

ÁREA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

9° CUATRIMESTRE

DESARROLLO DE APLICACIONES WEB

ADMINISTRACIÓN DE PROYECTOS DE TI II

INTEGRADORA I

ACTIVIDAD 2:

“ESTRUCTURACIÓN DE UN PROYECTO DE TI”

GRUPO: ITI-901

PRESENTA:

|  |  |  |
| --- | --- | --- |
| NOMBRE | MATRÍCULA | CORREO |
| Angel Isai Albarrán González | 17000034 | angel.isai.ag.98@gmail.com |
| Juan Jesús Delgado López | 17000097 | juan.jesus.99jjdl@gmail.com |
| Alfonso Mondragón Montes | 17000033 | mondragon\_alfonso@outlook.com |
| Erika Vega Valdes | 17000028 | erika99vega@gmail.com |

PROFESOR: M.I.S. RODOLFO MARTÍNEZ PUENTE

GENERACIÓN: 2019 – 2021

ACÁMBARO, GUANAJUATO. 28 DE MAYO DEL 2020

**CONTENIDO**

[**INTRODUCCIÓN** 4](#_Toc41602339)

[**JUSTIFICACIÓN DEL PROYECTO** 5](#_Toc41602340)

[**Descripción de la problemática** 5](#_Toc41602341)

[**Objetivo** 5](#_Toc41602342)

[**Alcance** 7](#_Toc41602343)

[**Usuarios de la aplicación** 7](#_Toc41602344)

[**Mercado meta** 8](#_Toc41602345)

[**Patrones de diseño** 8](#_Toc41602346)

[**Script del proceso de administración** 10](#_Toc41602347)

[**Estándares de calidad aplicables al proyecto de TI** 11](#_Toc41602348)

[**Ciclo de vida** 15](#_Toc41602349)

[**PLANEACIÓN DEL PROYECTO** 20](#_Toc41602350)

[**Carta de inicio del proyecto** 20](#_Toc41602351)

[**Minuta de reunión de arranque** 20](#_Toc41602352)

[**Organigramas** 20](#_Toc41602353)

[**Definición del Alcance** 21](#_Toc41602354)

[**Elementos clave para su planeación, monitoreo y control** 21](#_Toc41602355)

[**Perfiles de los participantes y del administrador** 21](#_Toc41602356)

[**Matriz de responsabilidades** 23](#_Toc41602357)

[**Lista de factores clave de desempeño** 23](#_Toc41602358)

[**Información clave para cada actor involucrado** 24](#_Toc41602359)

[**Métodos de comunicación, justificación y formato** 24](#_Toc41602360)

[**Diagrama de Gantt** 25](#_Toc41602361)

[**Diagrama de ruta crítica** 25](#_Toc41602362)

[**ESTRUCTURA DEL PROYECTO** 27](#_Toc41602363)

[**REFLEXIÓN** 28](#_Toc41602364)

[**BIBLIOGRAFÍA** 29](#_Toc41602365)

[**DIAGRAMA DE ACTIVIDADES** 31](#_Toc41602366)

# **INTRODUCCIÓN**

La importancia de la gestión y la importancia de administración de un proyecto en TI es una aplicación para la gestión de proyectos de la aplicación de técnicas y sistemas modernos de la gestión tal como lo dice Jerry “La aplicación de técnicas y sistemas modernos de gestión para ejecutar un proyecto de principio a fin, y para alcanzar los objetivos predeterminados de alcance, calidad, tiempo y costo.” (Anderson, 2018)

La administración de proyectos de TI ayuda a completar con las ambiciones que se tiene acerca de la empresa y su realización de proyectos tal como lo dice el director general de bienestar préstamos y ahorro Francisco “La Administración de Proyectos es fundamental para cumplir la misión de una organización, ya que, a través de ella, se detonan los procesos de cambio con una visión estratégica de acoplamiento, de conjunto y sinergia entre las diferentes áreas que integran dicha organización”. (Perezanta Ordoñez, 2014)

Tomando en consideración todo lo anterior y viendo como es que los grandes ejecutivos piensan de la administración de proyectos, lo corrobora Ben ya que nos dice que, sin una buena administración, todo el proyecto es una farsa, haciendo énfasis en el apartado de los presupuestos diciendo “La verdad es que ejecutar proyectos sin una buena gestión de proyectos es una economía falsa. A menudo se piensa que es una carga innecesaria para el presupuesto, y no hay duda de que puede ser costoso, hasta un 20% del presupuesto general del proyecto.” (Aston, 2019)

Viendo que cada uno de ellos lleva un control sobre los proyectos toma a consideración que el mal uso de una administración puede llevar a la falla de este, tal como nos dice el administrar de la página Kenos ” Problemas normales que pueden causar que un proyecto fracase, los factores que pueden afectar negativamente el éxito de un proyecto de TI incluyen avances en la tecnología durante la ejecución del proyecto, cambios a la infraestructura que impactan la seguridad y la gestión de datos”. (Kenos, 2018)

# **JUSTIFICACIÓN DEL PROYECTO**

## **Descripción de la problemática**

Dentro del Hotel Posada Virrey de Mendoza se detectó como una insuficiencia el control y administración de registros para la reservación de habitaciones dentro de este, además del sumando de servicios que se hayan ocupado durante el hospedaje del cliente dentro de las instalaciones.

Actualmente el proceso de registros se realiza manualmente es decir a puño y letra, se llenan unos pequeños registros de entrada al hotel y después se pasan a una hoja de reservaciones en el cual se tiene el registro del nombre del cliente, día de entrada, el servicio de baño, costo por noche, día de salida y por último la firma del encargado en turno; los encargados van desde los recepcionistas hasta los mismos dueños dependiendo el turno.

Tener este tipo de procedimiento es muy complicado tanto de ejecutarlo y de tener un orden ya que, es necesario llenar los problemas que se tienen si una hoja de registro se llegará a extraviar. Esto perjudica de manera grave a la empresa ya que se perdería tiempo realizando el proceso de nuevo y también, trae problemas con los clientes ya que se perdería confianza y fidelidad ya que, una empresa con poca responsabilidad no brinda un buen servicio.

## **Objetivo**

Lograr una aplicación web que ayude al Hotel Posada Virrey de Mendoza, en la cual se puedan registrar servicios, control de empleados y habitaciones, utilizando una aplicación en un lapso de 3 meses.

**Objetivos específicos**

* Agilizar el método de registro en el hotel una vez terminada la base de datos.
* Aumentar la administración de los registros un 50%.
* Aumentar la administración de los servicios en un 50%.

