Sistema de Gestión de Laboratorios Clínicos 2016

Plan de Desarrollo de Software

Versión 1.0

Historial de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 21/06/2016 | 1.0 | Versión propuesta para aprobación al final de la fase de inicio. | Escalante Ustariz Eddy, Salazar Colque Lisbeth. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tabla de Contenido

1 Introducción 4

1.1 Propósito 4

1.2 Alcance 4

1.3 Resumen 5

2 Vista General del Proyecto 6

2.1 Propósito, Alcance y Objetivos 6

2.2 Suposiciones y Restricciones 6

2.3 Entregables del proyecto 6

2.4 Evolución del Plan de Desarrollo del Software 9

3 Organización del Proyecto 10

3.1 Participantes en el Proyecto 10

3.2 Interfaces Externas 10

3.3 Roles y Responsabilidades 10

4 Gestión del Proceso 11

4.1 Estimaciones del Proyecto 11

4.2 Plan del Proyecto 11

4.2.1 Plan de las Fases 11

4.2.2 Calendario del Proyecto 12

4.3 Seguimiento y Control del Proyecto 16

5 Referencias 17

# Introducción

Este Plan de Desarrollo del Software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto correspondiente al segundo parcial de la asignatura de Ingeniería del Software II de la Universidad Autónoma Gabriel Rene Moreno. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto está basado en una metodología de Rational Unified Process en la que únicamente se procederá a cumplir con las tres primeras fases que marca la metodología, constando únicamente en la tercera fase de dos iteraciones. Es importante destacar esto puesto que utilizaremos la terminología RUP en este documento. Se incluirá el detalle para las fases de Inicio y Elaboración y adicionalmente se esbozarán las fases posteriores de Construcción y Transición para dar una visión global de todo proceso.

El enfoque desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

## Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

* El jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
* **Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.**

## Alcance

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo del “Sistema de Gestión de Laboratorios Clínicos”. El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones. Para la versión 1.0 del Plan de Desarrollo del Software, nos hemos basado en la captura de requisitos por medio del stakeholder representante de la empresa para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”, el cual se utilizará para refinar este documento. Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

## Resumen

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

* Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.
* Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.
* Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.
* Planes y Guías de aplicación — proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

# Vista General del Proyecto

## Propósito, Alcance y Objetivos

El Laboratorio Clínico es el encargado de realizar los análisis necesarios para el diagnóstico y seguimiento de los pacientes y usuarios. Por ello, se considera necesario el desarrollo de un sistema de gestión de análisis que el paciente solicita, además de los resultados de los mismos.

El proyecto debe proporcionar una propuesta para el desarrollo de todos los subsistemas implicados. Estos subsistemas se pueden diferenciar en dos bloques: Gestión de Parámetros y Gestión de Análisis Clínicos.

## Suposiciones y Restricciones

Las suposiciones y restricciones respecto del sistema, y que se derivan directamente de las entrevistas con el stakeholder de la empresa son:

Adaptación a la normativa de Protección de Datos

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

## Entregables del proyecto

A continuación, se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

1. **Plan de Desarrollo del Software**

Es el presente documento.

1. **Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración (para mostrar actores externos, internos y las entidades (información) que manipulan, un Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

1. **Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

1. **Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

1. **Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

1. **Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

1. **Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

1. **Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.) .

1. **Modelo de Implementación**

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

1. **Modelo de Despliegue**

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

1. **Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

# Organización del Proyecto

## Participantes en el Proyecto

* Escalante Ustariz Eddy
* Salazar Colque Lisbeth

## Interfaces Externas

El laboratorio clínico definirá los participantes del proyecto que proporcionarán los requisitos del sistema, y entre ellos quiénes serán los encargados de evaluar los artefactos de acuerdo a cada subsistema y según el plan establecido.

El equipo de desarrollo interactuará activamente con los participantes del laboratorio clínico para especificación y validación de los artefactos generados.

## Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

|  |  |
| --- | --- |
| **Puesto** | **Responsabilidad** |
| Jefe de Proyecto | El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto. |
| Analista de Sistemas | Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos. |
| Programador | Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario |
| Ingeniero de Software | Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue. |

# Gestión del Proceso

## Estimaciones del Proyecto

El presupuesto del proyecto y los recursos involucrados se adjuntan en un documento separado.

## Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

### Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar)

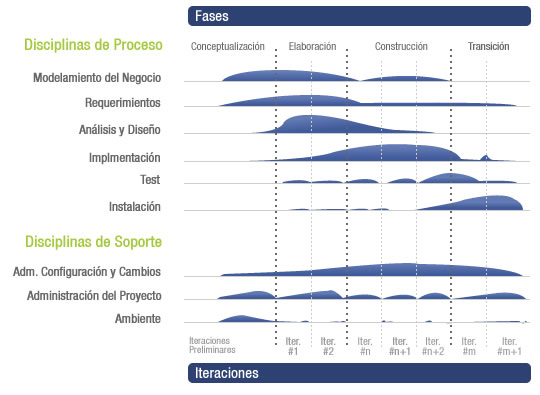
|  |  |  |
| --- | --- | --- |
| **Fase** | **Nro.**  **Iteraciones** | **Duración** |
| Fase de Inicio | 1 | 2 semanas |
| Fase de Elaboración | 1 | 2 semanas |
| Fase de Construcción | 2 | 4 semanas |
| Fase de Transición | - | - |

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

|  |  |
| --- | --- |
| **Descripción** | **Hito** |
| Fase de Inicio | En esta fase desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente / usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase. |
| Fase de Elaboración | En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana. |
| Fase de Construcción | Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 3.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta. |
| Fase de Transición | En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto. |

### Calendario del Proyecto

A continuación, se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.



Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Disciplinas / Artefactos generados o modificados**  **durante la Fase de Inicio** | **Comienzo** | **Aprobación** | | **Modelado del Negocio** |  |  | | Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio | Semana 7 | Semana 10 | | **Requisitos** |  |  | | Visión | Semana 8 | Semana 10 | | Modelo de Casos de Uso | Semana 9 | siguiente fase | | Especificación de Casos de Uso | Semana 9 | siguiente fase | | **Análisis / Diseño** |  |  | | Modelo de Análisis / Diseño | Semana 8 | siguiente fase | | Modelo de Datos | Semana 8 | siguiente fase | | **Implementación** |  |  | | Prototipos de Interfaces de Usuario | Semana 9 | siguiente fase | | **Pruebas** |  |  | | Casos de Pruebas Funcionales | Semana 9 | siguiente fase | | **Despliegue** |  |  | | Modelo de Despliegue | Semana 9 | siguiente fase | | **Gestión de Cambios y Configuración** | Durante todo el proyecto |  | | **Gestión del proyecto** |  |  | | Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones | Semana 7 | Semana 10 | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Disciplinas / Artefactos generados o modificados**  **durante la Fase de Elaboración** | **Comienzo** | **Aprobación** | | **Modelado del Negocio** |  |  | | Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio | Semana 7 | aprobado | | **Requisitos** |  |  | | Visión | Semana 8 | aprobado | | Modelo de Casos de Uso | Semana 9 | Semana 12 | | Especificación de Casos de Uso | Semana 9 | Semana 12 | | **Análisis / Diseño** |  |  | | Modelo de Análisis / Diseño | Semana 8 | Revisar en cada iteración | | Modelo de Datos | Semana 8 | Revisar en cada iteración | | **Implementación** |  |  | | Prototipos de Interfaces de Usuario | Semana 9 | Revisar en cada iteración | | **Pruebas** |  |  | | Casos de Pruebas Funcionales | Semana 9 | Revisar en cada iteración | | **Despliegue** |  |  | | Modelo de Despliegue | Semana 9 | Revisar en cada iteración | | **Gestión de Cambios y Configuración** | Durante todo el proyecto |  | | **Gestión del proyecto** |  |  | | Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones | Semana 11 | Revisar en cada iteración | |

|  |  |  |
| --- | --- | --- |
| **Disciplinas / Artefactos generados o modificados**  **durante la Fase de Construcción (Iteración 1)** | **Comienzo** | **Aprobación** |
| **Casos de Uso negociados para la Primera Release** |  |  |
| Caso de uso escogido | Semana 13 | Semana 14 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Disciplinas / Artefactos generados o modificados**  **durante la Fase de Construcción (Iteración 2)** | **Comienzo** | **Aprobación** |
| **Casos de Uso negociados para la Primera Release** |  |  |
| Caso de uso escogido | Semana 15 | Semana 16 |
|  |  |  |

