**Documento de Especificación de Sistema**

**DeliverYApp**

1. **Introducción**
   1. **Propósito del Documento**

Este documento tiene como propósito definir las especificaciones funcionales y no funcionales del sistema para la implementación de una aplicación WEB, capaz de gestionar los pedidos de los restaurantes.

* 1. **Alcance del Producto**

Diseño, desarrollo e implementación del Sistema “DeliverYApp”.

DeliverYApp será un entorno web en el que se podrán registrar las empresas gastronómicas y así de esa forma poder ofrecer el servicio de delivery dicho sistema permitirá efectuar pedidos del menú seleccionado por el cliente, una vez confirmado el pedido el usuario recibe las notificaciones de los pedidos y los procesa.

No contará con la funcionalidad de facturación y contabilidad.

Los objetivos del sistema son minimizar los problemas referentes a la gestión de pedidos de parte del cliente, optimizar la gestión de ventas.

El beneficio con que cuenta el sistema es el de agilizar los pedidos de tal forma que no se pierda tiempo con las llamadas.

La meta a alcanzar es mejorar el servicio de delivery y aumentar las ventas.

* 1. **Definiciones, acrónimos y abreviaturas** 
     1. **Del Negocio**

1. **Registrarse:** Proceso en el cualel usuario deberá de registrarse por medio de un formulario o por medio de las cuentas de redes sociales.
2. **Pedido/s:** Proceso en el que el cliente confirma su pedido del menú.
3. **Cancelación de Pedido:** Proceso en el cual el cliente cancela su pedido.
4. **Recepcionista:** Usuario encargado de recibir los pedidos y procesarla.
   * 1. **Del Sistema**
5. **Administrar:** acción de agregar, modificar, eliminar y consultar la información de un determinado objeto o persona.
6. **Usuario:** persona que puede ingresar a la parte privada del Sistema utilizando su usuario y contraseña.
7. **Rol:** conjunto de permiso que pueden asignarse a un usuario.
8. **Administrador del Sistema:** encargado del soporte técnico y operativo del Sistema.
9. **Pruebas:** Proceso mediante el cual se realizan pruebas para el perfecto funcionamiento del Sistema.
   * 1. **De Tecnología**
10. **Servidor web:** es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que llamamos hipertextos, pagina Web o páginas HTML.
11. **HTTP:** Protocolo de transferencia de hipertexto (HiperText Transfer Protocol) es el protocolo usado en cada transacción.
12. **Sistema de Gestión de Base de Datos (SGBD):** Son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.
13. **Base de Datos:** es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
14. **Aplicación:** es un programa informático para facilitar al usuario la realización de un determinado trabajo.
15. **MySQL:** Motor de bases de datos.
16. **RoR (Ruby on Rails):** Lenguaje de programación usado para la creación de aplicación.
17. **Apache:** servidor Web de código abierto.
18. **Dirección IP (Internet Protocol):** número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo dentro de una red que utilice el protocolo IP.
    1. **Referencias**

Especificación de Requisitos según el estándar de IEEE 830 – IEEE Std. 830-1998.

* 1. **Visión General del documento (Descripción del resto del documento)**

En este documento se encontrará las características del sistema, interfaces del sistema, interfaces del usuario, características de los usuarios, descripción de los requerimientos funcionales, no funcionales y del sistema.

1. **Descripción General**
   1. **Perspectiva del producto.**

El sistema DeliverYApp será un producto diseñado para trabajar en entornos web, lo que permitirá su utilización de forma descentralizada, además trabajará de manera independiente por lo tanto no interactuará con otro sistemas.

* 1. **Funciones del Producto**.

