomendable no especificarla en los acrónimos ya que se vería como una letra vacía):*

|  |
| --- |
| ***802.11:*** *El protocolo IEEE 802.11 o WI-FI es un estándar de protocolo de comunicaciones de la IEEE que especifica las normas de funcionamiento en una WLAN.* |
| ***D*** |
| ***DBMS (DATA BASE MANAGEMENT SYSTEM****): sistema dedicado a servir de interfaz entre las bases de datos y las aplicaciones que las utilizan.*  ***DET - Data Element Type:*** *Tipo de dato único no recursivo en un ILF o EIF.*  ***Disconnect Protection – (Protección de desconexión):*** *Situación o escenario en el cual un sistema garantiza que una pérdida de conexión no dejara ningún dato o información inconsistente.* |
| ***E*** |
| ***EI - External Inputs:*** *Es un proceso elemental en cuyo dato cruza la frontera de afuera hacia adentro.*  ***EO - External Outputs:*** *Un proceso elemental en cuyo dato derivado pasa atreves de la frontera de adentro hacia afuera.*  ***EQ - External Inquiry:*** *Un proceso elemental con ambos componentes de adentro y desafuera que resulta en recuperación de uno o más archivos de lógica interna y archivos de interfaz externa.* |
| ***F*** |
| ***FTR - File Type Referenced:*** *Es un tipo de archivo referenciado por una transacción. Un FTR debe además ser un archivo de lógica interna o archivos de interfaz externa.* |
| ***G*** |
| ***GUI****: En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario (IGU), provee una forma amigable de interacción con un sistema informático.* |
| ***I*** |
| ***ILF - Internal Logical Files****: Un grupo de datos lógicamente relacionados que reside totalmente dentro de la frontera de la aplicación y es mantenido a través del EI.* |
| ***R*** |
| ***RET - Record Element Type:*** *Un subgrupo identificable de elementos de datos dentro de un ILF o un EIF.* |
| ***S*** |
| ***SDD****: Documento que describe el modelo de diseño del sistema*  ***SPMP****: Documento que controla un proyecto de software. El SPMP define las actividades, productos de trabajo, indicadores de avance y recursos asignados al proyecto. En el SPMP también están definidos los procedimientos administrativos y convenciones aplicables al proyecto, como el reporte de estado, la administración del riesgo, y la administración de contingencias*  ***SRS****: Documento que describe el sistema de requerimiento de software* |
| ***U*** |
| ***UML- Unified Modelling Language:*** *lenguaje de modelado de sistemas de software.* |
| ***V*** |
| ***VAF - Value Adjustment Factor:*** *Está basado en las siguientes 14 Características Generales del Sistema.*   1. *Comunicaciones de datos* 2. *Procesamiento de datos distribuidos* 3. *Desempeño* 4. *Configuración pesada usada.* 5. *Tasa de transacciones.* 6. *Entrada de datos en línea.* 7. *Eficiencia para el usuario final.* 8. *Actualizaciones en línea.* 9. *Procesamiento complejo.* 10. *Reusabilidad.* 11. *Facilidad de instalación.* 12. *Facilidad de operación.* 13. *Sitios múltiples.* 14. *Facilidades de cambio.* |
| ***W*** |
| ***Warnings****: Advertencias de un posible error en el código en el momento de compilar el mismo.*  ***WLAN****: Es un sistema de comunicación de datos inalámbrico, utilizado como alternativa las redes de área local cableadas o como extensión de éstas. Utiliza tecnología de radiofrecuencia que permite mayor movilidad a los usuarios al minimizar las conexiones cableadas.* |

Tabla 7: Acrónimos

# ORGANIZACIÓN DEL PROYECTO

## Interfaces Externas

Son aquellas organizaciones o personas que no pertenecen a AlimNova® pero si vas a estar relacionados directamente con el desarrollo del proyecto, y con los miembros de AlimNova®, bien sea asesorándonos, prestándonos las salas, brindándonos información acerca de requerimientos y demás aspectos propios del proyecto, basándonos en esto destacamos como nuestras interfaces externas las siguientes:

Ilustración 1: Interfaces externas

## Estructura Interna

AlimNova® es una entidad compuesta por 6 estudiantes de ingeniería de sistemas de la Pontificia Universidad Javeriana, con roles asignados de acuerdo a sus capacidades y habilidades **[].** Bruegge (ir a referencias) definió algunas pautas para el éxito de un equipo de trabajo, entre ellas que el número de integrantes debe oscilar entre 5 y 6 personas, de manera que el trabajo no tenga sobrecarga o que pocas responsabilidades sean asignadas; esto con el fin de que haya calidad y eficiencia en el desarrollo del trabajo; además de la facilidad para programar reuniones presenciales.

AlimNova® es una organización de tipo funcional debido a su división en departamentos o áreas independientes, en la cual cada área tiene un líder específico encargado de determinadas obligaciones; este líder le asigna a su equipo de trabajo actividades para cumplir con sus respectivas responsabilidades acorde con los roles correspondientes.

Una de las ventajas principales de que sea funcional es que los objetivos se cumplen a cabalidad en un tiempo menor y generando un producto de mayor calidad puesto que cada persona se especializa en lo que le concierne; y cada empleado se concentra en su tarea específica. Una desventaja podría ser que tiende a bajar la cooperación entre departamentos lo cual implicaría un riesgo (ir a la sección de riesgos); dado esto, también hemos decidido que se debe manejar el modelo bazar (ver definiciones y referencias);este es un modelo que promueve la colaboración entre equipos; fue desarrollado en 1997 por Eric Raymond (www.emagister.com/uploads.../Comunidad\_Emagister\_28741\_Bazar.ppt), su principal objetivo es promover la colaboración entre todos los integrantes del equipo; pero respetando los roles establecidos para cada persona **[ ] ,** también establece unas bases para la buena comunicación entre el gerente de proyecto y los directores de los departamentos.

La comunicación dentro de la organización constituida de manera jerárquica se puede manejar de 3 formas: **comunicación** **vertical ascendente:** que va desde los empleados a los directivos, esta comunicación es importante porque por medio de esta se conocen los problemas y necesidades del empleado; la **comunicación** **vertical descendente:** va en sentido inverso a la anterior, de los directores a los empleados, esta comunicación tiene como fin informar a los empleados las tareas que deben cumplir; y finalmente la **comunicación** **horizontal:** este tipo de comunicación es realizado entre personas de un mismo nivel jerárquico.

*También exponer cómo se gestionará internamente el equipo, formas de comunicación, toma de decisiones, y organización en general, especificando en que se enfoca esta.*

**VERTICAL DESCENDENTE**

**VERTICAL ASCENDENTE**

**HORIZONTAL**

Ilustración 2: Formas de Comunicación

Dados los protocolos de comunicación descritos anteriormente, la comunicación será como se muestra en la figura 2.

La estructura interna de la organización será de la siguiente manera:

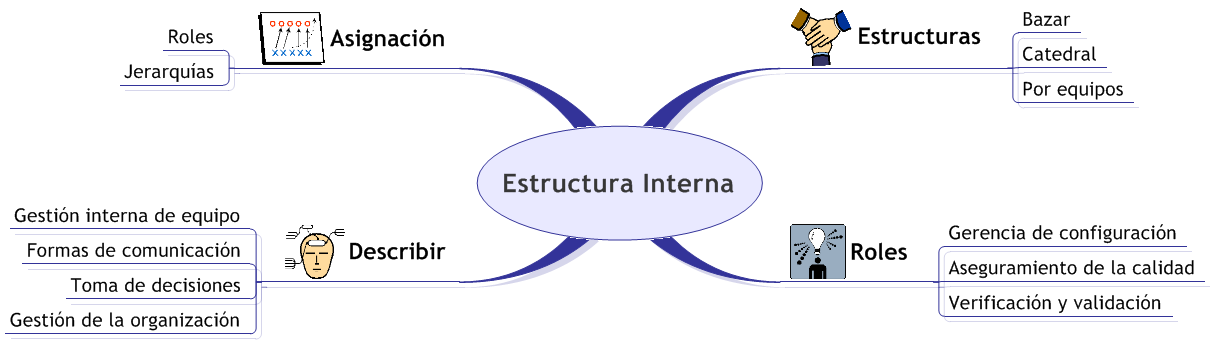


Ilustración 3: Estructura interna

Ilustración 4: Organigrama

## Roles y Responsabilidades

La asignación de roles fue un proceso llevado a cabo en la segunda reunión de AlimNova®. En ella se identificaron habilidades y capacidades de cada uno de los integrantes, de manera que el recurso de personal se aprovechará al máximo; estas se relacionaron con las responsabilidades correspondientes a cada rol, también se tuvo en cuenta que cada persona estuviera a gusto con el área en el que se va a desempeñar.

Existen diferentes tipos de habilidades, entre ellas se encuentran: (Bruegge)

*.*

Ilustración 5: Habilidades

* + **Dominio de Aplicación:** Se refiere a habilidades relacionadas con los términos y procedimientos bancarios, las fórmulas de calificación crediticia en el contexto de las aplicaciones de banca, etc.
  + **Comunicación:** Tener la capacidad de comunicación con los Stakeholders (revisar glosario) que no están familiarizados con el desarrollo del Software, además de la capacidad de negociar, principalmente la capacidad de expresar ideas complejas.
  + **Técnicas:** incluye el conocimiento en la tecnología que se usará en el proyecto, además de buen desempeño como programador, capacidad de detectar riesgos y poder encontrar soluciones eficientes con problemas imprevistos.
  + **Calidad:** ser una persona centrada en los detalles, capacidad de identificar límites y restricciones, generar casos de prueba apropiados y capacidad de seguir a cabalidad los procesos.
  + **Gestión:** habilidades personales, motivación, retroalimentación, capacidad de priorizar los riesgos del proyecto dadas las limitaciones.

Dadas las habilidades principales descritas anteriormente, a continuación se relacionara cada rol con las habilidades

Ilustración 6: Relación Roles - Habilidades

Definidas las habilidades de cada rol, a continuación mostraremos los roles definidos a cada integrante del grupo:

Ilustración 7: Definición Roles

Las responsabilidades que debe desempeñar cada rol se describirán en la siguiente matriz, además de las necesidades de capacitación en algún área:

|  |  |  |
| --- | --- | --- |
| ROL | RESPONSABILIDADES | CAPACITACIONES |
| Director de Proyecto | * Enviar las actas de las reuniones semanales * Mantener el calendario actualizado * Asignar tareas de manera equitativa a los integrantes del grupo * Hacer planeación correcta con relación a las entregas y tareas. * Supervisar el estado de las tareas * Comunicación con el cliente * Mantener la organización * Identificación, dependencia y calendarización de tareas * Mantener la comunicación con el cliente |  |
| Director de Desarrollo | * Configuración de pruebas * Diseñar estándares para el código de programación * Definir especificaciones técnicas relacionadas con el Software y Hardware * Análisis y diseño del problema |  |
| Director de Calidad y manejo de riesgos | * Inspecciones de documentación y de código * Diseño de casos de prueba * Diseño de métricas para calificar la calidad del producto del trabajo * Definición de riesgos tanto administrativos como técnicos |  |
| Administrador de configuraciones y documentación | * Administrar las versiones de código y de documentación * Administra la información del equipo |  |
| Arquitecto | * Elaborar el diseño de alto nivel * Apoyo en áreas sin experiencia por los integrantes del equipo * Manejo del dominio de aplicación y de negocio * Soluciona problemas técnicos * Juicio crítico e imparcial. |  |
| Analista de Requerimientos | * Mantener la comunicación con el cliente * Supervisar la documentación del producto de acuerdo a las necesidades del negocio. |  |

**Tabla 1: Resumen Roles, Capacidades, Responsabilidades**

## Reglamento y Sanciones

En esta sección del SPMP decidimos incluirla puesto que es importante que en cualquier lugar existan reglas, de esta manera se evitan conflictos entre personas. AlimNova® ha definido unas reglas que deben ser cumplidas por cada uno de los miembros, y en caso de que no se cumplan deberán aceptar sin refuta las decisiones respectivas.

