**UNIVERSIDAD AUTONOMA GABRIEL**

**RENÉ MORENO**

**FACULTAD DE INGENIERIA EN CIENCIAS DE LA COMPUTACION Y TELECOMUNICACIONES**

**CARRERA: INGENIERIA EN SISTEMAS**



**Plan de Administración de Proyecto de Software (PADS)**

**Materia:**

* **Ingenieria de software II**

**Docente:**

* **Ing. Rolando Antonio Martinez Canedo**

**Estudiante:**

* **UNIV. Yaure Alvarez Wilfredo…………213021692**

Santa Cruz –Bolivia

septiembre – 2019

Contenido

[**1.** **PLAN DE ADMINISTRACION EL PROYECTO** 3](#_Toc19198102)

[**1.1.** **Introducción** 3](#_Toc19198103)

[**1.2.** **Propósito del plan de proyecto** 4](#_Toc19198104)

[**2.** **Alcance** 4](#_Toc19198105)

[**3.** **Factores de calidad** 4](#_Toc19198106)

[**3.1.** **Software funcional** 4](#_Toc19198107)

[**3.2.** **Software confiable** 4](#_Toc19198108)

[**3.3.** **Software de eficiencia** 5](#_Toc19198109)

[**3.4.** **Utilidad del software** 5](#_Toc19198110)

[**3.5.** **Mantenibilidad del software** 5](#_Toc19198111)

[**3.6.** **Facilidad de pruebas del software** 5](#_Toc19198112)

[**3.7.** **Portabilidad del software** 5](#_Toc19198113)

[**4.** **Estimaciones** 6](#_Toc19198115)

[**4.1.** **Tamaño** 6](#_Toc19198116)

[**4.2.** **Complejidad** 6](#_Toc19198117)

[**4.3.** **Estructuración del cliente** 7](#_Toc19198118)

[**4.4.** **Estimaciones de kilo líneas de códigos (KLD)** 7](#_Toc19198119)

[**4.5. Modelo constructivo de costos (COCOMO II)** 8](#_Toc19198120)

[**5.** **Planificación temporal** 9](#_Toc19198121)

[**5.1.** **Identificar actividades** 9](#_Toc19198122)

[**5.2.** **Asignación de tiempo** 10](#_Toc19198123)

[**5.3.** **Diagrama de Gantt** 12](#_Toc19198124)

[**6.** **Análisis de riesgo** 13](#_Toc19198125)

[**7.** **Tabla de recursos** 15](#_Toc19198126)

[**8.** **Organización interna** 16](#_Toc19198127)

[**9.** **Mecanismos de seguimiento de control** 17](#_Toc19198128)

# **PLAN DE ADMINISTRACION EL PROYECTO**

## **Introducción**

Los avances tecnológicos de la última década se han introducido prácticamente a todos los ámbitos de la vida cotidiana, especialmente de lo que se refiere a las nuevas formas de aprendizaje, de comunicación y de manejo de información de cualquier tipo.

Cuando dos o más estudiante hablan mediante celular quieren darse información a veces no tienen a la mano dicha información o pueden dar información errónea solo con el fin de no quedar mal con la otra persona.

Las aplicaciones web o móviles han ayudado mucho en esta parte o problemática, brindando información correcta, concreta y con una gran disponibilidad, información que se la puede adquirir a cualquier hora desde cualquier lugar. Por ejemplo, cuando queremos saber en qué dirección se encuentra un lugar, buscamos en la aplicación de mapas y nos dirige a donde ir, no importando la hora ni el lugar, asi como estas hay muchas otras aplicaciones que nos ayudan a comunicarnos.

El presente trabajo va orientado al tema de brindar información acerca de las características animales que existen en el área protegida del valle de tucavaca, para los turistas que en alguna ocasión desean visitar dicha área protegida.