El sistema DeliverYApp permitirá realizar las siguientes funciones:

1. **Registro de Empresa (Login - Empresa):** Se le proveerá de un usuario y contraseña para que pueda acceder y tener control de los servicios que provee el sistema.
2. **Gestión de roles:** Se especificará los niveles de accesos del sistema.
3. **Gestión de usuario:** Se crearan los diferentes usuarios del sistema indicando sus roles.
4. **Gestión de menú:** Este contendrá un formulario para el registro de los datos del producto, platos, precios, foto, etc.
5. **Gestión de pedidos:** Este contendrá opciones para manejar los estados de los pedidos.
6. **Gestión de informes**: Este contendrá opción para generar informes estadísticos.
7. **Registro de cliente (Login - cliente):** El cliente deberá registrarse antes de solicitar el pedido.
8. **Elección y confirmación del pedido:** Este contendrá opciones de elección de los diferentes tipos de platos que a su vez deberá ser confirmado.
   1. **Características de los usuarios**:

* **Administrador del sistema:** usuario con gran conocimiento en el manejo del sistema, con una previa capacitación por parte de la entidad.
* **Usuario del sistema:** es el encargado de gestionar los pedidos, con una previa capacitación por parte de la entidad.
* **Cliente:** usuario visitante que solita los pedidos, debe contar con un conocimiento básico sobre el manejo de páginas web.
  1. **Restricciones**
* **Políticas de la empresa:** La aplicación se desarrollara mediante software de licencia abierta por lo tanto no se deberá pagar por el uso del servidor web(apache), sistema de gestión de base de datos (MySQL) y el lenguaje de programación RoR, por lo tanto la utilización de estos programas se harán mediantes las políticas establecidas por este tipo de licenciamiento.
* **Limitaciones del hardware:** para esta aplicación será necesaria un computador servidor en el cual se instalara el servidor web apache, MySQl, RoR y la aplicación DeliverYApp.
* **Interfaces con otras aplicaciones:** Debido a que el sistema no interactúa con otro sistema y es autónomo no se desarrollaran interfaces con otras aplicaciones. Las conexiones necesarias para la utilización del servidor web, MySQL, RoR y un DNS será por medio de la configuración de estos programas.
* **Funciones de control:** El sistema debe controlar los permisos que tiene los usuarios para su accesibilidad de una manera correcta, de tal forma que pueda acceder la información que le corresponde de acuerdo a su rol. Debe tener controles adecuados para la validación de datos.
* **Lenguaje de programación:** Todo el material que se realice para el usuario y la aplicación debe estar en lenguaje español.
* **Protocolo de comunicación:** Se usar los protocolos TCP/IP, HTTP.
* **Requisitos de habilidad:** La información correspondiente al pedido como numero de documento, nombre, dirección, teléfono y elección de menú deben estar ajustado a la realidad para evitar inconvenientes y garantizar el pedido.
* **Credibilidad en la aplicación:** El sistema deberá ser sometida a una serie de pruebas para establecer que se encuentra conforme a los requerimiento que se plasman en el documento, en tanto en la consistencia de datos como el rendimiento de la aplicación, tales como tiempo de respuesta.
* **Consideración acerca de la seguridad:** Cada Usuario deberá autenticarse y su acceso verificado por una sola terminal para su respectiva labor de acuerdo de lo que su labor especifica.
  1. **Suposiciones y Dependencias**
* La red interna deberá estar configurada para el manejo de protocolos TCP/IP, HTTP, DNS principalmente todo lo relacionado a desempeño y seguridad.
* Debe realzarse una capacitación adecuada y acorde a lo que cada usuario va a realizar. Su capacitación se hará en el momento que sea necesario y a las personas indicadas.

1. **Requisitos Específicos**
   1. **Interfaces Externas**

El sistema DeliverYApp no tendrá interconexión con otros sistemas informáticos por lo tanto no es necesario la utilización de interfaces específicas para este punto.

* 1. **Funciones del Sistema**
* El cliente primeramente deberá registrarse para realizar su pedido.
* Deberá contar con los módulos citados más arriba.
* Contará con módulos de reportes estadísticos de los pedidos realizados.
  1. **Rendimiento del sistema**
* Necesitará una terminal como mínimo.
* El computador del usuario deberá contar con un navegador web (Google Chrome, Firefox, Internet Explorer).
* Conexión a internet.
  1. **Requerimientos lógicos del sistema de base de datos**
* El sistema debe permitir la retención de datos para guardar un registro de ellos para posterior reportes.
* El sistema debe tener en cuenta que información solo puede ser eliminada y cual es de gran importancia.
* La infraestructura debe ser capaz de soportar estos procedimientos.
  1. **Propiedades emergentes del Sistema**
* Seguridad: será por medio de un usuario y una contraseña, el encargado de la asignación de tal tarea será el administrador del sistema teniendo en cuenta el nivel de acceso a cada usuario.
* Portabilidad: puesto que funcionara con cualquier plataforma web.
  1. **Características de calidad**