La siguiente tabla describe el reglamente y las sanciones:

|  |  |  |
| --- | --- | --- |
| REGLA | SANCIÓN | TIPO DE REGLA |
| Las reuniones se revisión se efectuaran los lunes y viernes en el horario de 11 am – 1pm | Si reitera en fallas 3 veces y estas no son justificadas tendrá un 10% menos en la nota del proyecto actual | **REUNIÓN** |
| La máxima duración de las reuniones de revisión serán de 2 horas | No aplica | **REUNIÓN** |
| La llegada debe ser puntual, se tendrán 10 minutos de espera a partir de la hora acordada | $ 3000 las primeras 2 veces, la tercera cancelara $5000 | **REUNIÓN** |
| Si algún miembro no puede asistir a las reuniones de revisión, deberá enviar con mínimo 1 día de anterioridad la causa de la inasistencia. La inasistencia deberá ser por alguna de las siguientes causas: enfermedad o motivos familiares, en todo caso el grupo decidirá la validez de la excusa | Si reitera 3 veces y las fallas no son justificadas tendrá un 10% menos en la nota del proyecto actual | **REUNIÓN** |
| La no entrega de tareas asignadas, la falta de interés en cuanto al trabajo a realizar, al igual que la 3 reiteración de entregas tardes. | **Genera la expulsión inmediata del equipo** | **TAREAS** |
| La entrega de una tarea tarde pero que se considere de calidad, se debatirá entre el grupo si influyeron otros factores. | -10% de la entrega actual | **TAREAS** |
| Debe existir el respeto entre los integrantes del grupo. | Algún roce será sancionado entre los relacionados con la compensación de las partes de alguna manera. | **RELACIONES INTERPESONALES** |
| Después de cada entrega se supervisara que las cuentas estén saldadas para los deudores | En caso de que deba dinero, se le cobraran ciertos intereses. | **TESORERIA** |

**Tabla 2: Reglas y Sanciones**

Motivaciones a los integrantes por su buen desempeño, es un punto muy importante que se debe tener en cuenta, con el fin de mantener el positivismo dentro del grupo, a continuación se definirán las motivaciones:

|  |  |
| --- | --- |
| DESEMPEÑO | MOTIVACIÓN |
| En cada entrega se decidirá quién fue el que más se destaco por sus trabajos de buena calidad, buena actitud, responsabilidad. Si hay más de un miembro que se destaque se realizara un consenso. | +10% en la nota de una entrega; generalmente la entrega actual. |
| La puntualidad de un miembro del equipo | +10% en la nota de una entrega. |
| Cada finalización de una entrega el grupo hará una especie de integración de manera que se consuman los recursos recolectados en la entrega actual. |  |

**Tabla 2: Desempeño y Motivaciones**

# PLAN DE PROCESOS DE GESTIÓN

## PLAN DE ARRANQUE

AlimNova® cuenta con seis personas en su grupo de trabajo, las cuales estarán durante todo el desarrollo del mismo, cada una de las personas involucradas va a tener un rol definido son las responsabilidades que este tiene **[],** así mismo el grupo como tal estará guiado por una serie de reglas, que de no ser cumplidas se realizarán los correspondientes llamados de atención y memorandos acorde a la falla, del mismo modo los miembros recibirán incentivos, como motivaciones ó felicitaciones, por un buen trabajo ó desempeño durante el tiempo de las entregas, estas normas fueron aceptadas y acordadas por los miembros del grupo **[**], cada una de estas tareas tienen como finalidad contribuir al desarrollo del proyecto, para así poder tener buenos resultados y mantener la claridad en el comportamiento que cada uno de los miembros debe tener con su trabajo y con el grupo, siendo uno de los más importantes el respeto y la responsabilidad ya que si alguien se retrasa o realiza un trabajo incompleto ó de mala calidad, este afectará el trabajo del grupo y con ello la calidad del trabajo completo.

Durante cada una de las fases no se contratará a nadie aparte de los miembros ya mencionados, dado el caso que se requiera complementar la información o aclarar conceptos y demás aspectos relacionados con el desarrollo del proyecto, se pedirá una asesoría tanto al profesor de la materia, Ingeniero Miguel Torres, como a otros profesores del departamento de Ingeniería de sistemas, para que nos ayuden con ciertas dudas o en la mejora de planteamientos para llevar a cabo el desarrollo del trabajo.

### Plan de Estimación

Ilustración 7: Plan de estimación

*Para poder estimar el cronograma, los costos del proyecto y requerimientos se debe escoger una herramienta que permitirá manejar la ruta critica, los sobrecostos, y todo tipo de requerimientos, en cuanto a recursos del proyecto (Ej.: Microsoft Project). Además podría darse una breve descripción del porqué se escogió cada una de las herramientas.*

*Es necesaria la elección del método para estimar los costos del proyecto (Ej.: estimación mediante puntos funcionales, Cocomo, líneas de código) [4], se debe exponer las cualidades que brinda el método escogido en el progreso del proyecto y se desarrolla el método escogido en esta sección.*

*Puesto que la idea de este punto es mostrar cómo se hará el desarrollo completo del proyecto es válido incluir en el cronograma [7] hitos que permitan determinar el estado actual del proyecto con respecto al cronograma, estos hitos por ejemplo podrían incluir todos los planes (SPMP, SRS, SDD) y otras actividades que podrían verse como “reuniones con el cliente”.*

Para este plan emplearemos puntos de función , teniendo en cuenta los casos de usos que serán definidos acorde a las funcionalidades que identifiquemos durante el desarrollo del proyecto de software, que para este caso es desarrollar un monopolio.

A partir de los casos de uso identificados y los requerimientos obtenidos a partir de ellos AlimNova® definirá el precio determinado para el desarrollo del proyecto y los requerimientos que se compromete a desarrollar.

Para realizar estos cálculos debemos tener una aproximación de los costos del proyecto incluyendo todas las tareas involucradas, teniendo en cuenta, las horas trabajadas por cada persona y los costos de los recursos que requiere cada persona para realizar su trabajo.

El sueldo de un ingeniero no graduado es de 60.000 por hora, teniendo en cuenta que se van a trabajar n horas por semana, y tomando un mes como 4 semanas el valor correspondiente a la mano de obra es de : 60000\*n\*4\*6 , es decir que el valor respecto al personal es de $.

Respecto al manejo del calendario se realizará con Google Calendar, una herramienta que permite llevar la agenda donde tendremos claridad acerca de las reuniones, revisiones, entregas entre otros aspectos del proyecto.

### Plan de Personal

Ilustración 8: Plan de personal

El propósito de este grupo es mantener el respeto entre cada uno de los miembros del grupo, respetar cada uno de los pensamientos y aportes de cada uno de los miembros y realizar las críticas determinadas de la mejor forma posible y con el ánimo de ayudar y mantener la motivación de la persona, manteniendo la calidad del trabajo y la calidad del grupo como tal.

Así mismo cada uno de los miembros del grupo conoce las reglas del grupo, las cuales ha aceptado y también ha aportado ideas para generar castigos ó premios acorde a determinados resultados, comportamientos **[]**.

Nuestro objetivo es generar un producto de trabajo bueno, que cumpla con altas métricas de calidad así como también es contribuir al desarrollo de cada uno de los miembros respecto al trabajo en grupo, a los conocimientos de la carrera y en el conocimiento de sí mismo, ya que requiere de un conocimiento interno, para poder definir y asignarle un rol a determinada persona, acorde con las habilidades y gustos de cada uno de los miembros, de esta forma cada persona tendrá un rol que le guste, y en el cual tenga habilidad con la finalidad que dé lo mejor de sí al desarrollo de este, le dedique tiempo y soporte a los demás en explicaciones o aspectos que tengan que ver con su rol, en esta sección podemos ver los roles, sus responsabilidades, las actividades relacionados con cada uno y demás temas relacionados con los roles **[ ]** .

Para definir estos roles tuvimos en cuenta el test de los colores, para así poder identificar las fortalezas y debilidades de cada uno de los integrantes, como también dialogamos entre nosotros dando a conocer cada una de nuestras fortalezas y debilidades, así como también la afinidad con determinado tema.

La finalidad de definir los roles es también poder clarificar las responsabilidades de cada uno de esto y con ello poder saber quién en determinado momento ó etapa del proyecto va a guiar y coordinar el desarrollo de cada una de las actividades de esa etapa, así como también es quien va a tener los criterios para definir las métricas de calidad de este tema, todo esto basado en el conocimiento e investigación realizado por esta persona, durante el desarrollo del proceso y de las reuniones con el cliente y demás profesores relacionados con cada tema a tratar.

Los roles asignados al cada miembro del grupo, fueron explicados detalladamente en la sección [**]**, basándonos en la descripción allí realizada en esta parte resaltaremos en qué fase del proyecto interviene cada rol y que recursos necesita para hacerlo.

|  |  |  |
| --- | --- | --- |
| ROL | RECURSOS | FASE Ó FASES INVOLUCRADOS |
| GERENTE | Computador, personal del grupo, herramientas de gestión de proyectos | Durante todo el desarrollo del proyecto, herramientas de gestión, para cada fase de reubicación del cronograma. |
| GERENTE DESARROLLO | Computador, herramientas de desarrollo, personal de apoyo | Luego de la fase de diseño, una vez este sea aceptado y se inicie fase de desarrollo. |
| CONTROL DE CALIDAD | Computador, definición métricas, documentos finalizados para revisión | Cada que este cerca una pre-entrega, cada que un documento, sea unificado, finalizado y luego de revisión cruzada entre los miembros del grupo |
| DOCUMENTACIÓN | Computador | Durante cada hito, se requiere la realización de un documento que informe el proceso y los resultados. |
| ARQUITECTO | Computador, herramientas de diagramas | Luego de tener los requerimientos claros y definidos con el cliente, se procede al diseño del sistema. |

Tabla 8: Tabla de roles, fases y recursos

### Plan de Entrenamiento de Personal

Ilustración 9: Plan de entrenamiento de personal

Basándonos también en las debilidades presentadas por cada persona del equipo y de diálogos entre los miembros del grupo, detectamos que es importante tener conocimientos avanzados acerca de Word, ya que pocos manejamos actividades avanzadas, manejo de referencias, tablas de contenido y demás actividades relacionadas con el desarrollo del documento y es por ello que decidimos aprender acerca de estas actividades, para hacer el trabajo más rápido y fácil.

Otra herramienta que notamos que fue es importante saber manejar es Excel, ya que al igual que Word tiene actividades muy importantes que puede ayudar al desarrollo del proyecto, para ordenar la información, presentarla de una forma más ordenada, etc, es por esto que se requiere también la capacitación del grupo en Excel.

Respecto al desarrollo es importante conocer la herramienta en la cual se va a desarrollar la cual es flash y es importante también que los miembros del grupo conozcan esta herramienta y la sepan usar, para poder tener resultados más rápidos y productivos en el momento del desarrollo de la interfaz.

Respecto a la parte de la capacitación tendremos en cuenta el tiempo que toma la persona en capacitarse si es una herramienta nueva y luego explicarle al grupo la herramienta como tal, si la persona ya sabe manejarla se omitirá el tiempo de aprendizaje y se dispondrá en el calendario el tiempo acorde a la herramienta para poder capacitar a todas las personas del grupo, así mismo se debe tener en cuenta que las personas del equipo de trabajo luego de la capacitación se comprometen a seguir trabajando en el conocimiento de la herramienta para conocerla y manejarla mejor.