## **Propósito del plan de proyecto**

El propósito de la planificación del proyecto de software es proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables de recursos, costos y planificación temporal, lo cual se consigue mediante un proceso de descubrimiento de la información que lleve a estimaciones racionales, y proveer una guía de desarrollo para todo el personal encargado de su implementación.

# **Alcance**

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo de nuevas herramientas para nuestra sociedad como ser “Software de reconocimiento de animales, usando inteligencia artificial”. 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 de diagramas estructurados se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones.

# **Factores de calidad**

## **Software funcional**

El software cumple con los requerimientos funcionales expresados por las necesidades de los usuarios de manera que cumpla con la seguridad e integridad, protegiendo los datos las de entrada como como de salida.

## **Software confiable**

El software cuenta con variables globales de control para evitar fallas por errores del sistema y respalda los datos del usuario manejando una base de datos en la nube.

## **Software de eficiencia**

El software es capaz de hacer un buen uso de los recursos disponibles del hardware relacionado con el software para brindar buenos tiempos de respuestas con respecto a las funcionalidades.

## **Utilidad del software**

El software está relacionado con términos y conceptos del ámbito de la información y comunicación académica entendible para los usuarios, tanto así que cuenta con interfaces intuitivas y comprensibles que facilitan la interacción con la aplicación web.

## **Mantenibilidad del software**

El software estará alojado en un servidor el cual tendrá la posibilidad de ser evaluado por los usuarios y así mejorar el software ofreciendo actualizaciones y hacer su respectivo mantenimiento

## **Facilidad de pruebas del software**

El software cuenta con una arquitectura que define la estructura basada en componentes que nos facilitara aplicar estrategia de prueba para la detección de errores.

## **Portabilidad del software**

El software contara con la facilidad de ser migrado entre diferentes versiones de sistema operativo, es 100% portable, realizado en plataforma web.



# **Estimaciones**

## **Tamaño**

El tamaño del software es una medida de la información procesada y proporcionada por el sistema, que se puede expresar en: puntos de función (PF), líneas de código (LDC), número y tipos de requerimientos. Podemos considerar como un proyecto mediano debido que la cantidad de funciones a realizar no es masiva.

## **Complejidad**

La complejidad se puede definir en relación con las interacciones entre componentes y subsistemas del sistema, y por otra, con la variedad de cada uno de los subsistemas. Entendemos por variedad, el número de estados posibles que puede alcanzar un sistema o un componente.

El factor técnico de complejidad toma en cuenta la medida de varias técnicas y otros implicados en el desarrollo y en el implemento de la información procesada requerida.

Los factores técnicos de complejidad se determinan, estimando el grado de influencia de algunos componentes “características generales de aplicación”. El grado de influencia en la escala recorre de cero (no presente o no influenciada) hasta 5 (influencia fuerte) según la métrica orientada a la función. Para determinar la complejidad del proyecto se debe tomar en cuenta lo siguiente:

La complejidad es relativa a la experiencia en proyectos anteriores.

Existen medidas sobre la complejidad de proyectos basadas en el diseño y código (métricas). En la fase de estimación no son aplicables porque no hay ni diseño ni código, por eso hay que utilizar medidas más subjetivas como, por ejemplo: los PF.

## **Estructuración del cliente**

Para determinar la estructuración del cliente se deben tomar en cuenta los siguientes puntos:

El grado de seguridad que tiene el cliente sobre sus requerimientos.

El grado de incertidumbre de los requisitos que se han definido.

La facilidad con la que se pueden dividir las funciones.

La naturaleza jerárquica de la información a procesar.

Cuando el grado de incertidumbre estructural resulta ser bajo se puede descomponer mejor el producto y realizar una mejor estimación.

## **Estimaciones de kilo líneas de códigos (KLD)**

El propósito de este método es poder estimar cuantas KLDC podría tener un software que recién vamos a desarrollar.