1. **Apéndice**

**Capítulo I**

**Modelado de Requerimiento del Sistema**

**Descripción Escrita del modelado del Sistema**

El cliente desde su navegador ingresa a la aplicacion “DeliverYApp” primeramente se debe registrar (Loguearse) con los datos requeridos por el sistema luego de eso podrá ver un mensaje de bienvenida y una lista de restaurantes en el cual tendrá la opción de elegir uno de ellos, una vez seleccionado el restaurante se desplegará una interfaz con las opciones del menú donde el cliente por medio de imágenes con sus respectivas descripciones optara por su plato de preferencia. Una vez seleccionado el menú el cliente tiene como opción marcar si su pedido será a domicilio o si pasara a retirar. Realizado esto aparecerán los datos personales con la lista de pedido que ha hecho el cliente con sus respectivos precios y el monto total a abonar. Si el cliente está de acuerdo presiona el botón para confirmar su pedido y se le mostrara un mensaje de recepción y confirmación.

**Tabla de Caso de Uso**

|  |  |
| --- | --- |
| **Nombre** | Grupo DeliverYApp |
| **Autores:** | Cliente, Sistema Autor |
| **Fecha:** | 06-04-2016 |
| **Descripción:** | Permite hacer el pedido de su restaurant preferido. |
| **Precondiciones** | El Cliente debe de estar logueado |
| **Flujo normal** | |
| 1. El cliente ingresa a la aplicación de DeliverYApp se loguea. 2. La aplicación le da la bienvenida y la opción de elegir el restaurante. 3. Una vez que el cliente elige el restaurante tiene las opciones de Menú, Bebidas, Ofertas del día, Combo etc. 4. El cliente elige unas de las opciones aparece la foto de las comidas y debajo de cada foto de comida el precio y los detalles. 5. El cliente selecciona el menú que desea una vez que esté satisfecho con el pedido ,Presiona la opción de Realizar pedido 6. En el Sistema aparecerá la opción de elegir si el pedido realizado se va a enviar o se va a ir a buscar ,ya seleccionado una de estas opciones aparece los datos personales y la lista de pedido que el cliente ha seleccionado y el monto total que va a alcanzar 7. Si el cliente está de acuerdo presiona el botón “Confirmar Pedido” 8. Se muestra un mensaje de confirmación “solicitud de pedido DeliveYApp Enviado “ 9. El usuario Recibe la notificación de pedido del cliente 10. El usuario pasa el pedido a la cocina | |
| **Flujo alternativo** | |
| 7- Cancelar pedido le lleva al paso 2 | |
| **Precondiciones: Se realiza pedido de (menú)** | |

**Gráfico de caso de uso general**



1.1 Registrarse de parte del Usuario

|  |  |
| --- | --- |
| **Nombre** | Grupo DeliverYApp |
| **Autores:** | Cliente, Sistema Autor |
| **Fecha:** | 06-04-2016 |
| **Descripción:** | Registrarse en el sistema |
| **Precondiciones** | El cliente debe contar con internet. |
| **Flujo normal** | |
| 1. Una vez adquirido el sistema se le provee al usuario el rol de Administrador del sistema. 2. Este tendrá acceso a las funciones de Crear, Editar, Eliminar, Guardar 3. Creará roles para sus empleados de acuerdo a sus funciones. | |
| **Flujo alternativo** | |
| En caso de no contar con el rol de Administrador deberá solicitarlo. | |
| **Precondiciones: Podrá acceder a la interfaz del sistema.** | |

1.2 Administrador de Roles

|  |  |
| --- | --- |
| **Nombre** | Grupo DeliverYApp |
| **Autores:** | Cliente, Sistema Autor |
| **Fecha:** | 06-04-2016 |
| **Descripción:** | Asignar Roles |
| **Precondiciones** | Contar con el Rol de Administrador del sistema |
| **Flujo normal** | |
| 1. Primeramente el administrador deberá Registrar a sus empleados con los datos solicitados en el formulario. 2. El administrador ingresa al módulo Asignar Roles 3. Se muestra la interfaz con un formulario con los datos a rellenar Usuario, cargo y contraseña. 4. Luego guarda y registra. | |
| **Flujo alternativo** | |
| Deberá estar registrado como empleado | |
| **Precondiciones: Guardara al cliente con el rol asignado.** | |