Otra herramienta importante que debemos aprender cada uno de los integrantes de AlimNova® es manejar el repositorio, ya que por medio de esta herramienta podremos manejar las versiones de los documentos de cada una de las entregas, referencias, documentos para investigación, etc, que sería de utilidad para uno o todos los miembros del equipo de trabajo, ya que es importante tener un espacio al que todos tenemos acceso y que necesitamos para comunicarnos mejor y manejar mucho mejor la información que cada miembro del grupo tiene o aporta.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Nombre (a)*** |  | ***Fecha (f)*** |  |
| ***Debilidad (b)*** |  | | |
| ***Participantes (c)*** |  |  | |
|  |  | |
|  |  | |
| ***Responsables (d)*** |  | | |
| ***Recursos (e)*** |  | | |
|  | | |
|  | | |

Tabla 9: Planeación actividad de entrenamiento

1. *Nombre de la actividad: se incluye un nombre y un numeral de modo que la localización de dicha actividad sea fácil de encontrar, en caso de modificaciones.*
2. *Debilidad a ser atacada: en este campo se pude incluir una breve descripción del problema y exponer la necesidad de la capacitación.*
3. *Participantes con la debilidad: si la falta de conocimiento no afecta a todos los miembros del equipo, el nombramiento de los roles involucrados debe hacerse específicamente.*
4. *Responsables del entrenamiento: dependiendo si son externos o integrantes del grupo se debe nombrar a la(s) persona(s) encargada(s) de la capacitación.*
5. *Recursos: si es necesario la adquisición de programas especiales y/o textos, artículos o papers, estos deben incluirse en este punto.*
6. *Día del entrenamiento: ya que la realización del cronograma de todo el semestre se debe hacer en este plan (SPMP) estos días de entrenamiento deben ser tenidos en cuenta para la entrega final del cronograma*

## PLAN DE TRABAJO

Ilustración 10: Plan de trabajo

### Actividades de Trabajo

Basándonos en el ciclo de vida de desarrollo de software que decidimos aplicar a nuestro proyecto el cual es el modelo cascada, vamos a tener en cuenta que cada hito va a terminar en el momento en que realizamos una entrega, en el cual a partir de la finalización realizamos el planteamiento de las actividades del siguiente hito, teniendo en cuenta el tiempo de capacitación de ser necesario para la nueva fase y las actividades de planeación y corrección de la entrega anterior, acorde a las observaciones de nuestro cliente, Miguel Torres, y por supuesto las actividades propias de la etapa correspondiente al desarrollo de la siguiente entrega.

*Para el desarrollo del proyecto se deben plantear las actividades que se van a ejecutar durante todo su ciclo de vida, estas actividades “iníciales” deben incluir desde la realización de los planes y documentos de diseño (SPMP, SRS y SDD) hasta las actividades de la entrega final al cliente (aplicación final, manuales entre otros).*

*Se aconseja establecer una distribución del trabajo de manera jerárquica, por ejemplo, utilizando los conceptos de proceso, actividad y tareas [9].*

Ilustración 11: Proceso, actividad y tarea

*Una vez se ha hecho una primera división, las actividades programadas para cada entrega deberían estar lo suficientemente claras como para poder ser descompuestas en las verdaderas tareas que finalmente serán ejecutadas por el equipo; los procesos se pueden determinar identificando las principales etapas del proyecto. Después de identificar las actividades se procede a atomizar las actividades en tareas y se deben describir cada una de ellas. . La tabla 11 presenta un posible formato para llevar a cabo la división de procesos en actividades, mientras que la tabla 12 muestra una guía para la división en tareas.*

|  |  |
| --- | --- |
| **Proceso** | **Actividades** |
| ***Primera Entrega*** | 1. *Actividad 1.1* 2. *Actividad 1.2* 3. *Actividad 1.3* |
| ***Segunda Entrega*** | 1. *Actividad 2.1* 2. *Actividad 2.2* 3. *Actividad 2.3* |
| ***Tercera Entrega*** | 1. *Actividad 3.1* 2. *Actividad 3.2* 3. *Actividad 3.3* |
| ***Cuarta Entrega*** | 1. *Actividad 4.1* 2. *Actividad 4.2* 3. *Actividad 4.3* |

Tabla 10: Procesos y actividades

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Proceso x** | | | | | | |
| ***Actividad*** | *Actividad 1.1* | | | | | |
| ***Tarea*** | *Fecha Inicio* | *Recursos* | *Responsables* | *Entregables* | *Riesgos* | *Tiempo*  *(Horas)* |

Tabla 11: Actividades y tareas

Ilustración 12: Descripción actividades y tareas

*Es importante tratar de profundizar lo máximo que se pueda en este punto, y realizar el plan de actividades con el mayor detalle posible, esto puede facilitar mucho el trabajo en las siguientes secciones, 5.2.2. Cronograma y 5.2.3 Asignación de Recursos.*

### Cronograma

*Las actividades y tareas identificadas en la sección 5.2.1 se deben graficar y organizar haciendo uso de un diagrama de GANTT o red de actividades PERT, en el cual se plasmen minuciosamente los tiempos asignados para desarrollar cada tarea, el flujo de las tareas (predecesores y antecesores) y la ruta crítica del proyecto.*

### Asignación De Recursos

*De nuevo tomando como base las tareas resultantes en la sección 5.2.1, se debe asociar a cada una de ellas los recursos necesarios para desarrollarla exitosamente. En este punto se debe tener en cuenta tanto los recursos materiales como los humanos, que en este caso son principalmente los integrantes del grupo.*

### Asignación De Presupuesto

*Con base en la información consignada en la sección 5.2.1 (actividades y tareas), 5.2.2 (tiempos) y 5.2.3 (recursos) se debe desarrollar un presupuesto para calcular el costo aproximado del proyecto.*

*Básicamente la intención de este punto es hacer un cálculo aproximado del costo que tendría el desarrollo del proyecto para el grupo de trabajo (diferente al precio de venta, haciendo uso de las principales tareas obtenidas en las sección 5.2.1 y estimando costos por miembro del equipo de recursos y otras actividades (comida, transporte), el resultado no debería estar tan lejos de la realidad.*

*La tabla 13 presenta un posible formato para realizar el cálculo del presupuesto del proyecto*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Proceso 1*** |  | | ***Cantidad del recurso*** | ***Unidad de medida*** | ***Precio por unidad*** | ***Total (pesos)*** |
| *Actividad*  *1* | *Integrante A* | *X* | *Horas/Días* | *$ zzzz* | *Cantidad\*Unidad* |
| *Integrante B* | *xx* | *Horas/Días* | *$ zzzz* | *Cantidad\*Unidad* |
| *Integrante C* | *xxx* | *Horas/Días* | *$ zzzz* | *Cantidad\*Unidad* |
|  | | | | | *TOTAL* |

Tabla 12: Presupuesto

## PLAN DE CONTROL

### Plan de Control de requerimientos

*Es bien conocido que la recolección de requerimientos es una de las actividades más complicadas durante el desarrollo de un proyecto de software, no solo se debe tener en cuenta lo que el cliente quiere, sino también las necesidades reales de todas las personas involucradas en el uso de la aplicación final.*

*Durante el proceso de desarrollo, es bastante probable que algunos requerimientos iníciales sean cambiados o incluso quitados de la propuesta inicial, en este caso el equipo de trabajo debe estar en capacidad de adaptar los documentos o incluso la aplicación para enfrentar estos cambios [2].*

*El plan de control de requerimientos debe incluir todas las herramientas que va a utilizar el grupo de trabajo, para controlar o para minimizar, en caso que se materialice, el impacto de estas variaciones y además evitar que estos cambios afecten el cronograma inicial. Algunas técnicas de control de requerimientos podrían incluir*

* *Comunicación con el Cliente*
* *Utilizar Prototipos*
* *Seguimiento Cronograma*

En esta seccion donde se debe desarrollar como se quiere controlar principalmente el cambio de los requerimientos por parte del cliente, se puede decir que en este caso unico el cliente no cambiara de parecer y no habran cambios en los requerimientos definidos inicialmente para la iniciazion del proyecto.

Esto es lo que tiene que ver con la relación del cliente; sin embargo en cuanto al desarrollo, en caso que un miembro del equipo quiera realizar cualquier tipo de cambio a un requerimiento, debe tener en cuenta cierto tipo de tareas a realizar.

**Ilustración : Manejo de Reportes**

Primero, el miembro(s) que considere(n) necesario el cambio, creacion u omision de algun requerimiento definido debe informar de forma escrita, por parte de un reporte, a la totalidad del equipo el por que es necesario el cambio.

El responsible de aceptar o no el(los) cambio(s) sera el analista de requeriemientos, de acerdo con determinados criterios que indiquen la viabilidad del requerimiento y otros aspectos a tener en cuente.

Adicionalmente se debera reprogramar el cronograma con el cambio de este requerimiento y todo lo que esto implica.

Por ultimo se debe realizar un reporte en el cual se le informe al cliente los cambios desarrollados.

### Plan de Control de cronograma

*Las actividades planeadas para el desarrollo de todo el proyecto, pueden haber sido mal estimadas (más o menos tiempo del realmente necesario), atrasando, o en general modificando el calendario inicialmente establecido en la sección 5.2.2; algunas formas de mantener un record de las tareas cumplidas con respecto a las tareas propuestas son:*

Ilustración 13: Control de cronograma

*Una vez se ha determinado un atraso en el cronograma es necesario, al igual que en la sección 5.3.1, establecer planes de recuperación del tiempo perdido, como reasignación de recursos, cambio de actividades o asignación de nuevas tareas a miembros del equipo de trabajo [3].*

El equipo de desarrollo AlimNova® utilizara el modelo de ciclo de vida llamado espiral para su planeación y desarrollo del proyecto.

El cronograma de AlimNova® esta basado en asifnacion de actividades y tareas a medida que se va avanzando en el proyecto. Por lo tanto en cada una de los hitos o entregas se y tendrá planeado hasta cierto punto para la planeación del siguiente hito.

### Planes de mitigación

Se controlaran la designacion de las tareas por medio de una tabla donde se tendra una descripcion de la tarea, el responsible y la fecha de entrega por medio de esto se podra mitigar de alguna forma el retraso de alguna tarea o actividad. Ver la siguiente tabla.