**Objetivos metodológicos**

* Planear las actividades para la elaboración de la aplicación web, tomando en cuenta el tiempo restante para su entrega.
* Recolectar información de la empresa o negocio para la estructura de la aplicación.
* Realizar una investigación y recopilar la información para la esquematización de la base de datos.
* Analizar la información recopilada para definir los atributos de las tablas.
* Realizar un diagrama ER con los datos recolectados e integrar la cardinalidad de cada una de las tablas.
* Codificar con el diagrama la base de datos en MySQL para ver el funcionamiento y detección de errores.
* Realizar pruebas e inserciones en la base de datos, antes de su implementación para identificar fallas.
* Realizar los ajustes necesarios a la base de datos en caso de ser necesarios, para su buen funcionamiento.
* Codificar la aplicación web.
* Contratar los servicios digitales para el alojamiento de la aplicación.
* Implementar la aplicación web en la institución.

## **Alcance**

Dentro de la empresa se desea desarrollar una aplicación web, la cual va tener control a través de dicha aplicación, esto en un plazo de 3 meses a partir de que se halla identificado todo lo que compondrá la aplicación, además de tener varios módulos encargados de las operaciones del hotel, las cuales constan de eliminar, agregar y editar algún registro realizado en alguno de los módulos de la aplicación, dicho punto solo si se desarrolla correctamente en tiempo y forma la aplicación web, esto contando o incluyendo la pequeña aplicación para su manejo.

## **Usuarios de la aplicación**

* Administrador (cuanta con todos los permisos hacia la aplicación).
* SUsuario (Casi como el administrador solo con alguna limitación para editar en la aplicación).
* Usuario (Uso básico de las acciones de la aplicación).

|  |  |
| --- | --- |
| **Tipo usuario** | **Características a manejar** |
| **Administrador** | Conocimiento completo de la aplicación web, uso completo del back end, conocimiento de gestión y administración para los demás usuarios. |
| **SUsuario** | Uso medido de la aplicación web, manejo básico de una PC, control del back end de la aplicación (acciones de la aplicación), manejo de algunos errores que se puedan producir para su corrección. |
| **Usuario** | Manejo básico de un equipo de cómputo, conocimiento de cómo acceder a la aplicación web, conocimiento de las acciones básicas que realiza la aplicación. |

## **Mercado meta**

Para el mercado meta se tiene como objetivo a empresas pequeñas enfocadas a dar un servicio a clientes sobre la renta de una estancia. Estas empresas con no más de 2 sucursales y estén enfocas en dicho giro, contar con el servicio de internet en sus instalaciones, tener por lo menos 2 quipos de cómputo, para dicho enfoque del mercado meta fue más a las empresas pequeñas y no tanto hacia las personas, pues el servicio que se ofrece va enfocado a las acciones que se realizan en los hoteles, albergues, etc.

## **Patrones de diseño**

* **Composite:** Este útil patrón permite crear y manejar estructuras de objetos en forma de árbol, en las que un objeto puede contener a otro(s). En este punto cabe aclarar que las estructuras de este tipo se componen de nodos (un objeto que a su vez contiene otros objetos) y Hojas (objetos que no contienen otros), y que ambos comparten una misma Interface que define métodos que deben implementar. (Medina Medina F. J., informaticapc.com, 2003)

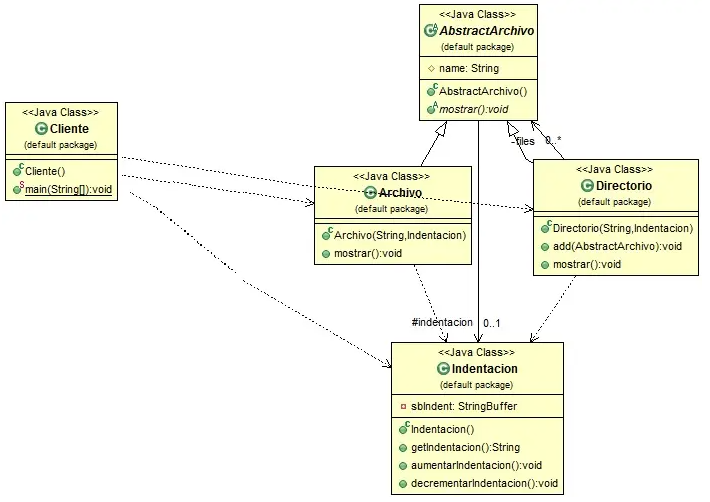


Figura . Ejemplo de Composite

* **Bridge:** Este patrón de diseño permite desacoplar una abstracción de su implementación, de manera que ambas puedan variar de forma independiente. En dichos casos nos resultaría útil el patrón Bridge (puente) ya que 'desacopla una abstracción' (un método abstracto) al permitir indicar (durante la ejecución del programa) a una clase qué 'implementación' del mismo debe utilizar (qué acciones ha de realizar). (Medina Medina F. J., informaticapc.com, 2003)

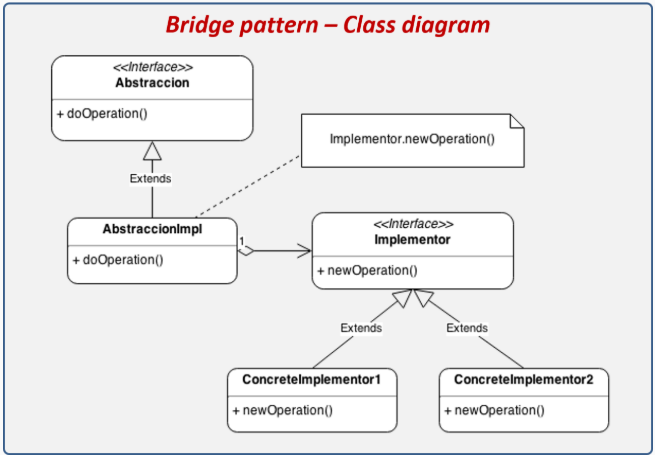


Figura . Ejemplo de Bridge

* **Adapter:** Este patrón permite que trabajen juntas clases con interfaces incompatibles. Para ello, un objeto adaptador reenvía al otro objeto los datos que recibe (a través de los métodos que implementa, definidos en una clase abstracta o interface) tras manipularlos en caso necesario. (Medina Medina F. J., 2003)