## Seguimiento y Control del Proyecto

**Gestión de Requisitos**

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá una serie de atributos tales como importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

**Control de Plazos**

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto y por el Comité de Seguimiento y Control.

**Control de Calidad**

Los defectos detectados en las revisiones y formalizados también en una Solicitud de Cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist (listas de verificación) incluidas en RUP.

**Gestión de Riesgos**

A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

**Gestión de Configuración**

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá una baseline (un registro del estado de cada artefacto, estableciendo una versión), la cual podrá ser modificada sólo por una Solicitud de Cambio aprobada.

# Anexos

## Cocomo II

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tipo de Objeto | Cuenta | Simple | Medio | Avanzado | Total |
| Pantallas | 16 | 1 | 2 | 3 | 16 |
| Informes | 14 | 2 | 5 | 8 | 28 |
| Componentes 3GL | 0 | - | - | 20 | 0 |
| Total (PO) | | | | | 44 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Proporciones de Productividad | Muy Baja | Baja | Normal | Alta | Muy Ata |
| Capacidad y Madurez del Desarrollador |  |  |  |  |  |
| Capacidad y Estructuración del Entorno |  |  |  |  |  |
| Total | 7 | 10 | 12 | 25 | 50 |

PRO = 22

PO = 44

PON = PO \* [(100-%Recuperación) /100]

PON = 44 \* [(100-0) / 100]

PON = 44

E = PON/ PRO = 44/22

E = 2 Personas/Mes

Se requiere dos personas para realizar la implementación en un mes: el Jefe de Proyecto y el Analista encargado de levantar los requisitos y modelar el proyecto se contará con un equipo de 4 personas.

## Estimaciones de Costos

Estimaciones Sueldos por cada Rol

* Jefe de Proyecto 1000 $
* Analistas de Sistemas 700 $
* Programador 700 $
* Ingeniero de Software 800$

Dando un Total de 3200 $ al mes en Sueldos y Salarios

Depreciación de Equipos

* Costo de Computador #1 900 $ se deprecia al mes 18.75 $
* Costo de Computador #2 700 $ se deprecia al mes 14.58 $
* Costo de Computador #3 800 $ se deprecia al mes 16.67 $
* Costo de Computador #4 900 $ se deprecia al mes 18.75 $

Dando un Total de 70 $ al mes en Depreciación

Gastos Varios

* Internet 2 Megas 203 Bs por mes
* Factura de Luz 200 Bs por mes
* Almuerzo para 4 personas 2500 Bs al mes
* Agua de Sifón 50 Bs por mes