**Ilustración : Planes mitigación**

### Plan de Control de Presupuesto

*Muchas de las herramientas utilizadas para completar las diferentes actividades durante el ciclo de vida del proyecto, requieren de licencias y permisos para su utilización (compiladores, herramientas CASE, etc.) además durante todo el proceso puede ser necesario el uso de computadores extra (manejo remoto de la aplicación, recursos para las actividades del cronograma) y por otra parte están los gastos de transporte y de alimentación.*

*Principalmente por estas razones se debe contar con un plan de manejo de recursos que permita tener en cuenta todos estos gastos externos. Ejemplos de controles para el presupuesto pueden ser:*

Ilustración 14: Control de presupuesto

*Puesto que el manejo de presupuesto va de la mano con el cronograma de actividades (recursos utilizados para cada actividad), es importante realizar un seguimiento parecido al explicado en la sección 5.2.2, esto es manejo de milestones, corrección y reasignación de recursos entre otros.*

### Plan de Control de Calidad

*El plan de control de calidad es la forma en que el equipo de trabajo va a determinar la como se van a evaluar los diferentes artefactos a lo largo del ciclo de vida del proyecto. La mejor forma de determinar cómo se medirá la calidad, es dividir las actividades del cronograma en secciones y para cada sección escoger una técnica que permita realizar esta medición. La tabla 14 muestra una división de las posibles métricas usadas, según el artefacto a entregar*.

|  |  |
| --- | --- |
| **Sección** | **Tipo de medición** |
| ***Documentos*** | * *Templates (Construx)* * *Plantillas* * *Estándares (IEEE)* |
| ***Reportes*** | * *Plantilla diseñada en la sección 5.3.5 Plan de Reportes* |
| ***Codificación*** | * *Criterios de Indentación* * *Documentación de Código (Javadoc)* * *Criterio para uso de comentarios* |
| ***Procesos*** | * *Seguimiento del Cronograma* * *Utilización de Documentación* * *Generación de Informes* |

Tabla 13: Control de calidad

### Plan de Reportes

*Esta sección busca estandarizar el mecanismo de comunicación interna del grupo de trabajo, tanto para las fechas de reunión como para las fechas de trabajo individual dividido por roles y actividades. Estos formatos le permitirán al equipo ahorrar tiempo en la unión de secciones para los documentos pero principalmente establecerá un formato de interacción con las entidades exteriores (clientes y demás stakeholders).*

*Ejemplos de posibles reportes que se pueden implementar son dados en la tabla 15*

|  |  |  |  |
| --- | --- | --- | --- |
| Reportes | Factores | | |
| *Requerimientos* | 1. *Avance* | 1. *Cambios* | 1. *Problemas / Inconsistencias* |
| *Calidad* | 1. *Requerimientos* | 1. *Entregables* | |
| *Cronograma* | 1. *Estado de avance (Actual Vs Programado)* | | |
| *Codificación* | 1. *Errores encontrados* | | |

Tabla 14: Reportes

*Cada uno de los factores que se planeen tener en cuenta, deben contar con explicación de sus secciones y además debe incluir a que miembros del equipo el reporte será dirigido finalmente.*

*Ejemplo:*

*Estado del Cronograma:*

* *Secciones:*
  + *Fecha de creación del reporte.*
  + *Asistentes a la reunión.*
  + *Actividades adelantadas al cronograma.*
  + *Actividades atrasadas a lo planeado.*
  + *Estado de los recursos.*
  + *Medidas correctivas.*
  + *Dirigido a: Director de Desarrollo.*

### Plan de Recolección de Métricas

*El plan de recolección de métricas permite al equipo de trabajo realizar una constante comprobación de la calidad de los productos a entregar, planes (SPMP, SRS, SDD), documentación de código o del programa como tal, este control facilita la realización de un seguimiento a cada una de las fases de desarrollo y además permite destacar las partes o procesos atrasados o que no cumplen las especificaciones de entrega.*

*El objetivo de estas métricas es que el grupo de trabajo se trace sus propios estándares de calidad, de tal forma que la aceptación, desde un documento hasta una sección de código, tenga unas pautas que se puedan seguir y que permitan determinar si se cumple con los objetivos propuesto para cada artefacto entregable.*

*La tabla 16 presenta un ejemplo de métricas realizado para la codificación de una función.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Métrica** | **Codificación** | | | |
| ***Parámetros de medición*** | *Ideal* | *Aceptable* | *Inaceptable* | *APROBADO* |
| ***Presentación del Código*** |  | *X* |  | *S* |
| ***Clasificación del Código*** | *X* |  |  | *S* |

Tabla 15: Recolección de métricas

Ilustración 15: Parámetros de aceptación

*Las métricas que cada grupo plantee deben incluir las frecuencias de recolección, el tipo de medición (método de evaluación) para la validación y además deben ir divididas para requerimientos funcionales y no funcionales [4].*

## PLAN DE ADMINISTRACIÓN DE RIEGOS

Ilustración 16: Plan de administración de riesgos

*En el desarrollo de todo proyecto se maneja un grado de incertidumbre, es decir, existen probabilidades de que se presenten situaciones problemáticas o conflictos que puedan afectar el cumplimiento de los objetivos; por lo tanto, el proceso de administración de riesgos aparece como una oportunidad para identificar, analizar (priorizar), planear y supervisar dichas situaciones. No obstante, este proceso implica una alta inversión de tiempo y de costo, pues requiere un seguimiento contínuo de verificación y evaluación para obtener información suficiente que permita reportar con anticipación los problemas [9].*

*Por esta razón, se aconseja realizar este plan con detenimiento identificando los riesgos más representativos pues es una actividad que consumirá muchos recursos.*

*Este plan incluye:*

Ilustración 17: Plan de administración de riesgos

*Para facilitar la labor de analisis de los riesgos, se deben clasificar los riesgos generales entre las categorias de riesgos de proyecto y riesgos de producto. Además, para la priorización se deben emplear criterios como la probabilidad de que el riesgo ocurra y el efecto o impacto asociado a este. A continuación se presenta un ejemplo del manejo de estos criterios.*

|  |  |  |
| --- | --- | --- |
| Rango | Probabilidad | Efecto |
| <10% | Muy bajo | Insignificante |
| 10% - 25% | Bajo | Marginal |
| 25% - 50% | Moderado | Serio |
| 50% - 75% | Alto | Crítico |
| >75% | Muy Alto | Catastrófico |

Tabla 16: Criterios de análisis de riesgos

*Esta escala puede ser utlizada para la creación de una matriz de riesgos, con la que se podrán categorizar y determinar a cuáles se les hará seguimiento por medio de planes de mitigación y contingencia, dando prioridad a los que se encuentran en la esquina inferior derecha (riesgos intolerables) y subiendo de manera diagonal de derecha a izquierda.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **EFECTO** | | | | |
|  |  | **Insignificante** | **Marginal** | **Serio** | **Crítico** | **Catastrófico** |
| **PROBABILIDAD** | **Muy bajo** |  |  |  |  |  |
| **Bajo** |  |  |  |  |  |
| **Moderado** |  |  |  |  |  |
| **Alto** |  |  |  |  |  |
| **Muy alto** |  |  |  |  |  |

|  |  |
| --- | --- |
|  | Tolerable |
|  | Tolerancia media |
|  | Intolerable |

Tabla 17: Matriz de Riesgos

*Los planes de mitigación se realizarán con el fin reducir la probabilidad de que aparezcan y los planes de contingencia serán creados con el fin de tener acciones alternativas en caso de que los riesgos se materialicen y se conviertan en problemas.*

## PLAN DE CIERRE

Ilustración 18: Plan de cierre

*Se deben especificar todos los planes necesarios para asegurar ordenadamente la clausura del proyecto de software “Post Mortem”. Entre estos están: la reasignación del personal de trabajo, realizar el archivo de los materiales y documentos que dejo el proyecto, las actividades que se realizarán después de terminar el proyecto, reportes de lecciones aprendidas y logros obtenidos con la realización de este proyecto.*

Ilustración 19: Plan de cierre

1. PLAN DE PROCESOS TÉCNICOS

Ilustración 20: Plan de procesos técnicos

## MODELO DE CICLO DE VIDA DEL PROCESO

*Se especifica todas las actividades para cada proceso de apoyo del proyecto, para cada actividad se debe incluir los productos de trabajo, el tiempo estimado para la realización y la revisión, y el cronograma.*

*Puede escogerse un modelo de ciclo de vida que se adapte al proyecto, y a partir de éste definir todas las actividades que se deben realizar, los entregables y productos.*

*Teniendo en cuenta que se deben realizar entregas periodicas al cliente y que este manejara un comunicación directa con el director del proyecto, es aconsejable escoger como modelo de ciclo de vida, el modelo de diente de tiburon. Cada una de las entregas, se representará como un diente, y entre cada periodo de entregas se incluirá una tarea de revisión.*

AlimNova® escogió trabajar con el modelo de Espiral.



Ilustración 21: Ciclo de vida del proyecto

El modelo en espiral trabaja 4 grandes hitos, se basa en la identificación de riesgos, cumple con el desarrollo iterativo e incremental.

¿Por qué es útil para nuestro proyecto?

El proyecto que desarrollará AlimNova® está sometido a diferentes restricciones, uno de los retos es el realizar el proceso de creación de software en 4 meses, por ende es importante que el modelo permita prevenir posibles problemas, que haga del diseño una solución práctica, y que permita ver la evolución del programa en etapas de un tamaño proporcionado, es por eso que el enfoque a riesgos, la planeación continua y el desarrollo incremental hacen de este modelo un correcto seguimiento para nuestro proyecto.

SALIDA DE PROCESOS:

[1] El proceso de un arquitecto va mucho más allá de la actividad de diseño, por ello serán tareas del arquitecto.

**Trabajar con los requerimientos:** El arquitecto debe recolectar los requerimientos y entender las necesidades y la calidad apropiada de los atributos, debe ser explícito de buen entendimiento por el grupo de trabajo*. Mirar plan de Requerimientos*

**Trabajar con aplicaciones que implican a los stakeholders:** Además de verificar que se entienden las necesidades y que los atributos son claros y se alejan de la ambigüedad, se debe hacer una incorporación adecuada en el diseño, (fácil instalación, monitoreo, administración y actualizaciones del sistema o sus componentes). *. Mirar plan de Calidad*

**Trabajar con Project Management:** El arquitecto debe estar de la mano con la administración ayudando con la planeación, estimación y ubicación de las tareas, debe velar porque cada una de estas actividades se lo más clara posible para el grupo, orientar a las funcionalidades claras y seguir correctamente el diseño del sistema. *Mirar Calendarización, Red de actividades.*

La arquitectura se basa en tres ejes fundamentales de Proceso:

F1.

Miremos con detenimiento cada una de las tres fases mencionadas:

**IDENTIFICANDO ARQUITECTURA DE REQUERIMIENTOS:**

F2.

El diagrama (F2) nos muestra las entradas y salidas para determinar la arquitectura de los requerimientos.

*La arquitectura se preocupa por aspectos como rendimiento, protección, seguridad, control (basado en eventos) disponibilidad, mantenibilidad.*

Estos podrían ser algunos ejemplos sobre la arquitectura de requerimientos:

|  |  |
| --- | --- |
| Quality Attribute | Architechture Requirement |
| Performance | La aplicación debe producir en un tiempo de terminado el 90% de las respuestas. |
| Security | Todas las comunicaciones deben ser autenticadas y encriptados |
| Resource Management | Los componentes del servidor deben correr en una máquina de 512MB de memoria |
| Usability | La interfaz de usuario debe correr en un Browser, para soportar usuarios remotos |
| availability | El sistema debe tener una disponibilidad del 0,98 |
| Reliability | No pueden perderse mensajes, después de 30 segundos debe recibirse una notificación |
| Scalability | El sistema debe soportar el manejo de 500 usuarios de forma concurrente en el inicio de la partida |
| Modifiability | El sistema debe soportar migración de JAVA a .NET |

Cada uno de los requerimientos pueden tener riesgos y prioridades *Mirar Plan de Riesgos, Mirar Plan de Requerimientos.*

**DISEÑO DE LA ARQUITECTURA DE REQUERIMIENTOS:**

F3.

Las tareas del arquitecto son importantes pero deben ir muy de mano con la calidad de la arquitectura, para lo cual es necesario monitorear la calidad el diseño *Mirar Plan de Calidad.*

Muy buena documentación de requerimientos y una buena identificación de las necesidades de los stakeholders no hacen mucho si el diseño de la arquitectura es pobre.

Para introducir la figura (F3), debemos tener en cuenta que la etapa de diseño goza de dos pasos que son Iteración y naturalidad.

La primera se basa en una estrategia basada en patrones de diseño mientras la segunda especifica componentes individuales que hacen la aplicación misma.