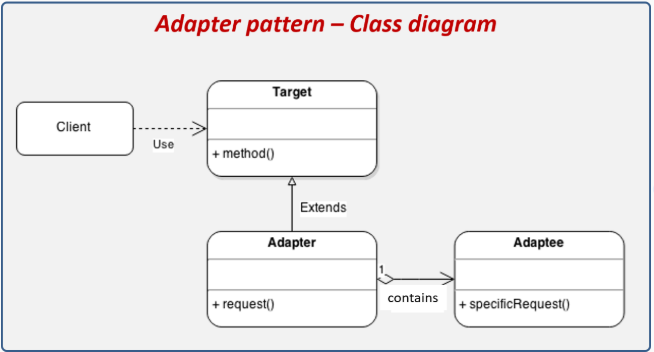


Figura . Ejejemplo de Adapter

* **Singleton:** Utilizaremos el patrón Singleton cuando por alguna razón necesitemos que exista sólo una instancia (un objeto) de una determinada Clase. Dicha clase se creará de forma que tenga una propiedad estática y un constructor privado, así como un método público estático que será el encargado de crear la instancia (cuando no exista) y guardar una referencia a la misma en la propiedad estática (devolviendo ésta). (Medina Medina F. J., informaticapc.com, 2003)

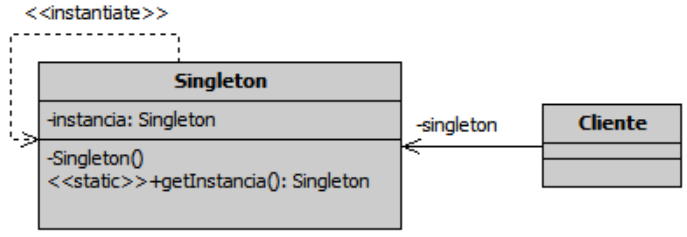


Figura . Ejemplo de Singleton

Se usará singleton este por como esta empleada la empresa, además el uso de MVC también está contemplado para la realización de la aplicación web, para ello se tomó en consideración algunos aspectos de algunos otros para ver los posibles resultados si es que no se usara este, por así decirlo con un Sdapter o bridge en uso de la programación de la aplicación, siendo el Singleton por el que se decantó más en el equipo, por la manera en el que se actúa y funciona el patrón de diseño, además de intentar ingresar algún otro si fuera necesario para la codificación de la aplicación.

Para los protocolos que usaremos el IEEE 830 para la especificación de los requerimientos del proyecto, además del ISO 2700 esto tratando de que cuando se contraten los servicios de host implementar el HTTPS, adjunto a esto, con la verificación de seguridad en el código de la aplicación, para las pruebas del IEEE 829 está la documentación de errores y como es que fueron corregidos, siendo estos los que tenemos presentes.

## **Script del proceso de administración**

[Vinculo al Script del proyecto](Entregables/Script-Administracion-de-Proyectos.docx)

## **Estándares de calidad aplicables al proyecto de TI**

**CMMI**

Modelo de Capacidad y Madurez o CMM (Capability Maturity Model), es un modelo de evaluación de los procesos de una organización. Fue desarrollado inicialmente para los procesos relativos al software por la Universidad Carnegie-Mellon para el SEI (Software Engineering Institute).

Este modelo establece un conjunto de prácticas o procesos clave agrupados en Áreas Clave de Proceso (KPA – Key Process Area). Para cada área de proceso define un conjunto de buenas prácticas que habrán de ser:

* Definidas en un procedimiento documentado.
* Provistas (la organización) de los medios y formación necesarios.
* Ejecutadas de un modo sistemático, universal y uniforme (institucionalizadas).
* Medidas.
* Verificadas. (misaelutec, 2020)

En la siguiente figura se muestran los niveles de madurez por etapas.



Figura 5. Niveles de CMMI. (Tutorialspoint, 2020)

La forma en la que este modelo se implemente es para mejorar los procesos del proyecto, en este caso seria para la aplicación web, de manera que con los niveles de madurez mejoraremos el software.

**ISO 9001**

La ISO 9001 es una norma ISO internacional elaborada por la Organización Internacional para la Estandarización (ISO) que se aplica a los Sistemas de Gestión de Calidad de organizaciones públicas y privadas, independientemente de su tamaño o actividad empresarial. Se trata de un método de trabajo excelente para la mejora de la calidad de los productos y servicios, así como de la satisfacción del cliente. (ISOtools, 2020)

El sistema de gestión de calidad se basa en la norma ISO 9001, las empresas se interesan por obtener esta certificación para garantizar a sus clientes la mejora de sus productos o servicios y estos a su vez prefieren empresas comprometidas con la calidad. Por lo tanto, las normas como la ISO 9001 se convierten en una ventaja competitiva para las organizaciones. (ISOtools, 2020)

Esta norma es de las mas importantes, ya que nuestro producto tiene que tener calidad, de manera de cumpla con las expectativas del cliente, esta norma nos enfoca en realizar una aplicación mejor que las anteriores.

**IEEE 830**

El estándar IEEE 830-1998 está enfocado en recomendaciones prácticas para la especificación de requerimientos, fue desarrollado por la IEEE y la IEEE-SA (Standards Association), indica la estructura y organización de toda la información que debe incluirse en un buen documento de especificación de requerimientos de software. Los objetivos que tiene este estándar son: ayudar a los clientes a describir con precisión lo que quieren en el software y a las personas encargadas de recibir esta información establecer una estructura estándar (definir el formato y contenido de las especificaciones de requerimientos de software y manual del mismo) para la especificación de requerimientos de software (ERS) en sus organizaciones. (Huerta, Ramírez Cruz, & Sánchez, 2016)

Este estándar lo hemos manejado desde siempre, en todos los proyectos los requerimientos son indispensables, básicamente aplicamos este estándar al definir cada uno de los requerimientos solicitados por el cliente.

**IEEE 829**

Estándar 829 para la documentación de prueba de software y sistema, proporciona un conjunto estandarizado de documentos para las pruebas de software. Los procesos de prueba determinan si los productos de desarrollo cumplen con los requisitos de la actividad y si es satisfactorio para las necesidades del usuario. Las tareas del proceso de prueba se especifican para diferentes niveles de integridad. Los elementos de documentación para cada tipo de documento de prueba pueden ser seleccionados. El alcance de las pruebas abarca sistemas basados en software, hardware y sus interfaces. (Rivas, 2015)

El estándar cubre el desarrollo y el uso de documentos de prueba de software. Define el formato, el contenido y el uso de dicha documentación. La norma describe un conjunto de documentos de prueba básicos que están asociados con los aspectos dinámicos de las pruebas (es decir, la ejecución de procedimientos y código). La norma define el propósito, esquema y concurso de cada documento básico. La norma no requiere metodologías de prueba específicas, enfoques, técnicas, instalaciones o herramientas, y no especifica la documentación de su uso. (Rivas, 2015)

Los documentos son:

* Plan de pruebas master (MTP).
* Plan de pruebas de nivel (LTP).
* Diseño de prueba de nivel (LDT).
* Caso de prueba de nivel (LTC).
* Procedimiento de prueba de nivel (tipo básico).
* Nivel de riesgo de prueba (LTL).
* Informe anomalía (AR).
* Nivel intermedio estado de prueba informe (LITSR).
* Informe de prueba de nivel (LTR).
* Informe de prueba master (MTR). (Rivas, 2015)

Este estándar se maneja en todos los documentos solicitados, el realizar un buen formato con datos legibles y fácil de interpretar para los involucrados.