Total, de 2953 Bs al mes equivalente 422 $ Dólares en Gastos Varios

***Los Costos de Desarrollo al Mes dan una Suma de 3692 $ Al Mes***

## Estimaciones de Tiempo

Usando los resultados del análisis del Método de Cocomo II

Tiempo de desarrollo 1 mes

Costo por mes 3692 $

Costo Total del Software 3692 $

## Planificación de Tiempo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre de tarea | Duración | Comienzo | Fin | Predecesoras |
| Desarrollo del proyecto de la materia Ing. de Software | **23 días** | **vie 27/05/16** | **sab 18/06/16** |  |
| Perfil del Proyecto | **2 días** | **vie 27/05/16** | **sab 28/05/16** |  |
| Formular el Alcance del proyecto | 1 día | vie 27/05/16 | vie 27/05/16 | 3 |
| Definir los Objetivos Generales y Específicos | 0,5 día | sab 28/05/16 | sab 28/05/16 | 4 |
| Definir Hardware y plataforma de desarrollo | 0,5 día | sab 28/05/16 | sab 28/05/16 | 4 |
| Finalizado el Perfil del Proyecto | 0 días | sab 28/05/16 | sab 28/05/16 | 6'5 |
| Identificar Problemas | 0,5 día | dom 29/05/16 | dom 29/05/16 | 7 |
| Identificar Categorías | 0,5 día | dom 29/05/16 | dom 29/05/16 | 9 |
| Modelo de Negocios | 1 días | lun 30/05/16 | lun 30/05/16 |  |
| Construir diagrama de negocios para resultados de los análisis clínicos | 0,5 días | lun 30/05/16 | lun 30/05/16 | 12 |
| Construir diagrama de negocio para la recepción de análisis | 0,5 días | lun 30/05/16 | lun 30/05/16 | 12 |
| Finalizado el modelado de Negocio | 0,5 días | lun 30/05/16 | lun 30/05/16 | 15'14 |
| Flujo de Trabajo: Captura de Requisitos | **3 días** | **mar 31/05/16** | **vie 03/06/16** |  |
| Identificar Casos de Usos y Actores | 1 día | mar 31/05/16 | mar 31/05/16 | 16 |
| Detallar Casos de Uso | 1 día | mie 01/06/16 | mie 01/06/16 | 18 |
| Prototipo Casos de Uso | 1 día | jue 02/06/16 | jue 02/06/16 | 19 |
| Hacer el Diagrama General de Casos de Uso | 0,5 días | vie 03/06/16 | vie 03/06/16 | 19 |
| Finalizado el Flujo de Trabajo para la Captura de Requisitos | 0 días | vie 03/06/16 | vie 03/06/16 | 21 |
| Flujo de Trabajo: Análisis | **2 días** | **sab 04/06/16** | **lun 06/06/16** |  |
| Análisis de Arquitectura | 0,5 días | sab 04/06/16 | sab 04/06/16 | 22 |
| Análisis de Casos de Uso | 1 días | sab 04/06/16 | dom 05/06/16 | 22 |
| Análisis de Clases | 0,5 días | dom 05/06/16 | dom 05/06/16 | 22 |
| Análisis de Paquetes | 0,5 días | lun 06/06/16 | lun 06/06/16 | 24 |
| Finalizado el Flujo de Trabajo para el Análisis | 0 días | lun 06/06/16 | lun 06/06/16 | 24'25'26'27 |
| Flujo de Trabajo: Diseño | **2 días** | **mié 08/06/16** | **vie 10/06/16** |  |
| Construir Diagrama de Despliegue | 1 día | mié 08/06/16 | mié 08/06/16 | 28 |
| Construir Diagrama de Componentes | 0,5 días | jue 09/06/16 | jue 09/06/16 | 30 |
| Diseño lógico de la Base de Datos | 0,5 días | jue 09/06/16 | jue 09/06/16 | 28 |
| Diseño físico de la Base de Datos | 0,5 días | vie 10/06/16 | vie 10/06/16 | 32 |
| Finalizado el Flujo de Trabajo para el Diseño | 0 días | vie 10/06/16 | vie 10/06/16 | 33'31 |
| Flujo de Trabajo: Implementación | **3 días** | **sab 11/06/16** | **lun 13/06/16** |  |
| Construir diagrama de implantación | 0,5 días | sab 11/06/16 | sab 11/06/16 | 34 |
| Construcción del Código Fuente | 2 días | sab 11/06/16 | lun 13/06/16 | 34 |
| Fin de la Implementación | 0 días | lun 13/06/16 | lun 13/06/16 | 37 |
| Prueba | **5 días** | **mar 14/06/16** | **sab 18/06/16** |  |
| Definir modelo de pruebas | 2 días | mar 14/06/16 | mie 14/06/16 | 38 |
| Implementar casos de pruebas | 3 días | jue 15/06/16 | sab 18/06/16 | 40 |
| Fin | 0 días | sab 18/06/16 | sab 18/06/16 | 41 |