**VALIDACION:** *(Mirar último enciso de pruebas y validación ).*

## Métodos, Herramientas y Técnicas

*Se debe dar a conocer la metodología que se usara para el desarrollo del proyecto. Se deben especificar el lenguaje de programación, las herramientas (incluyendo maquinas virtuales) y los recursos técnicos que serán usados a lo largo del proyecto, dejando claro el motivo por el cual fueron escogidos; para esto hay que tener en cuenta la experiencia de los desarrolladores, la compatibilidad, la necesidad de reducir capacitaciones y el costo de licencias.*

*Es importante definir que herramientas fueron requeridas no solo para el desarrollo del software si no tambien para la realizacion de pruebas y manejo de versiones del mismo. Otro aspecto, es describir cuales productos de trabajo serán entregables y cuales no. Finalmente es necesario que el gerente del proyecto y el líder del equipo determinen las políticas de comunicación y las reglas de trabajo. No hay que dejar de lado describir el motor de bases de datos, el administrador de bases de datos y todo lo relacionado con la administracion de la misma.*

**

Ilustración 22: Métodos, Herramientas y Técnicas

*Para todo este proceso siempre se debe tener en cuenta las necesidades arrojadas en la fase de analisis del proyecto y los recursos técnicos con los que se cuentan para el desarrollo del proyecto.*

**Identificación de actividades Infraestructura**

1. **Investigación de herramientas específicas para desarrollar los diferentes componentes**

**Los diferentes niveles de acoplamiento fueron clasificados según el tiempo que toma su investigación y desarrollo, precisamente uno de los grandes problemas en el desarrollo de Ingeniería de software descansa en el acoplamiento de los componentes, convenimos por ende nuestra propia clasificación:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | GUI | LÓGICA | RMI | PERSISTENCIA |
| GUI |  |  |  |  |
| LÓGICA |  |  |  |  |
| RMI |  |  |  |  |
| PERSISTENCIA |  |  |  |  |

**Donde:**

|  |  |
| --- | --- |
|  | Alta |
|  | Media |
|  | Baja |

Estos rangos describen las dependencias más importantes de acoplamiento para el sistema completo de software, dadas en rojo están las relaciones de mayor atención, se pretende acoplar JAVA con flash de la manera más transparente y hacer que la comunicación cliente servidor este muy bien especificada, fácilmente legible a nivel de lógica, haciendo uso de métodos RMI y buenas prácticas de programación, en orden a facilitar las demás actividades del proceso y del desarrollo de la arquitectura, con lo que nos enfocamos en dos de los requisitos fundamentales del producto de software:

Interfaz gráfica fuerte y arquitectura cliente servidor.

1. **Identificación de Dependencia en las Actividades**

**La dependencia de actividades es importante para poder realizar un buen ejercicio de calendarización y permitir el desarrollo eficiente en la programación y documentación a través de la identificación de las actividades concurrentes y las actividades dependientes para que exista cohesión en la documentación y en el mantenimiento de la arquitectura.**

**(ver red de act)**

1. **Recursos de Actividades**

**Para la Arquitectura se tendrán en cuenta asesorías para las áreas importantes de acoplamiento, como los son las comunicaciones RMI con el profesor Juan Pablo Garzón y sobre el manejo de animaciones y servicios ofrecidos por Adobe Flash .**

**Asesoria: La asesoría sobre temas técnicos, ofrecida por profesores de planta. Estás estarán restringidas por el horario de atención.**

**Uso de Salas: Para el uso de las diferentes herramientas, están a disposición las salas de la facultad de Ingeniería, donde podemos trabajar todos los componentes establecidos excepto la parte gráfica, este componente puede trabajarse en la sala del edificio Barón.**

**Uso de recursos bibliográficos: La investigación como eje fundamental del proyecto se beneficia del uso de la biblioteca de la Universidad y del material que este a disposición de los miembros del grupo.**

***Para la comunicación en RMI se utilizará bibliografía sobre Sistemas Distribuidos***

***Para el área de interfaz gráfica entraremos en el área de diseño industrial.***

**Recursos Propios: Podemos hacer uso de los recursos de los que disponga cada uno de los miembros del grupo.**

**(necesita red de act)**

**HITOS en Infraestructura**

**La entrega de documentos finales de cada actividad en la arquitectura estará enfocada en “Pensando en el Acoplamiento”.**

**Para las diferentes actividades se entregarán los siguientes documento e información:**

1. **Informe de evolución, de lo desarrollado y problemas a solucionar grupalmente.**
2. **Propuestas o sugerencias sobre la siguiente iteración.**
3. **Investigación y Herramientas:**

**Se debe presentar un informe a los miembros del grupo que explique las funcionalidades básicas de la herramienta utilizada, ¿Qué es compatible con los demás componentes?, ¿Qué no lo es? , avance sobre el conocimiento de la herramienta, inconvenientes de resolución grupales.**

1. **Identificación de Dependencia en las Actividades:**

**A nivel de rol debe presentarse un informe que corresponda al rol que desempeña cada miembro del grupo describiendo el orden en que deben ejecutarse las actividades de los procesos, haciendo énfasis en que actividades pueden realizarse en forma concurrente e identificando cuales dependen de otras actividades.**

**La propuesta esta encaminada a aportar en áreas o temas que correspondan más allá del rol que se esta ejecutando.**

1. **Recursos de Actividades**

**A nivel de rol debe presentarse un informe que corresponda al rol que desempeña cada miembro del grupo ofreciendo información que permita la capacitación o el entendimiento de los demás miembros del grupo sobre algún aspecto técnico o teórico, y ofrecer una gama de posibilidades de investigación sobre el rol asignado.**

***(La propuesta para este punto no es necesaria)***

**\*\*\*Definir ENTREGABLES\*\*pueden ser los mismos hitos**

**EN CUANTO A LAS TÉCNICAS:**

**Para el uso de las técnicas se trabajarán aspectos importantes.**

**- Diseño Técnico y diseño básico:**

**Cualquier diseño realizado debe tener una versión para presentar al cliente y una versión técnica con detalle para los miembros del grupo.**

-Revisiones cruzadas:

Se conformaran parejas para desarrollar revisiones anteriores a las presentaciones grupales.

-Desarrollo Cooperativo (XP)

Se trabajara el desarrollo de prototipos e implementaciones en parejas, se hará más fácil la comunicación y el entendimiento de las líneas de código para todos los miembros del grupo.

## Plan de Infraestructura

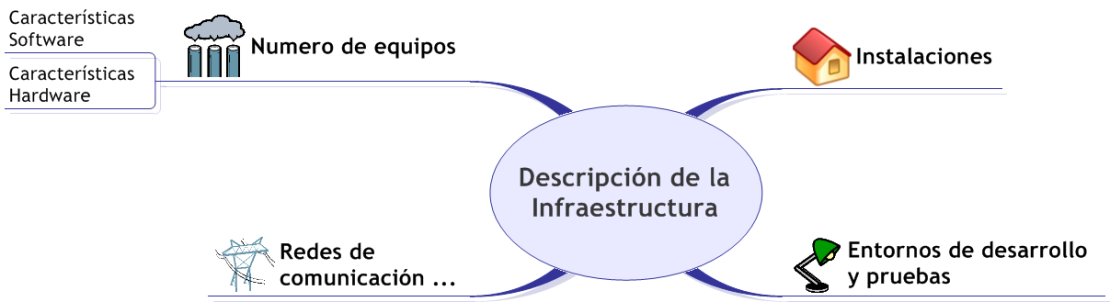
**

Ilustración 23: Plan de Infraestructura

**(SE deben investigar datos de los computadores y redes de las salas, y equipos personales)**

## Plan de Aceptación del Producto

*Aquí hacemos referencia a las actividades, metodologías, herramientas, responsables y todo aquello que sea indispensable para lograr la aceptación de los entregables de este proyecto de software por parte del cliente. Además, se han de definir los criterios de aceptación y el modo de formalizar de manera escrita el compromiso por parte del cliente en la cuál se especifique que el cliente está satisfecho con dichos entregables. Hay que tener en cuenta que los aspectos relativos a la descripción de lo que puede y lo que no puede hacer el software. Estos aspectos se aclaran con el resultado del análisis, las pruebas, inspecciones del software.*

Ilustración 24: Plan de aceptación del producto

*Las actividades pueden incluir el desarrollo, comunicación de entregables y artefactos, la sincronizacion del cronograma, y la entrega de prototipos periodicos entre otros.*

**(Este debe hacerse según plan de cierre)**

1. PLAN DE PROCESOS DE SOPORTE

Ilustración 25: Plan de procesos de soporte

*Esta sección en particular es bastante extensa, Construx ofrece plantillas completas para cada uno de estos planes ya que en proyecto de gran alcance es necesario verificar y validar cada uno de los artefactos entregados por los miembros del equipo. En esta sección se deben proponer y especificar métricas para tener la posibilidad de calificar los artefactos entregados, dichas métricas deben permitir realizar una clasificación para cada artefacto al darle un peso valorativo y de esta forma aprobarlo o de lo contrario tomar medidas como arreglar dicho artefacto o reiniciar su elaboración.*

*Para el entendimiento de estos planes es necesario tener en claro los siguientes conceptos:*

Ilustración 26: Definiciones básica del plan de procesos de soporte

## PLAN DE ADMINISTRACIÓN DE LA CONFIGURACIÓN

*La IEEE 610 define el plan de administración de la configuración así:*

*“La administración de la configuración es una disciplina que aplica estrategias técnicas, administrativas y de seguimiento para documentar las características físicas y funcionales de un ítem de configuración, además sirve para controlar, almacenar y reportar todos los cambios realizados sobre estos ítems, por otra parte, este plan también sirve para verificar que todos estos ítems estén cumpliendo los requerimientos del proyecto.”*

*Para empezar a desarrollar este plan es fundamental definir cuáles serán los documentos tipo línea base y de porcentaje de avance además se deben especificar las formas para dar versiones a cada uno de estos documentos. El método que se use para definir las versiones en los documentos puede ser definido por el mismo equipo de trabajo, pero es importante que este formato especifique el avance del ítem que se esté realizando. A continuación se muestra un ejemplo de una posible numeración de versiones para los artefactos de configuración:*

Ilustración 27: Numeración de versiones

*En cuanto al almacenamiento de las versiones de estos artefactos de configuración Construx [15] aconseja definir claramente donde y quien podrá guardar estos documentos o de lo contrario utilizar un sistema para trabajo concurrente que permita a todo el equipo de trabajo modificar un mismo artefacto de configuración al mismo tiempo. Para este fin existen varias herramientas conocidas bajo el nombre de CVS (Concurrent Versions System por sus siglas en ingles), algunas de estas son Tortoise (disponible en [13] y el CVS que trae por defecto Eclipse en su última versión [14]*

*En este plan se debe definir cómo realizar el control de cambios. Para la estandarización del control de cambios se debe especificar formatos con el fin que los miembros del equipo de trabajo, realicen las peticiones de cambio, estas peticiones tienen como objetivo llevar un historial de cómo ha sido afectado un artefacto en particular [1]. A continuación se presenta una tabla que sirve como modelo para formalizar una petición de cambio sobre un artefacto de configuración:*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fecha** | **Nombre del solicitante del cambio** | | | **Cambio solicitado** | | | **Razón de petición** | | |
|
| *Fecha en que se solicita el cambio* | *Nombre completo de la persona que solicita permiso para el cambio.* | | | *Descripción del cambio que se desea hacer. En este campo se especifica cuál será el artefacto afectado por el cambio.* | | | *El solicitante debe diligenciar porque desea realizar cambios sobre el artefacto, de esta forma el administrador de configuración puede decidir si aceptar o no los cambios.* | | |
| **Persona responsable de la petición** | | **Cantidad de cambio.** | | | | **Grado de impacto del cambio** | | | |
| **Alto** | **Medio** | | **Bajo** | **Alto** | | **Moderado** | **Insignificante** |
| *Persona encargada de la comunicación entre el solicitante del cambio y el administrador de configuración, en general este campo solo se diligencia si la estructura organizacional es demasiado estricta en sus líneas de comunicación.* | | *Para diligenciar las casillas de “Cantidad de cambio” y “Grado de impacto del cambio” es necesario que el equipo de trabajo liderado por el administrador de configuración definan cuales serán las métricas para calificar los cambios que se desean realizar sobre los documentos. Es importante recalcar que la clasificación realizada en “Alto Medio y Bajo” o “Alto Moderado Insignificante” es solo un ejemplo, el equipo puede definir la escala de sus métricas como guste. Esta sección de la tabla es diligenciada por la persona que desea realizar el cambio.*  *La sección de cantidad de cambio se diligencia de acuerdo a que tan grande o pequeña será la parte modificada del artefacto, y por otra parte, la casilla de Grado de impacto del cambio se diligencia de acuerdo a que tantos artefactos afectara la modificación que se desea realizar.* | | | | | | | |