**ISO 27000**

ISO/IEC 27000 es parte de una familia en crecimiento de estándares sobre Sistemas de Gestión de la Seguridad de la Información (SGSI) de ISO/IEC, el [ISO 27000 series](https://es.wikipedia.org/wiki/ISO_27000_series). ISO/IEC 27000 es un grupo de estándares internacionales titulados: Tecnología de la Información - Técnicas de Seguridad - Sistemas de Gestión de la Seguridad de la Información - Visión de conjunto y vocabulario. Tiene como fin ayudar a organizaciones de todo tipo y tamaño a implementar y operar un Sistema de Gestión de la Seguridad de la Información (SGSI). (ISO, 2018)

La norma ISO/IEC 27000 fue preparada por el Comité Técnico conjunto [ISO/IEC JTC 1](https://es.wikipedia.org/wiki/ISO/IEC_JTC_1) Tecnología de la Información, SC 27 Técnicas de Seguridad. (ISO, 2018)

ISO/IEC 27000 proporciona:

* Una visión general de normas sobre Sistemas de Gestión de la Seguridad de la Información (SGSI).
* Una introducción a los Sistemas de Gestión de la Seguridad de la Información (SGSI).
* Una breve descripción del proceso para Planificar - Hacer - Verificar - Actuar (Plan - Do - Check - Act, PDCA).
* Los términos y las definiciones utilizadas en la familia de normas Sistemas de Gestión de la Seguridad de la Información (SGSI).

Esta norma internacional es aplicable a todo tipo de organizaciones desde empresas comerciales hasta organizaciones sin ánimo de lucro. (ISO, 2018)

La ISO 27000 es de las mas importantes en proyecto de software, esta norma se ve reflejada en la seguridad con la que contará el software, desde el login hasta la seguridad en la base de datos.

## **Ciclo de vida**

**Modelo Secuencial Lineal**

El modelo secuencial lineal sugiere un enfoque sistemático, secuencial, para el desarrollo del software que comienza en un nivel de sistemas y progresa con el análisis, diseño, codificación, pruebas y mantenimiento.(Pressman, 2002)

**Características:**

* Es el más utilizado.
* Es una visión del proceso de desarrollo de software como una sucesión de etapas que produce productos intermedios.
* Si se cambia el orden de las fases, el producto final será de inferior calidad.

**Ventajas:**

* Permite una construcción rápida del sistema.
* Es útil para sistemas de un tamaño muy reducido, que no requiera más de dos o tres programadores y que no requiera un mantenimiento posterior.
* No pierde tiempo en las etapas de planificación, documentación, control de calidad, etc.
* Cualquiera, sin preparación técnica, lo puede utilizar.

**Desventajas:**

* Los errores de análisis y diseño son costosos de eliminar, y se propagan a las fases siguientes con un efecto conocido como bola de nieve.

**Modelo Espiral**

Es un modelo de proceso de software evolutivo que conjuga la naturaleza iterativa de construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial. Proporciona el potencial para el desarrollo rápido de versiones incrementales del software. (Pressman, 2002)

**Características:**

* Contiene una nueva etapa que es el análisis de riesgos, no incluida anteriormente.
* Este modelo es el indicado para desarrollar software con diferentes versiones actualizadas como se hace con los programas modernos de PC´s.
* La ingeniería puede desarrollarse a través del ciclo de vida clásico o el de construcción de prototipos.

**Ventajas:**

* No requiere una definición completa de los requerimientos del software a desarrollar para comenzar su funcionalidad.
* En la terminación de un producto desde el final de la primera iteración es muy factible aprobar los requisitos.
* Sufrir retrasos corre un riesgo menor, porque se comprueban los conflictos presentados tempranamente y existe la forma de poder corregirlos a tiempo.

**Desventajas:**

* Resulta difícil convencer a grandes clientes de que el enfoque evolutivo es controlable.
* Debido a su elevada complejidad no se aconseja utilizarlo en pequeños sistemas.
* Genera mucho tiempo en el desarrollo del sistema.
* Modelo costoso.
* Requiere experiencia en la identificación de riesgos.

Modelo Incremental

El modelo incremental combina elementos del modelo lineal secuencial (aplicados repetidamente) con la filosofía interactiva de construcción de prototipos. El modelo incremental aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software. (Pressman, 2002)

**Características**:

* Los incrementos son pequeños.
* Permite una fácil administración de las tareas en cada iteración.
* La inversión se materializa a corto plazo.
* Es un modelo propicio a cambios o modificaciones.
* Se adapta a las necesidades que surjan.

**Ventajas:**

* Mediante este modelo se genera software operativo de forma rápida y en etapas tempranas del ciclo de vida del software.
* Es un modelo más flexible, por lo que se reduce el coste en el cambio de alcance y requisitos.
* Es más fácil probar y depurar en una iteración más pequeña.

**Desventajas:**

* Requiere de mucha planeación, tanto administrativa como técnica.
* Requiere de metas claras para conocer el estado del proyecto.
* Orientado a ciertos tipos de clientes-usuario
* Cuando el cliente quiere entrega rápida o parciales.

El modelo que se usara en este proyecto es el espiral, puesto que sus fases se adaptan muy bien al proyecto propuesto, a continuación, se describe cada fase de este ciclo de vida:

**Planificación:** Durante esta fase se definir el giro del proyecto, en que se va a enfocar en que empresa, los objetivos del proyecto y como este ayudará a la empresa. Para realizar la planeación se deberá realizar una investigación en base a una entrevista realizada al cliente, de esta forma se busca que requerimientos necesita el software y que datos se necesitaran. Una parte fundamental para el proyecto es realizar la planeación, para esto usaremos un diagrama de Gantt para una mejor administración de los recursos, quien realizara cada actividad y analizar la ruta que tomara el proyecto durante su elaboración.

**Análisis de riesgo:** Durante esta fase se definirá el diseño de la aplicación, así como también la paleta de colores que se usara, el mapa de navegación y los componentes multimedia requeridos, esta sección es importante realizarla con el menor de los fallos, ya que el editar y corregir diseños de una aplicación durante la programación, es bastante tedioso, por lo cual retrasara el proyecto. Algo importante durante esta fase es realizar algún prototipo de esta manera analizaremos con mas facilidad los detalles y errores que se pueden corregir en ese instante.