1. **VE =** 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Proyecto | KLDC, Tiempo y Costo | | | |
| **Optimista** | **Más Probable** | **Pesimista** | **Esperado** |
| KLDC | 350,483 | 450,043 | 1041,159 | **531.969** |
| Tiempo | 4 | 5 | 8 | **5,33** |
| Costo | 30000 | 32000 | 56000 | **35666,67** |

**KLDC VE = = 571,802** KLDC

**Tiempo VE = = 5.33** Meses

**Costo VE =**  **= 35666,67 $**

## **4.5. Modelo constructivo de costos (COCOMO II)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tipo de Objeto** | **Cta.** | **Factor de peso** | | | **Total** |
| **Simple** | **Medio** | **Avanzado** |
| **Pantalla** | 14 | **1** | **2** | **3** | 28 |
| **Reportes** | 15 | **2** | **5** | **8** | 75 |
| **Componente 3GL** | 12 | **-** | **-** | **10** | 120 |
| **P.O.** | **223** |

**PoN =  = )**

**PoN =156,10**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Proporciones de productividad | Muy baja | Baja | Normal | Alta | Muy alta |
| Capacidad y experiencia del desarrollador |  |  |  |  |  |
| Capacidad y madurez del entorno |  |  |  |  |  |
| PROD | 4 | **7** | **13** | 25 | 50 |
| 20  **PROD**. |

**E =  =**

**E =**7,805

Analizando el resultado del esfuerzo que nos dice que el proyecto concluiría en un mes con 7 personas y representado en horas seria 1800.

Concluyendo COCOMO II estima un resultado razonable de acuerdo al trabajo del equipo.

# **Planificación temporal**

Todo proyecto requiere una planificación del tiempo a emplear en las diversas actividades que se van a llevar a cabo para el cumplimiento del mismo, a través de 2 diagramas se pretende mostrar la distribución de tiempos planificada, primeramente, el diagrama de Gantt, a través del cual se podrá apreciar el tiempo que se le va a otorgar para la realización de cada actividad y las actividades que son requisitos para realizar otras actividades.

Mientras que en el diagrama PERT se podrá apreciar las relaciones de cada actividad una con otra y así mismo se podrá ver la ruta critica el proyecto, es decir aquellas actividades que un retraso en las mismas ocasionaría un retraso en todas las demás actividades y por lo tanto un retraso en el proyecto en sí.