Tabla 18: Peticiones de cambio

*Con el fin de realizar el seguimiento adecuado a cada uno de los artefactos de configuración, y que todos los miembros del equipo conozcan claramente cuál es la última versión y que contiene dicha versión, es necesario que en caso de aceptar una petición de cambio el administrador de configuración comunique a todos los miembros del equipo cual fue el artefacto afectado y una descripción de dicho cambio. A continuación se muestra un ejemplo de una posible tabla para realizar el seguimiento de cambios a los artefactos de configuración.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Documento de Cambio* | *Nuevo nombre del documento* | *Responsables* | *Fecha de cambio* | *Razones por las que cambia el documento* | *Archivos que implican modificación* |
| *Nombre del documento modificado* | *Nuevo nombre del documento que por lo general es un incremento en la versión* | *Personas que realizaron el cambio en el documento.* | *Fecha en la que se realizo el cambio* | *Razones por las cuales el administrador de configuración acepto el cambio en el documento.* | *Este campo se diligencia con todos los archivos que están sujetos a ser modificados cuando se realice el cambio.* |

Tabla 19: Seguimiento de cambios

### Introducción

En este apartado se especificara de manera detallada la forma y los estándares que se utilizarán para el control de cambios en los artefactos de software que durante el desarrollo del proyecto irán surgiendo. Estos cambios pueden ser actualizaciones, complementos, correcciones y hasta reelaboración.

La realización de este plan promete la organización secuencial de documentos y código fuente del proyecto mediante la cual se podrá llevar un seguimiento riguroso de las modificaciones de los diferentes artefactos con el fin de identificar productos relevantes para el proyecto en su totalidad así como los hitos del proyecto.

### Administración

El encargado de administrar los procesos de configuración del proyecto será William Jiménez, Administrador de Configuraciones y Documentación a quien los demás miembros del equipo comunicaran peticiones de cambio de artefactos, especificadas según los estándares a sugerir, y mediante la discusión con todos los miembros del grupo se tomara una decisión, la cual también estará debidamente plasmada en los respectivos documentos. Además será el encargado de llevar el seguimiento de las versiones de los artefactos software y de mantener la nomenclatura que se utilizara en el registro de dichas versiones.

### Responsabilidades

A continuación se especificaran las responsabilidades del administrador de configuraciones y en general del equipo de trabajo dentro de la administración de configuraciones.

Administrador de Configuraciones y Documentación

* Proponer herramientas que faciliten la administración de configuraciones del proyecto.
* Manejar apropiadamente la(s) herramienta(s) seleccionada(s) por el grupo.
* Mantener los estándares seleccionados para el manejo de versiones durante el desarrollo del proyecto.
* Recibir iniciativas de modificación de los artefactos software.
* Comunicar modificaciones en artefactos software a todos los miembros del equipo.

Equipo de trabajo

* Acoger el formato estándar de versionamiento seleccionado de documentos y de código fuente con el fin de identificar claramente los distintos artefactos del proyecto.

Conocer los medios de acceso y herramientas de gestión de los artefactos software del proyecto.

### Clasificación de documentos

Fundamentalmente se utilizaran dos tipos de documentos en el manejo de los ítems de configuración:

Documentos tipo Línea Base

Se refiere a los documentos listos o terminados, es decir los que compondrán los diferentes entregables del proyecto. El formato será el siguiente:

Nombre\_ del\_documento\_V(x.0).formato\_de\_ archivo

x representa el numero de línea base del documento.

Documentos tipo Porcentaje de Avance

Como su nombre lo indica se refiere a los documentos no terminados pero que cuentan con algún avance el cual es expresado en la nomenclatura. El formato a utilizar será el siguiente:

Nombre\_ del\_documento\_V(x-1.n).formato\_de \_archivo

x es el numero de línea base que se quiere alcanzar y n el porcentaje de avance el cual será representado con números del 1 al 99.

### Almacenamiento de versiones

El equipo de trabajo ha optado por usar básicamente dos herramientas para la administración de configuraciones y almacenamiento de versiones, Tortoise y Google Code.

La razón por la cual se han escogido estas herramientas es por su relativa facilidad de uso y su gran utilidad en el contexto de un proyecto de esta magnitud.

**Tortoise SVN- 1.6.7:**

Es una herramienta de control de código fuente para Windows la cual se puede integrar con cualquier ambiente de desarrollo.

<http://tortoisesvn.net/>

**Google Code:**

El alojamiento de proyectos en Google Code es un servicio de alojamiento de software libre rápido, fiable y sencillo. El alojamiento de proyectos en Google Code permite:

* crear proyectos instantáneos sobre cualquier tema,
* alojar código de Subversion con un 1 gigabyte de espacio de almacenamiento y admitir alojamiento para descargas con 2 gigabytes de espacio de almacenamiento,
* consultar código fuente integrado y utilizar herramientas de revisión de código para facilitar la visualización de código, la revisión de contribuciones y el mantenimiento de una base de código de gran calidad,
* realizar un seguimiento de problemas y búsquedas wiki de proyectos sencillas, pero flexibles y potentes, que pueden adaptarse a cualquier proceso de desarrollo,
* marcar como destacados y actualizar flujos que facilitan el seguimiento de los proyectos y los desarrolladores que te interesan.

<http://code.google.com/intl/es-ES/projecthosting/>

### Control de Cambios

Para el control de cambios se acordó utilizar un formato de especificación tabular, sugerida por la plantilla de ironworks.

El responsable de administrar estas tablas de petición de cambio, emitidas por cualquier miembro del equipo, será William Jiménez, Administrador de Configuraciones. Estas tablas de petición registran en primera instancia fecha, quien solicita el cambio, cuál es el cambio solicitado y su razón de ser. También presenta una métrica definida de cuan grande es el cambio y que impacto tiene sobre el proyecto.

Ahora si la petición de cambio plasmada en la anterior tabla fue aceptada se procede a realizar el respectivo registro en una nueva tabla.

El control de cambios se realiza con el fin de llevar un orden bien definido de cada artefacto de software que el proyecto genere durante su realización.

### Actividades a realizar

La siguiente tabla presenta un resumen de las actividades a realizar e el ámbito de la administración de configuraciones teniendo en cuenta los roles mas significativos dentro de cada actividad.

|  |  |  |
| --- | --- | --- |
| Actividades de Administración de configuraciones | Realizadas por | Comentarios |
| Desarrollo del Plan de administración de configuraciones | * Germán Morales, Director de Desarrollo. * William Jiménez, Administrador de Configuraciones y Documentación |  |
| Identificar ítems de configuración | * William Jiménez, Administrador de Configuraciones y Documentación. |  |
| Realizar auditorías de documentos línea base | * William Jiménez, Administrador de Configuraciones y Documentación. |  |
| Identificación de defectos | * William Jiménez, Administrador de Configuraciones y Documentación. * David Suárez, Director de Calidad. |  |
| Lanzamiento/Entrega de documentos línea base |  |  |
| Control interno de documentos línea base | * Laura Arias, Gerente. * William Jiménez, Administrador de Configuraciones y Documentación. |  |
| Control del estado de actividades de la administración de configuraciones | * William Jiménez, Administrador de Configuraciones y Documentación. |  |

Tabla : Actividades Administración Configuraciones

## PLAN DE VERIFICACIÓN Y VALIDACIÓN

*Antes de empezar a desarrollar este plan es necesario tener en cuenta la diferencia entre estos dos términos, a continuación se muestra la definición de cada uno de estos según el diccionario enciclopédico VOX y RAE:*

* *Verificar: Comprobar o examinar la verdad de algo [RAE].*
* *Validar: Comprobación de un conjunto de datos para determinar si su valor se halla dentro de unos límites de fiabilidad [VOX].*

*Un ejemplo orientado a ingeniería de software my sencillo para entender la diferencia entre estos dos términos es:*

* *Verificar es revisar que un requerimiento haya sido implementado.*
* *Validar es revisar que dicho requerimiento cumpla con su función dado un conjunto de pruebas en donde se encuentren los datos que permite el requerimiento.*

*Según la NASA, las actividades del plan de verificación y validación deben evaluar que los elementos de software cumplan con los requerimientos establecidos con el cliente final. Nótese que la NASA hace referencia a los “elementos de software” como a todos los artefactos que se tienen que entregar para un proyecto de este tipo como por ejemplo los manuales, la documentación, las versiones Beta etc. En resumen, la guía del plan de verificación y validación de la NASA propone los siguientes cuatro consejos para estos tipos de planes:*

* *Verificar que cada elemento de software ayude a cumplir los requerimientos.*
* *Evaluar cada elemento de software antes que este sea usado como base (entrada) para otras actividades.*
* *Asegurar que las evaluaciones sobre los elementos de software sean realizadas por una persona diferente al autor del mismo.*
* *Asegurar que el esfuerzo y dedicación en la verificación y validación de cada elemento de software sea suficiente para poder aprobar dicho elemento como base para las siguientes actividades.*

*La NASA también aconseja realizar reportes sobre las pruebas realizadas al código, estas pruebas son las mismas mencionadas en el plan de aseguramiento de la calidad así que se puede realizar una referencia a la sección 7.4. En resumen, el plan de verificación y validación se ejecuta en paralelo en todo el ciclo de vida del proyecto para realizar pruebas sobre todos los documentos y entregables que se realicen, por otro lado es importante aclarar que el equipo de trabajo puede definir sus formas de verificación y validación con los métodos que gusten, es más, pueden solo hacerlo sobre el código por falta de tiempo en el cronograma.*

### Introducción

El propósito de este plan consiste en ofrecer un marco único para la verificación y validación de los artefactos software del proyecto. Esta parte del documento pretende controlar los procesos del ciclo de vida del proyecto con el fin de tener una idea clara de su avance y su calidad.

La idea consiste en identificar las actividades, mecanismos, los factores que intervienen en el proceso de V&V como los recursos a consumir, desarrollo, operación y mantenimiento de de los procesos de V&V así como estrategias para evaluar los avances del proyecto. Además se definirán entradas y salidas del sistema requeridas en concordancia con los requerimientos levantados previamente.

**Ilustración: Verificación y Validación**

### Actividades y Mecanismos

Para el proceso de verificación y validación se llevaran a cabo distintas actividades que proporcionaran información la cual dará a conocer el estado de un artefacto de acuerdo a uno o varios de los requerimientos preestablecidos.

Principalmente se realizaran pruebas del sistema evaluando su desempeño y su concordancia con los requerimientos del cliente, esto no solo con el fin de entregar el producto adecuado sino de proporcionar la mas alta calidad posible al entregable. Las revisiones de los artefactos serán frecuentes y paralelas a la realización del proyecto y serán definidas por el Director de calidad y Manejo de Riesgos en acuerdo con los demás miembros del equipo.