**Implementación:** Se realizará básicamente toda la programación requerida en base a los requerimientos obtenidos, la programación de la base de datos, realizar la conexión con esta, además de la unión de la parte visual con la lógica de la aplicación, durante esta fase se realizarán constantes testeos por parte del equipo, para identificar los errores o realizar mejoras antes de la entrega del software. Para la codificación se busca usar un framework de entornos web, el cual será CodeIgnater, puesto que tiene varias librerías que agilizaran esta parte.

**Evaluación:** En esta fase realiza un análisis del proyecto desarrollado, para antes de realizar otra vuelta del proyecto observar lo que se realizó en este. Se debe analizar en detalle si los riesgos detectados anteriormente ya tuvieron solución. Básicamente, en esta fase se determinará el avance del proyecto y dará pistas de hacia dónde debe enfocarse la próxima iteración.

# **PLANEACIÓN DEL PROYECTO**

## **Carta de inicio del proyecto**

[Vinculo a la Carta de Inicio](Entregables/PP_PLT_v1_Acta%20del%20proyecto.doc)

## **Minuta de reunión de arranque**

[Vinculo Minuta 1](Entregables/Minutas/PHA_v1_Minuta1.doc)

[Vinculo Minuta 2](Entregables/Minutas/PPQA_PLT_v1_Minuta2.doc)

## **Organigramas**

**Organigrama Empresarial**

**Organigrama Equipo Web Rabbit**

## **Definición del Alcance**

[Vinculo a la definición del alcance](Entregables/Declaración%20del%20alcance.docx)

## **Elementos clave para su planeación, monitoreo y control**

[Vinculo al documento de Elementos clave](Entregables/Elementos%20clave%20para%20su%20planeación.docx)

## **Perfiles de los participantes y del administrador**

|  |  |  |  |
| --- | --- | --- | --- |
| Rol | Nombre | Actividades | Descripción |
| Gerente del Proyecto | Erika Vega Valdes | * Realiza la planeación del proyecto. * Dar solución a problemas. * Controlar las entradas y salidas del presupuesto. | Es el responsable de la definición del proyecto, y la asignación de recursos, además de brindar ayuda en la definición de actividades y aprobación de las mismas. |
| Líder del Proyecto | Erika Vega Valdes | * Orientar al equipo de trabajo. * Solución de conflictos. * Llevar el control de tiempos. * Controlar los entregables. | Él se encarga y es responsable de atender las necesidades del Analista, Diseñador y todos los demás roles, brindando solución a los requerimientos.  Establece el control de los avances de proyecto. |
| Documentador | Angel Isai Albarrán González | * Diseñar y construir un repositorio de información compartido. * Agregar los nuevos documentos al repositorio. * Especificar el formato que será usado para la documentación. * Asegurarse que los documentos mantienen el mismo estándar. | Es el responsable de llevar la documentación del proyecto, con los lineamientos y los formatos bien establecidos. Mantener la información activa para todos los miembros del equipo. |
| Analista | Juan Jesús Delgado López | * Entrevista al cliente. * Realizar el análisis de los requisitos. * Analizar la estructura básica del sistema. | Es el encargado del diseño del sistema, Análisis general y detallado, documento de flujo y operación de especificaciones. |
| Diseñador | Angel Isai Albarrán González | * Generar el diseño arquitectónico y diseño detallado del sistema basándose en los requisitos. | Se encarga de realizar un concepto del sistema que ayude a cumplir con los objetivos. |
| Programador | Alfonso Mondragón Montes | * Realizar la codificación del sistema. * Hacer la documentación del código. | Definir y mantener el código fuente de uno o varios componentes, garantizando que cada componente implemente la funcionalidad correcta. |
| Especialista en Base de Datos | Alfonso Mondragón Montes | * Diseñar la base de datos. * Crear la base de datos. * Verificar y corregir errores en la base de datos. | Son responsables de manejo, mantenimiento, desempeño y la confiabilidad de la base de datos. Están a cargo del diseño de nuevos modelos del mismo. |
| Responsable de Calidad | Juan Jesús Delgado López | * Revisar los documentos de requisito de usuarios y de software. * Revisar el plan de administración de proyecto. * Revisar el plan de testeo. * Revisar la fase de diseño arquitectónico. * Revisar la fase de diseño detallado. * Revisar las políticas de control de cambios. * Revisar la documentación. | Garantizar el cumplimiento de los compromisos hechos con el proyecto desde el punto de vista del proceso a seguir. |
| Tester | Alfonso Mondragón Montes | * Es el encargado de asegurar la calidad de cada uno de los productos. * Interactuar con el diseñador. * Proveer una buena indicación de confiabilidad del software. | Garantizar que se cumplan los requerimientos funcionales establecidos.  Es el encargado de dar el visto bueno y que cuente con todo lo establecido.  Realiza pruebas. |
| Cliente | Rosa Clementina Saucedo Rosas | * Se encarga de dar aceptación al proyecto. * Brinda información para poder definir los requerimientos. | Es el responsable de establecer los requerimientos, revisarlos y autorizarlos, además de verificar el software a fin de cerrar formal el proyecto. |

## **Matriz de responsabilidades**

[Vinculo a la Matrix de responsabilidades](Entregables/Matriz%20de%20responsabilidades.xlsx)

## **Lista de factores clave de desempeño**

**Productividad:** Esto permite a los gerentes del proyecto gestionar los recursos invertidos. Es importante tener en cuenta no solo las horas de trabajo sino también las tareas desempeñadas. Para lograr un desempeño bueno durante el proyecto se tendrá que observar el porcentaje de tiempo de trabajo en las diferentes tareas todo esto para optimizar aún más el tiempo.

**Seguimiento:** Para esta parte es necesario e importante un buen gestor el cual brindará grandes herramientas para tener siempre y a la mano el control de los avances del proyecto y de los participantes. Cualquier retraso en una tarea tendrá consecuencias para las siguientes tareas y por su puesto el retraso del proyecto.

**Calidad:** En esta parte el cliente tiene un lugar muy importante ya que a él le debe de gustar y será quien dé el visto bueno al proyecto. Para lograr realizar esta parte es necesario tener en cuenta que el proyecto sea entregado en tiempo y forma estipulada, además de dar una buena presentación al cliente.

**Costes:** Para medir los costes de manera detallada es un poco difícil ya que por un lado si se puede sacar el coste de material, pero para el recurso humano es muy difícil ya que no se sabe con exactitud. Aun que un software como lo es Project brinda herramientas para hacerlo de manera más fácil y calcular con exactitud los costes de los gastos del proyecto.