## **Identificar actividades**

**1ra Fase de Inicio**

Requisitos del software

A 1.1 Encontrar actores y casos de uso

A 1.2 Priorizar casos de uso

A 1.3 Detallar un caso de uso

A 1.4 Prototipar la interfaz de usuario

A 1.5 Estructurar el modelo de casos de uso

**2da Fase de Elaboración**

Análisis del software

A 2.1 Realizar análisis de la arquitectura

A 2.2 Analizar casos de uso

A 2.3 Analizar clases

A 2.4 Analizar paquetes

**3ra Fase de construcción**

Diseño del software

A 3.1 Realizar el diseño de la arquitectura

A 3.2 Diseñar casos de uso

A 3.3 Diseñar clases

**4ta Implementación del software**

A 4.1 Realizar la implementación de la arquitectura

A 4.2 Integrar el sistema

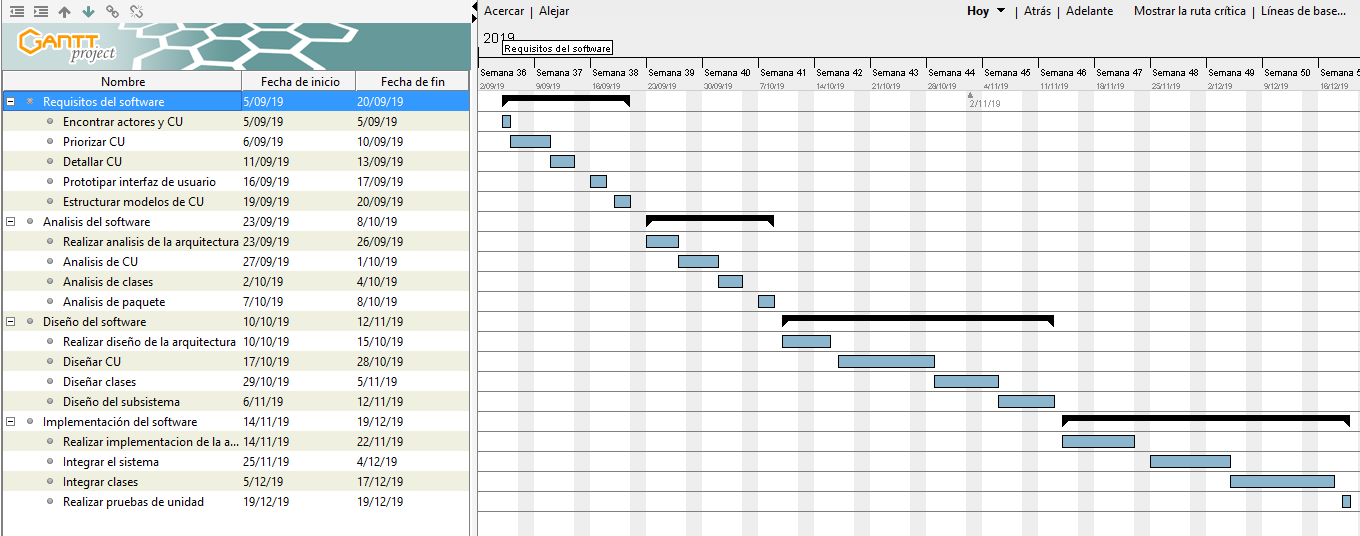
A 4.3 Integrar clases

A 4.4 Realizar pruebas de unidad

## **Asignación de tiempo**

|  |  |  |
| --- | --- | --- |
| **Actividad** | **Después de** | **Duración de la Actividad** |
| A 1 Requisitos del Software | - | 11 días |
| A 1.1 Encontrar Actores y Casos de Uso | - | 1 días |
| A 1.2 Priorizar Casos de Uso | A 1.1 | 3 días |
| A 1.3 Detallar Casos de Uso | A 1.2 | 3 días |
| A 1.4 Prototipar la Interfaz de Usuario | A 1.3 | 2 días |
| A 1.5 Estructurar el Modelo de Casos de Uso | A 1.4 | 2 días |
| A 2 Análisis del Software | A 1 | 13 días |
| A 2.1 Realizar el Análisis de la Arquitectura | A 1 | 4 días |
| A 2.2 Analizar Casos de Uso | A 2.1 | 3 días |
| A 2.3 Análisis de Clases | A 2.3 | 4 días |
| A 2.4 Análisis de Paquetes | A 2.4 | 2 días |
| A 3 Diseño del Software | A 2 | 23 días |
| A 3.1 Diseño de la Arquitectura | A 2 | 4 días |
| A 3.2 Diseñar Casos de Uso | A 3.1 | 8 días |
| A 3.3 Diseñar Clases | A 3.2 | 6 días |
| A 3.4 Diseño de Subsistema | A 3.3 | 5 días |
| A 4 Implementación del software | A 3 | 33 días |
| A 4.1 Realizar la Implementación de la Arquitectura | A 3 | 7 días |
| A 4.2 Integrar el Sistema | A 4.1 | 8 días |
| A 4.3 Implementar un subsistema | A 4.2 | 8 días |
| A 4.3 Implementar Clases | A 4.2 | 9 días |
| A 4.2 Realizar pruebas de Unidad | A 4.3 | 1 días |