Los mecanismos de verificación serán definidos por el equipo, liderados por Director de calidad y manejo de riesgos quien dará a conocer las métricas de evaluación de los artefactos del producto y de esta manera proceder de manera adecuada en el uso de estos mecanismos en pro del mejoramiento de la calidad del proyecto.

Se proponen los siguientes mecanismos:

* Uso de listas de chequeo.
* Creación de escenarios de entradas y salidas de prueba para el sistema tanto de manera parcial como total.

Los mecanismos definidos pueden variar a lo largo del proyecto o se pueden agregar más.

### Factores que intervienen en la verificación y validación

**Recursos**

Hardware y Software: Para realizar las actividades y tareas de de V&V se requerirán maquinas donde se pueda correr el producto y de esta manera observar su comportamiento en cada una de sus etapas. Además se utilizará software de soporte como herramientas de medición o de manejo de meticas, paquetes que permitan el registro de datos obtenidos (Excel) entre otros.

### Entradas y Salidas Requeridas

Con ánimo de lograr cumplir con los requerimientos, principalmente funcionales, del cliente se pretende evaluar el sistema en cada una de sus etapas de desarrollo proporcionando las respectivas entradas requeridas por el sistema las cuales permitirán validar el cumplimiento de los requerimientos del cliente.

Por medio de la ejecución de esta estrategia se podrá extraer información cuantitativa (por ejemplo tiempos de respuesta del sistema) e información cualitativa (por ejemplo inconsistencias con el diseño o de requerimientos) que originaran la necesidad de estrategias de solución y en consecuencia una aproximación cada vez más cercana a lo que el cliente solicita.

El diseño de estos escenarios de simulación del sistema puede ser realizado por el Director de calidad y manejo de Riesgos junto con el Director de Desarrollo y el Arquitecto si fuera necesario. De la misma manera se deberán analizar las salidas en la finalización de cada proceso para definir si estas son realmente consistentes con la funcionalidad del sistema y estar al tanto de si los requerimientos se están implementando debidamente.

### Roles

Sin duda la persona que actuará como líder en este proceso, junto al Gerente del proyecto, será el Director de Calidad y Manejo de riesgos quien se encargara de definir

El Administrador de configuraciones deberá estar pendiente de los cambios que los procesos de V&V puedan proporcionar.

Por otra parte el Director de Desarrollo y el Arquitecto deberán estar al tanto de las especificaciones y sugerencias que el Director de Calidad y Manejo de riesgos.

## PLAN DE DOCUMENTACIÓN

### Objetivo

En esta parte del documento se pretende brindar la información, tanto al cliente como a los integrantes del grupo, acerca del tipo de documentos que se manejaran a lo largo del proyecto. También se definen los diferentes estándares y plantillas que se usaran, los encargados de la revisión y la creación de cada parte del documento y por ultimo estipular fechas tentativas para la entrega de cada uno de los documentos acá mencionados.

### Documentos a entregar

A lo largo del proceso de desarrollo, se realizara una serie de documentos los cuales facilitaran la comunicación tanto con el cliente como entre los integrantes del grupo para que de esta manera se pueda estar realizando retroalimentaciones en donde se especifiquen algunas situaciones apara tener en cuenta a lo largo del proyecto. La principal idea de estos proyectos es facilitar el entendimiento de todas los stakeholders. La tabla que se verá a continuación muestra los documentos a entregar con su respectivo estándar y los responsables de dicho documento.

|  |  |  |  |
| --- | --- | --- | --- |
| Documento | Encargado | Descripción | Estándar |
| SPMP | * Laura Arias (Gerente del proyecto). * William Jiménez (Administrador de configuraciones y documentación). * David Suarez (Director de calidad y de manejo de riesgos). | Software Proyect Managament Plans, es el documento donde se define el plan para seguir a lo largo del proyecto. | IEEE |

**Tabla :Entregables**

*Según la IEEE esta subclase del SPMP debe contener un plan de cómo se documentaran todos los entregables del proyecto de software, este plan incluye productos que se entregan a los clientes y a los mismos desarrolladores. Además se debe definir que sección de la organización se encargara de proveer información para la realización de los documentos y por supuesto dicha sección de la organización también se encargara de realizar las revisiones sobre cada documento. Por otra parte, los encargados de este plan deben generar lista de chequeos para verificar los ítems para cada documento, por ejemplo, una lista de chequeo para los documentos puede ser los mismos numerales utilizados para dicha plantilla. En resumen este plan debe contener: [3]*

Ilustración 28: Contenido plan de documentación

## PLAN DE ASEGURAMIENTO DE LA CALIDAD

*En general, el objetivo de este plan es establecer lineamientos para garantizar un producto final libre de defectos, incluyendo software y documentación. La lista de chequeo que se muestra a continuación es propuesta por Construx [15] y sirve para asegurar que los estándares de calidad establecidos para el proyecto se cumplan en cada fase del ciclo de vida del mismo:*

* + *Lista de chequeo para el plan de aseguramiento de la calidad [16].*

Ilustración 29: Lista de chequeo para el aseguramiento de la calidad

*Por lo general en un proyecto de software existen productos y entregables generales, a continuación se muestran algunos de estos dando un ejemplo de cómo una organización puede asegurar la calidad en este entregable o artefacto de configuración.*

* ***Código:*** *Se debe establecer como los desarrolladores tomaran decisiones de acuerdo a lo que se va a implementar, es decir, que algoritmo es el indicado para desarrollar una parte del programa. Una posible forma de especificar como tomar dichas decisiones e implementar el código, es realizar reuniones con el arquitecto de software para informarle como se está realizando cierta parte del programa, y así el arquitecto aprobara o no dicha decisión si esta cumple con los requerimientos y arquitectura establecida.*
* ***Documentación de código:*** *Para la documentación de código el equipo de trabajo puede realizar su propio formato de código, especificando como se documentará cada sección del mismo, pero por lo general los proyectos de software toman formatos previamente definidos como el que utiliza Java en su API; por lo general, las herramientas de desarrollo permiten documentación con el formato Javadoc [17] que es el mismo mencionado anteriormente.*
* ***Manuales:*** *Por lo general los productos de software contienen dos diferentes tipos de manuales, el manual de usuario, y el de instalación. Los manuales de usuario deben contener un escenario de ejemplo para cada funcionalidad que el software ofrezca, así que el equipo de trabajo debe especificar la forma en que se harán estos escenarios, por ejemplo, si se desean colocar imágenes de muestra, se debe definir previamente la dimensión de la imagen, así como el tamaño de las hojas y de los párrafos. Por otra parte para la realización de los manuales de instalación se deben especificar previamente para que plataformas operativas se realizara el software y así realizar pruebas para cada plataforma y documentar el proceso de instalación, en algunas ocasiones es necesario instalar aplicaciones extras para que el software funcione correctamente como por ejemplo la maquina virtual de java, librerías de C o frameworks para diferentes lenguajes, para este aplicaciones se debe detallar en el manual de instalación cual es la versión que se usara y donde conseguir la documentación asociada a este.*
* ***Documentación para el producto de software (SPMP, SRS, SDD):*** *Se debe especificar en qué forma se realizará cada uno de los diferentes documentos a entregar, por lo general esto se define haciendo referencia a alguna plantilla existente como las de la IEEE, Construx, ISTAR etc.*
* ***Plan de pruebas:*** *En este plan se deben especificar las métricas para calificar los resultados de las pruebas y tener formalizado que hacer en caso de que una prueba arroje un resultado no deseado. Por lo general siempre se aplican estos cuatro tipos de prueba sobre un producto de software [18]:* 
  + ***Pruebas de programa con datos de prueba:*** *En estas pruebas se validan y verifican secciones separadas del código, en otras palabras, cada clase por separado (si es programación orientada a objetos). Los datos con los que se desea realizar las pruebas deben ser ingresados con el fin de revisar que sucede cuando estos se encuentran en el intervalo aceptado por el método o función probada y que pasa cuando estos datos exceden el intervalo. El propósito principal es revisar en que secciones del código se debe realizar manejo de excepciones.*
  + ***Pruebas de vínculos con datos de prueba:*** *En este tipo de pruebas se empiezan a fusionar clases o módulos, su objetivo al igual que las pruebas anteriores es revisar si con la vinculación y datos dentro y fuera del intervalo la nueva fusión de clases funciona correctamente.*
  + ***Sistema completo con datos de prueba:*** *Esta prueba se realiza cuando el sistema está completo, al igual que las pruebas anteriores, los datos ingresados deben estar dentro y fuera del intervalo para revisar su comportamiento en estas situaciones. Por lo general, estas pruebas ayudan a modificar la interfaz gráfica para indicarle al usuario que ha ingresado mal un dato o que simplemente no lo ingreso.*
  + ***Pruebas de aseguramiento de requerimientos funcionales:*** *Esto es más una lista de chequeo donde se verifica que el software este cumpliendo con todos los requerimientos funcionales del software.*
  + ***Pruebas de aseguramiento de requerimientos no funcionales:*** *Las pruebas para los requerimientos no funcionales deben depender de los documentos de diseño del sistema, ya que en estas pruebas, se debe especificar por ejemplo tiempos de respuesta del sistema o el número de horas que el sistema estará activo, esto depende del tipo de requerimiento no funcional que se desee probar.*

Ilustración 30: Proceso de pruebas

El director de calidad y de manejo de riesgos será el encargado de hacer las pruebas necesarias para asegurar la calidad de todo el proyecto. Cabe resaltar que antes de que cada documento llegue a manos del director de calidad y de manejo de riesgos tendrá que pasar por una revisión exhaustiva por parte de la persona encargada de elaborar esa parte del documento, luego de esto la tarea del director de calidad y de manejo de riesgos será mirar si lo que hay cumple con el alcance del proyecto y sobre todo con los estándares establecidos para que después de su aval se pueda incluir dicha parte del documento al documento final.

Las partes de la documentación que serán puestas bajo revisión para asegurar su calidad son representadas en la siguiente ilustración.

Ilustración ¿?: Composición de la documentación del proyecto

En la ilustración anterior mostramos todas las partes que componen la documentación del proyecto, donde cada una estas será remitida a revisión de calidad bajo los parámetros mencionados antes de la ilustración. Vale la pena resaltar que el encargado de la ejecución de esta actividad es el director de calidad y de manejo de riesgos y que este proceso de revisión de la calidad se hará en cada punto crítico del proyecto.

### 7.4.1 CODIGO

ilustración ¿?: Plan de aseguramiento de calidad para el código

En la ilustración anterior se muestra cómo será la ejecución del plan de aseguramiento de la calidad para el código. Las personas que se encuentran involucradas en esta actividad son el director de desarrollo, el arquitecto y el director de calidad y de manejo de riesgos. La ejecución de este plan se iniciara luego del levantamiento de requerimientos en el hito 2.

### 7.4.2 PROCESO

Se entiende como proceso a todos los pasos que se siguen para realizar cualquier tipo de código o de documento que se encuentre listo para ser adaptado al documento final. El plan de control de dichos productos se basa en que todo proceso que sea aprobado por el director de calidad y de mantenimiento tiene que tener un reporte que será archivado para conocer los detalles de dicho proceso en caso de que se presente algún fallo en algún proyecto, el control de estos procesos también se seguirá por medio de unas plantillas lo que pretenden es estandarizar el producto que finalmente será entregado al cliente. Para mantener el estándar mencionado anteriormente vamos a seguir unas plantillas para la documentación (véase sección ¿? plantilla para la presentación de documentos) y para el código (véase sección ¿? Plantilla para la presentación de código). El director de desarrollo y de manejo de riesgos es la persona encargada de hacer que se cumplan estos estándares basándose en las plantillas mencionadas anteriormente.