**Margen de beneficios:** Todo proyecto debe tener un margen de beneficios establecidos desde el principio, que debe ser medido y controlado durante la realización del proyecto. Esto permite comprobar si nuestra estimación estaba correcta y nos ayuda a elaborar presupuestos futuros.

## **Información clave para cada actor involucrado**

[Vinculo a Información clave](Entregables/Informcaión_clave.xlsx)

## **Métodos de comunicación, justificación y formato**

Hoy en día existen muchas herramientas las cuales nos ayudan en la cumplir con la comunicación en un equipo de trabajo o proyecto. Las aplicaciones para móviles de mensajería instantánea por medio de la conexión a internet son grandes soluciones, específicamente por la situación sanitara que se vive actualmente.

Debido al distanciamiento social y a la suspensión de clases presenciales en este ciclo escolar, la tecnología nos dota de soluciones para poder mantener una comunicación en equipo de trabajo y poder realizar un buen proyecto.

Consideramos que los medios digitales como los correos electrónicos aportan a la comunicación entre el equipo de trabajo y la institución u organización para la cual se realiza dicho proyecto.

Las plataformas de video llamadas y mensajería son alternativas buenas debido a la diversidad y versatilidad en su uso. Se puede mantener una conversación en tiempo real, compartir archivos y contenido multimedia.

Aun así, la comunicación puede ser complicada y en ocasiones resulta difícil de recordar los acuerdos, por lo que los documentos como las minutas y el control de estos nos permite entablar soluciones en los que se toma en cuenta a todos los participantes que se requieran en el momento. Los oficios, memorándums o simples cartas también son una buena opción para establecer una comunicación un poco más formal entre un gran número de personas. Optamos por no usar estas herramientas debido a que los integrantes que conforman el equipo son pocos. La divulgación de información se puede presentar compartiendo el archivo como la minuta o el cronograma de actividades mediante las herramientas tecnológicas ya mencionadas anteriormente.

[Vinculo a Matriz de comunicación](Entregables/Matriz%20de%20comunicación%20Plantilla.xlsx)