## **Diagrama de Gantt**



# **Análisis de riesgo**

|  |  |  |
| --- | --- | --- |
| **Riesgo** | **Probabilidad (%)** | **Impacto** |
| **R1. Al programador le tomara tiempo dominar el lenguaje.** | 20 | Significativo |
| **R2. Incumplimiento con la fecha de entrega** | 20 | Critico |
| **R3. Incumplimiento con las fechas intermedias** | 20 | Crítico |
| **R4. No se cumplirá con el horario asignado a los programadores** | 30 | Significativo |
| **R5. Fallas en el hardware o software** | 40 | Significativo |
| **R6. Mala planificación en el diseño** | 40 | Significativo |
| **R7. Abandono de personal** | 25 | Significativo |
| **R8. Cambios de requisitos** | 25 | Significativo |
| **R9. Problemas de comunicación en el equipo.** | 20 | Significativo |

|  |  |  |
| --- | --- | --- |
| **RIESGO** | **Plan de aversión → Reducir** | |
| **Probabilidad** | **Impacto** |
| R1. Al programador le tomara tiempo dominar el lenguaje. | Se debe capacitar y actualizar al personal en diferentes lenguajes. | Buscar un desarrollador para que lo capacite en un tiempo corto. |
| R2. Incumplimiento con la fecha de entrega | Se deberá evaluar más a detalle el proyecto. | Evaluar el proyecto a mitad del desarrollo, para ver si contratar más personal. |
| R3. Incumplimiento con las fechas intermedias | Planificar las fechas holgadas. | Trabajar más tiempo de lo programado. |
| R4. No se cumplirá con el horario asignado a los programadores | Establecer horarios fijos de acuerdo a la disponibilidad de tiempo de los programadores. | Trabajar fines de semana y horas extras. |
| R5. El cliente no estará conforme con el software | Hacer el respectivo mantenimiento preventivo a los equipos periódicamente | Obtener equipos y/o acceso informático con garantía |
| R6. Mala planificación en el diseño | Realizar una buena captura de requisitos. Utilizar un ciclo de vida flexible. | Volver a una etapa anterior. |
| R7. Abandono de personal | Buen trato a los programadores, brindar buen ambiente e incentivar por objetivos logrados. | Elegir estándares de codificación. |
| R8. Cambio de requisitos | Realizar una etapa de requerimientos minusiosa sobre las funciones que tiene que realizar el software e investigar detalladamente como lo hacían antes | Informar al cliente de que los cambios que solicita afectan en el tiempo, costo y esfuerzo del proyecto. |
| R9. Problemas de comunicación en el equipo. | Realizar actividades de trabajo en equipo donde participen de manera colaborativa. | Confraternizar los fines de semana con el equipo. |