### 7.4.3 MANUALES

Se entiende como un manual a aquel documento el cual describe una serie de pasos para el buen uso del proyecto de software entregado. Definimos dos tipos de manuales los cuales son: los de usuario, los cuales deberán describir de manera ilustrativa cada una de las funcionalidades del software, tratando de usar una pequeña cantidad de texto y donde las imágenes que allí se usen tengan un tamaño apropiado para el fácil entendimiento de dicho manual; y los de instalación los cuales deberán tener todas las especificaciones posibles para lograr el éxito de la instalación del producto entregado.

ilustración ¿?: composición del manual del proyecto

La ilustración anterior muestra aquellos tipos de manuales que forman parte del manual del proyecto. El de usuario va dirigido a aquella persona que finalmente va a interactuar con el software y el de instalación pretende es describir los pasos para poder instalar el software entregado, este se realiza a un nivel mas técnico que el de usuario.

### 7.4.4 PRUEBAS

Es un plan que comienza a ejecutarse en el segundo hito, aunque las métricas asociadas para este plan son complicadas de cuantificar podríamos decir que es un plan difícil de definir, por lo que podremos decir que al no tener las métricas claras para este plan el proceso de control de calidad será un poco difícil de llevar. Para llevar a cabo el proceso de calidad de las pruebas dividimos estas en cinco tipos de pruebas las cuales son:

* **Pruebas de igualdad:** El objetivo principal de este tipo de prueba es asegurar la concordancia de la aplicación con el o los requerimientos en las cuales se baso el desarrollo de dicha aplicación, de esta manera se validara la funcionalidad de la aplicación.
* **Pruebas de uso:** La meta de este tipo de prueba es mirar la usabilidad de la aplicación presentada tenga una interacción amigable entre el hombre y la maquina**.**
* **Pruebas de Pre-adaptación:** Este tipo de prueba tiene como meta verificar si cada componente cumple con todos los requisitos por los cuales dicho componente fue creado.
* **Pruebas de Post-adaptación:** Esta prueba lo que pretende es buscar si existe algún tipo de inconcordancia entre los componentes desarrollados individualmente por los miembros del equipo.
* **Pruebas de usuario:** Esta prueba se realiza para poder determinar si el proyecto cumple con la calidad y la funcionalidad por la cual el proyecto fue desarrollado.

## REVISIONES Y AUDITORIAS

*Este plan está muy ligado con el nivel de calidad que se le quiere dar al proyecto, ya que dependiendo de la exigencia de los niveles de calidad se deben contratar auditorias para diferentes secciones del producto como por ejemplo, una auditoria para documentos y otra distinta para código. Según la IEEE este plan especifica los siguientes elementos [3]:*

* *Empresas, entidades o personas encargadas de hacer las auditorias.*
* *Fechas en que se realizaran las auditorias.*
* *Auditorias para el aseguramiento de la calidad*
* *Número de personas que se encargaran de la revisión de cada entregable.*
* *Fechas en las que se revisará el avance del proyecto.*

## PLAN DE RESOLUCIÓN DE PROBLEMAS

*Especifica recursos (dinero, tiempo, personal), métodos, herramientas (software), técnicas, y procedimientos para ser usados en el reporte, análisis y priorización de los problemas que surjan a lo largo del proyecto. El plan de resolución de problemas debe definir que miembro del equipo será designado a los diferentes problemas que pueden surgir en un proyecto de software, a continuación se muestra un ejemplo de esta asignación [3]:*

* ***Director de desarrollo:*** *Será el encargado de gestionar los retrasos en la implementación mediante subcontrataciones y/o pago de horas extras.*
* ***Arquitecto:*** *Serán los encargados de verificar que la implementación realizada siga la arquitectura establecida, esto se realiza mediante pruebas que comprueben la no dependencia entre los módulos, y verificar que la comunicación entre los componentes del sistema se realice de acorto al protocolo y puertos especificados. Si las pruebas den resultados no deseados el arquitecto exigirá al equipo de trabajo las modificaciones correspondientes.*
* ***Administrador de configuración:*** *En caso de retrasos en los lanzamientos de las líneas bases, su labor, es realizar un consolidado de los artículos de avance para llegar a una posible aproximación y así no atrasar por completo al equipo de trabajo.*
* ***Gerente:*** *En el plan de revisiones se definen las fechas en que se hará el seguimiento al cronograma, por lo tanto, si se descubre un retraso el gerente deberá hacer uso de sus recursos (dinero, personal, tiempo) para evitar que dicho retraso represente una amenaza para el proyecto. Los recursos se manejan de diferente forma de acuerdo a las etapas del proyecto. A continuación se muestra un ejemplo de un plan de resolución de problemas en un retraso en la codificación, donde se clasifican las soluciones de los problemas de acuerdo a las etapas del proyecto:*
* *Etapa 1, retrasos en el inicio de la implementación: Si se evidencian retrasos en esta etapa del proyecto se integraran nuevos miembros al equipo de desarrollo ya que estos no retrasaran las labores debido a que estarían al mismo nivel de conocimiento que los demás.*
* *Etapa 2 y 3, retrasos en implementación y pruebas: En esta etapa no se contrataran nuevos desarrolladores, debido a que estos retrasaran al equipo debido a que se les tendrá que capacitar y explicar detalladamente el código implementado, así que la solución será solicitar al equipo horas extra de trabajo para recuperar el tiempo perdido.*
* ***Director de calidad:*** *En caso de encontrar defectos en los entregables del proyecto o en los procesos definidos del mismo, es obligación del director de calidad delegar tareas para satisfacer los requisitos de calidad no logrados por el equipo de trabajo. Si los defectos encontrados en los entregables del proyecto son causados por el no cumplimiento de los deberes de los miembros del equipo de trabajo, el director de calidad debe diseñar estrategias para que estos errores no se vuelvan a repetir, dichas estrategias pueden ser memorandos o sanciones laborales.*

## PLAN DE ADMINISTRACIÓN DE SUBCONTRATOS

*Esta subclase del SPMP contiene planes para administrar cualquier clase de subcontrato que contribuya al desarrollo de los productos del proyecto de software. Cuando los subcontratos no son pactados desde el principio se deben definir listas de chequeo para aprobar dichas subcontrataciones, por el contrario, si ya se pacto un subcontrato, se debe especificar un cronograma con la entidad encargada de la ejecución de este, dicho cronograma será objeto de un plan administración y seguimiento especificado en esta sección.*

*En ocasiones, las organizaciones cuentan con diferentes formatos para la celebración de los contratos, por ejemplo, diferentes términos para auditorias y para realización de programas, dichos contratos deben estar como anexos del presente documento [3].*

## PLAN DE MEJORAS DEL PROCESO

*Esta sección del SPMP debe determinar las áreas de la organización que pueden ser objeto de mejora, y por supuesto, es esta sección también se especifica cómo se realizarán estos planes de mejora. Este plan debe estar fuertemente ligado con el plan de resolución de problemas, por ejemplo, la fuente causante de un problema puede encaminar al equipo de trabajo a mejorar el proceso que genera dicho problema, la ejecución de este plan trae grandes beneficios como la reducción en el esfuerzo dedicado a la revisión de los documentos y la reutilización tanto de documentación cómo de código. Por lo general cuando el proyecto ya están en marcha, los procesos no pueden sufrir grandes cambios debido a que generarían retrasos en el cronograma, entonces, es deber del director de calidad, documentar los cambios que se le puede hacer al proceso que presenta problemas para que en la realización de un nuevo proyecto se realicen las respectivas mejoras sobre el proceso [3].*

Ilustración 31: Plan de mejoras de proceso

1. ANEXOS

*En esta sección se deben enumerar los documentos y materiales que se entregan anexos al documento del SPMP para facilitar su comprensión. Por lo tanto, se aconseja dividir dichos materiales según el medio en el que se proporcionan, incluyendo una breve descripción de cada uno.*

**2. REFERENCIAS**

1. TORRES, Miguel Eduardo, DIAZ, Luis Carlos, ALDANA Juan Carlos y. Sección Materias/Ingeniería de Software/Proyecto – Material de clase – Diapositivas, Métricas. 2009 Disponible en: <http://sophia.javeriana.edu.co/~metorres/Materias/IngSoftware/Diapostivas/ProcesoSW_Metricas.pdf>
2. Ximena Higuera Moriones, Andrés Teyes Aguilar, Nicolás Gaitán,Manuel Valencia, Jairo Ipial, Christian Lemus. Ancient Risk .Incasoft. 13 Octubre 2009.
3. http://classes.cec.wustl.edu/~cse528/SPMPReport05.pdf
4. Software Project Management Plan (SPMP) template.

Online http://users.csc.calpoly.edu/~jdalbey/205/Mgmt/SPMP.html.

*[1] Construx Software, Configuration Management CXOne Standard, Construx Software Builder, Inc, Noviembre 2002.*

*[2] NASA (National Aeronautics and Space Administration) SEI (Software Engineering Laboratory), Recommended Approach to Software Development, Revisión 3, Junio 1992.*

*[3] IEEE (Institute of Electrical and Electronics Engineers), IEEE Standard for Software Project Management Plans, IEEE-SA Standards Board, Diciembre 1998.*

*[4] ESA (European Space Agency) Board for Software Standarisation and Control (BSSC), Guide to Software Project Management, Revisión 1, Marzo 1995.*

*[5] Construx Software, Project Management CXOne Standard, Construx Software Builder, Inc, Noviembre 2002.*

*[6] Diccionario de la Real Academia Española. Disponible en: http://www.rae.es/*

*[7] Sommerville I. Ingeniería de Software. 7th ed. Romo MM. Madrid: Pearson Educación. S.A.; 2005.*

*[8] Larman C. UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. 2nd ed. Aragón DF. Madrid: Pearson Educación. S.A.; 2003.*

*[9] Bruegge B, Dutoit AH. Ingeniería de Software orientada a objetos. 1st ed. Trujano G. México: Pearson Educación; 2002.*

*[10] Página de Miguel Torres [homepage de Internet]. Bogotá. Ing. Miguel Eduardo Torres Moreno MSc. Copyright - Miguel Torres 2007. [Actualizado el 26 Feb. 2007; citado 11 Feb. 2007]. Materias - Ingeniera de Software – Plantilla SRS [aprox. 3era pantalla].Disponible en: http://sophia.javeriana.edu.co/~metorres/*

*[11] JabRef Reference Manager. Disponible en: http://jabref.sourceforge.net/*

*[12] Objetivos SMART. Disponible en: http://changingminds.org/disciplines/hr/performance\_management/smart\_objectives.htm*

*[13] Tortoise CVS, Repositorio de Archivos. Disponible en: http://www.tortoisecvs.org/*

*[14] Eclipse Herramienta IDE. Disponible en: http://www.eclipse.org/*

*[15] Construx, Software Development Best Practices. Disponible en: http://www.construx.com/*

*[16] Construx Software, Qualilty Plan CXOne CheckList, Construx Software Builder, Inc, 2002.*

*[17] JAVADOC Documentation Tool. Disponible en: http://java.sun.com/j2se/javadoc/*

*[18] Kendall KE, Kendall JE. Análisis y diseño de sistemas. 6th ed. Horan B. México: Pearson Educación. S.A.; 2005.*

*[19]* *IEEE Computer Society Style Guide – References, 2006, disponible en: http://www.computer.org/portal/site/ieeecs/menuitem.c5efb9b8ade9096b8a9ca0108bcd45f3/index.jsp?&pName=ieeecs\_level1&path=ieeecs/publications/author/style&file=refer.xml&xsl=generic.xsl&*