## **Diagrama de Gantt**

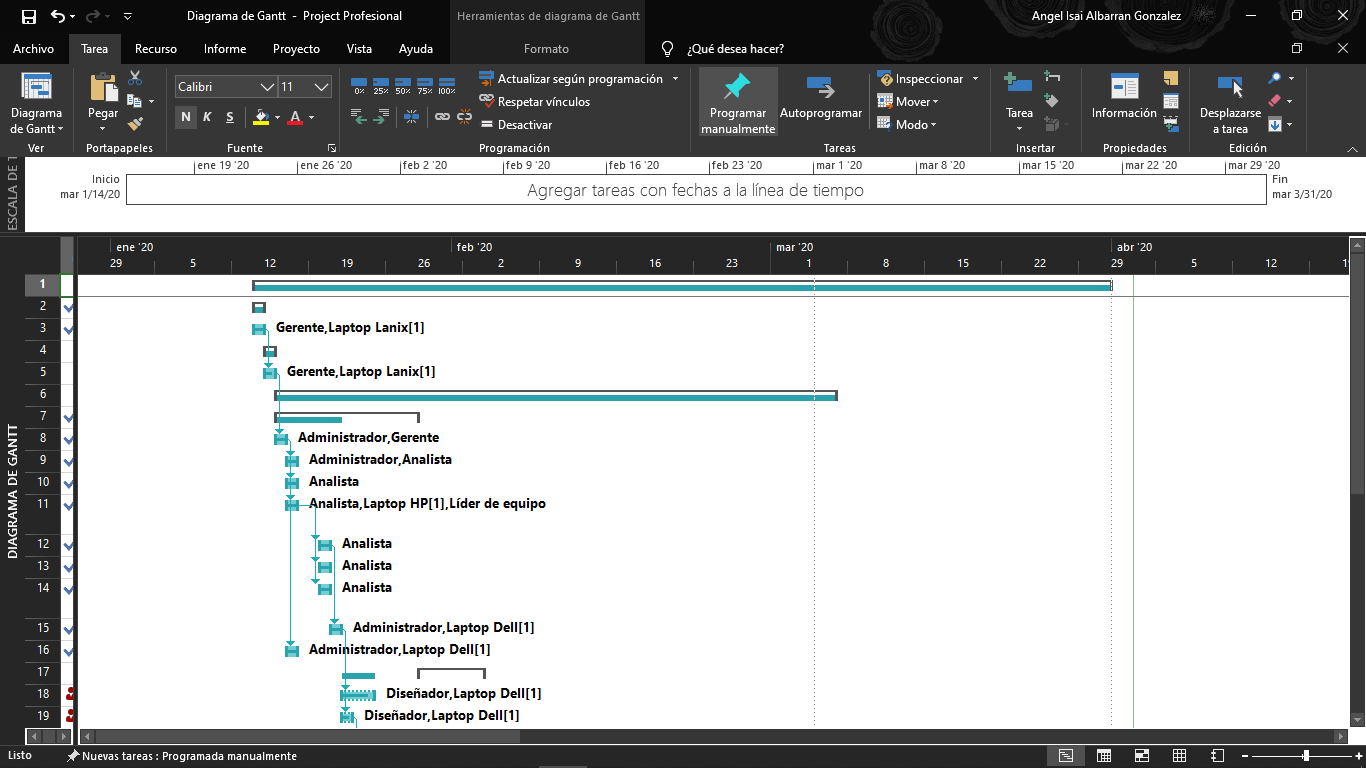


Figura . Diagrama de Gantt 1

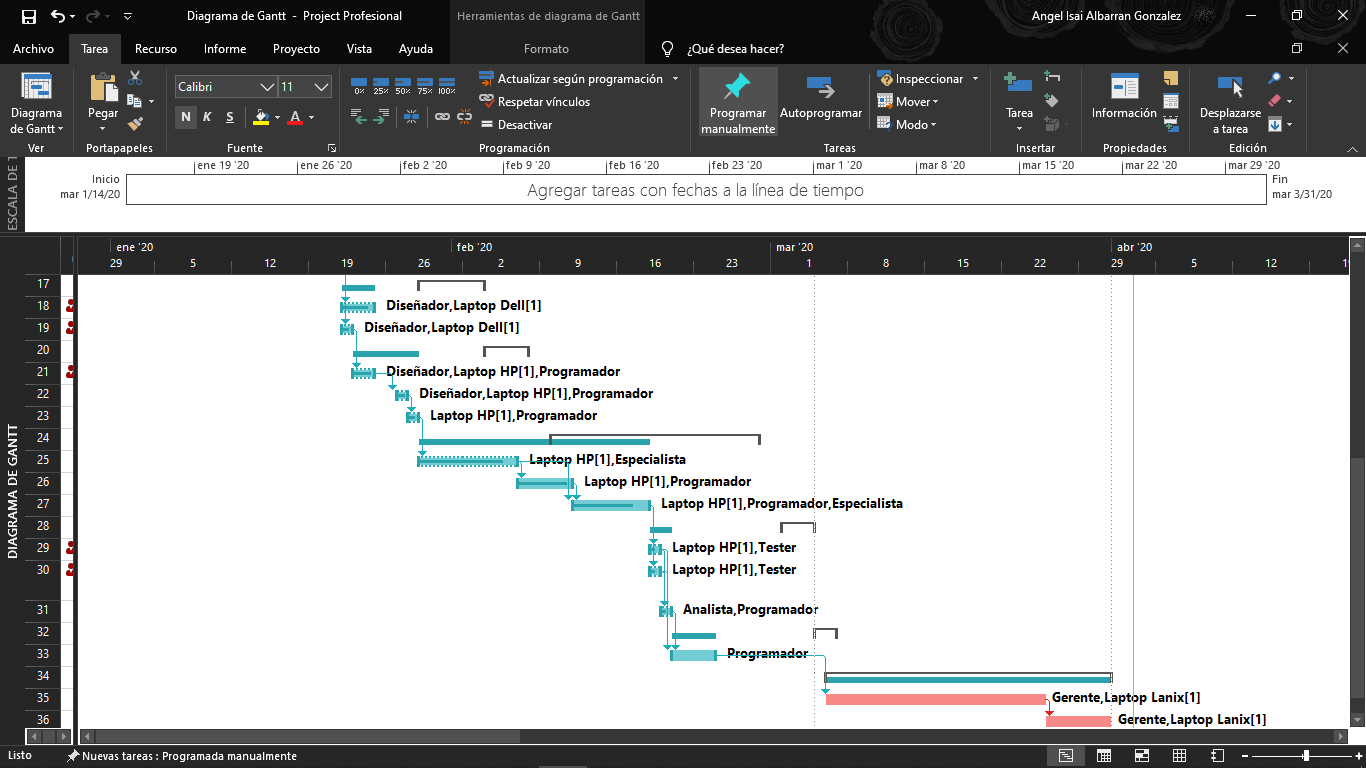


Figura . Diagrama de Gantt 2

[Vinculo Diagrama de Gantt](Entregables/Diagrama%20de%20Gantt.mpp)

## **Diagrama de ruta crítica**

De la de validación a la actividad de pruebas se tiene 2 días de holgura, de la actividad de informe final y acta de fin de proyecto se tiene como holgura 3 días teniendo como total 5 días de holgura, lo cual puede funcionar por si en algún momento se retrasa alguna actividad no se pierda tiempo como tal del proyecto y se siga manteniendo el tiempo establecido para finalizar el proyecto con éxito.

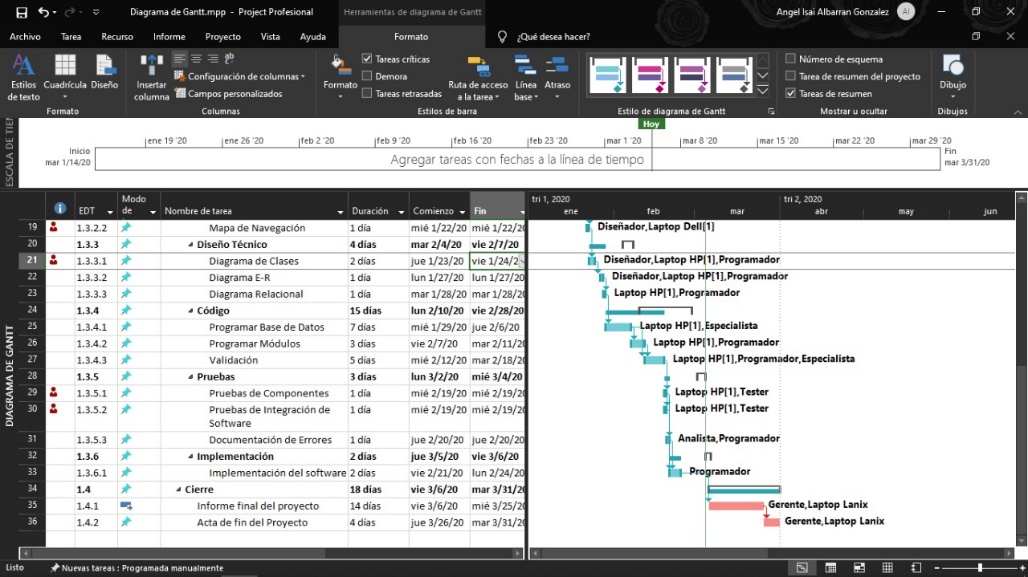


Figura . Diagrama de Ruta Crítica

[Vinculo Diagrama de Ruta Crítica](Entregables/Diagrama%20de%20Gantt.mpp)

# **ESTRUCTURA DEL PROYECTO**

La estructura del proyecto se define con: una carpeta donde se encontraran todos los directorios principales. Al tener las carpetas principales en el mismo directorio es sencillo organizar elementos del “back-end” y mantenerlos separados del “front-end”, facilitando mayormente la programación. Se puede observar que en la parte administrativa se encuentra en el directorio “Admin” todo lo relacionado a la administración y gestion que se se definen en este proyecto. Igualmente creemos que es mas cencillo integrar el proyecto siguendo esta estructura ya que tanto la parte administrativa como la parte frontal inculyen su propia carpeta de librerías que incluyen scripts de estilo CSS y scripts de funcionalidad JS.

La carpeta administrativa incluye un directorio de modulos. En el cual iran los distintos elementos de que conformaran el panel administrativo. Al tenerlos segmentados cada uno por carpetas podemos observar que cada modulo y sus scripts de funcionalidad pueden ser modificados por separado.

La seccion de componentes es la carpeta del conjunto donde se localizan los bloques que conforman la parte del sitio web visible al cliente.

Tomamos en consideraciòn este proyecto mas que nada por la facilidad que representa el poder juntar diversos segmentos del proyecto en su momento dado.