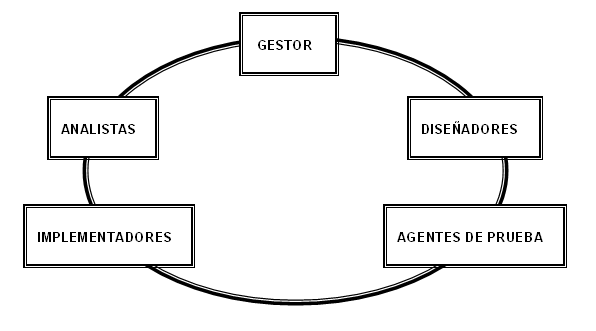
# **Tabla de recursos**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Recurso** | **Fechas** | | **Cantidad** | **Costo** | **%** | **Costo Unit** | **Costo** |
| **Desde** | **Hasta** |  | **Unitario ($us)** | **Depre** | **Neto** | **Total ($us)** |
| **Hardware** |  |  |  |  |  |  |  |
| PC | 05/09/2019 | 19/12/2019 | 5 | 615 | 25 | 490 | 2450 |
| CELULAR HUAWEI Y6 | 05/09/2019 | 19/12/2019 | 1 | 100 | 30 | 66 | 66 |
| ROUTER INALAMBRICO | 05/09/2019 | 19/12/2019 | 1 | 170 | 30 | 20 | 20 |
|  |  |  |  |  |  |  | 0 |
| **Software** |  |  |  |  |  |  | 0 |
| SO Windows 10 HOME (64 bits) | 05/09/2019 | 19/12/2019 | 5 | 184 | 30 | 34 | 170 |
| PHP Laravel | 05/09/2019 | 19/12/2019 | 5 | 0 | 0 | 0 | 0 |
| Herramienta CASE (Enterprice Architec Business & Software Engineering Edition) | 05/09/2019 | 20/12/2019 | 5 | 599 | 30 | 449 | 2245 |
| GanttProject | 05/09/2019 | 21/12/2019 | 5 | 0 | 0 | 0 | 0 |
| Flutter | 05/09/2019 | 22/12/2019 | 5 | 0 | 0 | 0 | 0 |
|  |  |  |  |  |  |  |  |
| **Gente** |  |  |  |  |  |  |  |
| Gestor de proyecto | 05/09/2019 | 19/12/2019 | 1 | 1500 |  | 1500 | 1500 |
| Encargado de control de calidad | 05/09/2019 | 19/12/2019 | 1 | 700 |  | 700 | 700 |
| Analistas y diseñadores | 05/09/2019 | 19/12/2019 | 1 | 900 |  | 900 | 900 |
| Programador | 05/09/2019 | 19/12/2019 | 2 | 350 |  | 350 | 700 |
|  |  |  |  |  |  |  |  |
| **Logística** |  |  |  |  |  |  |  |
| Material de escritorio | 05/09/2019 | 19/12/2019 |  | 25 | 25 | -100 | 0 |
| Muebles y Enseres | 05/09/2019 | 19/12/2019 |  | 799 | 20 | 699 | 0 |
| Material de Limpieza | 05/09/2019 | 19/12/2019 |  | 10 |  | 10 | 0 |
| Refrigerio | 05/09/2019 | 19/12/2019 | 10 | 25 | 0 | 25 | 250 |
|  |  |  |  |  |  |  |  |
| **Infraestructura** |  |  |  |  |  |  |  |
| Local | 05/09/2019 | 19/12/2019 | 1 | 400 |  | 400 | 400 |
| Energía Eléctrica | 05/09/2019 | 19/12/2019 | 1 | 30 |  | 30 | 30 |
| Agua | 05/09/2019 | 19/12/2019 | 1 | 20 |  | 20 | 20 |
| Internet | 05/09/2019 | 19/12/2019 | 1 | 50 |  | 50 | 50 |
|  |  |  |  |  |  |  | 9501 |

# **Organización interna**

La estructura de equipo que utilizaremos para el desarrollo del producto será la de Descentralizado Democrático (DD), ya que es la más conveniente para el equipo de trabajo.

El equipo constará de 1 personas con cargos que tendrán que adaptarse a cada área de trabajo.

****

# **Mecanismos de seguimiento de control**

El seguimiento y control de un proyecto se lo realiza para asegurar que el equipo de desarrollo cumple con el Plan de Proyecto, esto se realiza con el fin de medir costo, tiempo y performance del proyecto.

Entre las tareas a realiza se encuentran:

Seguir y revisar los resultados y logros del proyecto

Revisar el Plan de Proyecto para reflejar los resultados obtenidos y ajustar las tareas restantes en caso de ser necesario

Analizar el progreso en la ejecución del Plan.

Tomar acciones correctivas en caso de desvíos.

Fijar nuevas metas.

El seguimiento y control se lo pretende realizar de la siguiente manera:

Realizar reuniones periódicas del estado del proyecto en las que todos los miembros del equipo presentan un informe de los progresos y de los problemas.

Evaluar los resultados de todas las revisiones realizadas a lo largo del proceso de ingeniería de software.

Determinar si se han conseguido los hitos formales del proyecto en la fecha programada, para ello se deben definir primeramente los objetivos que se esperan conseguir al llegar a cada uno de los hitos

Comparar la fecha real de inicio con las previstas para cada tarea del proyecto.

Reuniones informales con los profesionales del software para obtener su valoración subjetiva del progreso hasta la fecha y los problemas que se avecinan