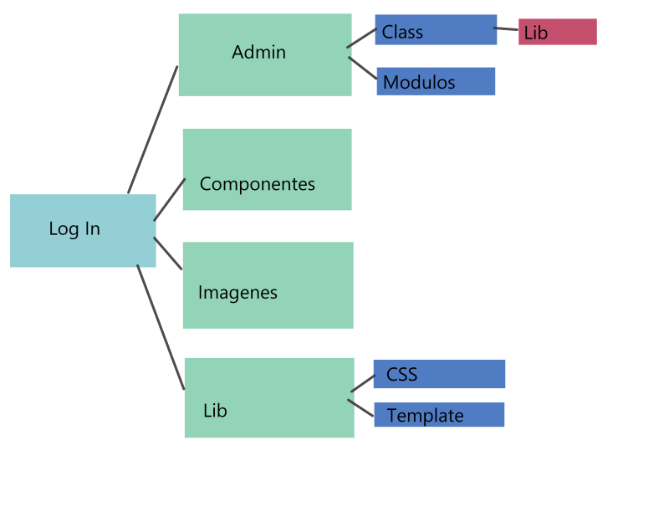


Figura . Diagrama de Estructura de carpetas

# **REFLEXIÓN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Integrantes |  |  |  |  | Total |
| 1. Angel Isai Albarrán González | 8 | 9 | 8 | 9 | 34 |
| 2. Juan Jesús Delgado López | 9 | 9 | 9 | 9 | 36 |
| 3. Alfonso Mondragón Montes | 8 | 9 | 8 | 8 | 33 |
| 4. Erika Vega Valdes | 10 | 9 | 9 | 9 | 37 |

# **BIBLIOGRAFÍA**

Anderson, J. (1 de Febrero de 2018). *www.cuidatudinero.com*. Recuperado el 26 de Mayo de 2020, de www.cuidatudinero.com: https://www.cuidatudinero.com/13098742/definicion-de-gestion-de-proyecto-ti

Aston, B. (21 de Enero de 2019). *thedigitalprojectmanager.com*. Recuperado el 27 de Mayo de 2020, de thedigitalprojectmanager.com: https://thedigitalprojectmanager.com/es/por-que-es-importante-la-gestion-de-proyectos/

Huerta, E. B., Ramírez Cruz, F., & Sánchez, L. (17 de Diciembre de 2016). *Estándar IEEE 830*. Obtenido de Estándares de TI: https://estandaresti.wordpress.com/2016/12/17/estandar-ieee-830-1998/

ISO. (1 de Febrero de 2018). *ISO/IEC 27000:2018*. Obtenido de ISO: https://www.iso.org/standard/73906.html

ISOtools. (1 de Enero de 2020). *ISO 9001*. Obtenido de ISOtools: https://www.isotools.org/normas/calidad/iso-9001/

Kenos. (7 de Febrero de 2018). *www.kenos.com.mx*. Recuperado el 27 de Mayo de 2020, de www.kenos.com.mx: https://www.kenos.com.mx/gestion-proyectos-ti-kenos/

Medina Medina, F. J. (1 de Enero de 2003). *informaticapc.com*. Recuperado el 23 de Mayo de 2020, de informaticapc.com: https://informaticapc.com/patrones-de-diseno/composite.php

Medina Medina, F. J. (1 de Enero de 2003). *informaticapc.com*. Recuperado el 23 de Mayo de 2020, de informaticapc.com: https://informaticapc.com/patrones-de-diseno/bridge.php

Medina Medina, F. J. (1 de Enero de 2003). *informaticapc.com*. Recuperado el 23 de Mayo de 2020, de informaticapc.com: https://informaticapc.com/patrones-de-diseno/adapter.php

Medina Medina, F. J. (1 de Enero de 2003). *informaticapc.com*. Recuperado el 23 de Mayo de 2020, de informaticapc.com: https://informaticapc.com/patrones-de-diseno/singleton.php

misaelutec. (1 de Enero de 2020). *Caracteristicas de los modelos cmmi,psp,tsp e ieee* . Obtenido de Misaelutec: https://misaelutec.wordpress.com/caracteristicas-de-los-modelos-cmmipsptsp-e-ieee-para-dante/

Perezanta Ordoñez, F. J. (24 de Marzo de 2014). *www.eluniversalqueretaro.mx/*. Recuperado el 27 de Mayo de 2020, de www.eluniversalqueretaro.mx/: http://www.eluniversalqueretaro.mx/content/la-importancia-de-la-administracion-de-proyectos

Pressman, R. S. (2002). *INGENIERÍA DEL SOFTWARE.* Madrid: McGRAW-HILL.

Rivas, M. (29 de Mayo de 2015). *Prezi.com-IEEE-829*. Recuperado el 22 de Enero de 2020, de Prezi.com-IEEE-829: https://prezi.com/yh-yob98\_ayd/ieee-829-norma-para-el-software-y-la-documentacion-de-prueba/

Tutorialspoint. (1 de Enero de 2020). *CMMI - Niveles de Madurez*. Obtenido de Tutorialspoint: https://www.tutorialspoint.com/es/cmmi/cmmi\_maturity\_levels.htm

# **DIAGRAMA DE ACTIVIDADES**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Actividad | Responsable (s) | Fechas | | | | Áreas de mejora y tentativa solución |
| Inicio Propuesto | Fin  Propuesto | Inicio Real | Fin Real |
| 1 | Definir problemática, Objetivo, alcance, usuarios de la aplicación, mercado meta, justificar patrones de diseño. | Juan Jesús | 18//05/2020 | 24//05/2020 | 18/05/2020 | 25/05/2020 | Organizar mejor el tiempo, realizando revisiones de manera seguida. |
| 2 | Script del proceso de administración, estándares de calidad aplicables, esquema del ciclo de vida, justificación de las actividades del proceso. | Angel | 18//05/2020 | 24//05/2020 | 18/05/2020 | 26/05/2020 | Organizar mejor el tiempo, realizando revisiones de manera seguida. Analizar mas afondo la información requerida. |
| 3 | Carta de inicio, minuta, Organigrama, alcance, elementos clave, perfiles de los participantes, matriz de responsabilidades, lista de factores. | Erika | 18//05/2020 | 24//05/2020 | 18/05/2020 | 25/05/2020 | Organizar mejor el tiempo, realizando revisiones de manera seguida. |
| 4 | Información clave para cada actor | Alfonso | 24//05/2020 | 27//05/2020 | 18/05/2020 | 26/05/2020 | Organizar mejor el tiempo, realizando revisiones de manera seguida. |
| 5 | Métodos de comunicación | Alfonso | 24//05/2020 | 27//05/2020 | 26/05/2020 | 28/05/2020 |  |
| 6 | Diagrama de Gantt y ruta crítica. | Alfonso | 18//05/2020 | 23//05/2020 | 18/05/2020 | 28/05/2020 |  |
| 7 | Estructura del proyecto | Alfonso | 24//05/2020 | 27//05/2020 | 26/05/2020 | 28/05/2020 |  |