1.3

|  |  |
| --- | --- |
| **Nombre** | Grupo DeliverYApp |
| **Autores:** | Cliente, Sistema Autor |
| **Fecha:** | 06-04-2016 |
| **Descripción:** | Registrar los menú |
| **Precondiciones** | Tener un rol |
| **Flujo normal** | |
| 1. El usuario asignado deberá ingresar con su usuario y contraseña 2. Se habilita el menú 3. Ingresa al módulo registrar menú 4. Se muestra el formulario con los campos a rellenar. 5. Una vez rellenado el formulario tiene la opción de agregar más. 6. En caso de que no tenga que agregar más tiene la opción de guardar. 7. Si por algún motivo relleno mal tiene la opción de eliminar o cancelar. | |
| **Flujo alternativo** | |
| Si se cancela vuelve al paso 4 | |
| **Precondiciones: creación del menú** | |

**Capítulo II**

***Modelado de Requerimiento del Negocio***

**Descripción Escrita del Modelado de negocio**

El cliente llama al negocio, el recepcionista o encargado atiende la llamada y le consulta sobre su necesidad, el cliente si no conoce el menú o lo que le gustaría consumir le pregunta al encargado sobre el menú vigente, el encargado le dicta el menú posteriormente el cliente elige según su preferencia, el encargado pregunta si está satisfecho con su pedido o si le gustaría agregar algo más, el cliente pide algo mas o solo confirma el pedido, el encargado solicita los datos personales del cliente como: nombre, número de teléfono y dirección, una vez obtenido los datos el encargado le comunica al cliente el tiempo estimado en el que le llegara el pedido y finaliza la llamada, el pedido es pasado de la recepción a la cocina en donde es procesada donde luego de elaborar completamente el pedido, esta es remitida al cliente en cuestión.

**Tabla de Caso de Uso**

|  |  |
| --- | --- |
| **Nombre** | Grupo DeliverYapp |
| **Autores:** | Cliente, Usuario |
| **Fecha:** | 06-04-2016 |
| **Descripción:** | Proceso de Pedido de Delivery |
| **Precondiciones** | El cliente debe de proporcionar sus datos personales solicitados. |
| **Flujo normal** | |
| 1. El Cliente llama por teléfono al restaurante 2. El usuario contesta la llamada y pregunta que se le ofrece 3. El usuario consulta las opciones de comidas.( en caso de que el cliente ya sabe lo que quiere pasamos al paso 5) 4. El usuario le dicta el menú 5. El usuario elige el menú 6. El usuario anota al cliente su pedido 7. El usuario Pregunta si quiere agregar algo más a su pedido 8. En caso d que el cliente ya esté satisfecho con su pedido solo confirma o si desea algo más volvemos al paso 5 9. El usuario solicita los datos personales al cliente(nombre, apellido, dirección, número teléfono) 10. El usuario comunica el tiempo estimado en que llegara el pedido finaliza la llamada 11. El cliente le da su Ok y finaliza la llamada 12. El usuario pasa el pedido a la cocina | |
| **Flujo alternativo** | |
| 2- Si el usuario no contesta el cliente pasa por el restaurante   1. Si el cliente no pasa los datos personales el usuario le dice al cliente que no podrá mandarle su pedido | |
| **Precondiciones: Se generó pedido de delivery** | |

**Grafico**



**Estudio de Viabilidad**

* **Técnica**

Se ha evaluado la opción de escoger el lenguaje de programación de Ruby on Rails (RoR) por el hecho de que la mayoría de los integrantes del grupo poseen un conocimiento básico, dispuestos en aprender e innovar y además porque ofrece una inmensa librerías (gemas) que nos facilitara muchas cosas en el momento de desarrollar el sistema.

* **Económicamente**

No representa un costo muy elevado.

* **Operativamente**

Cuenta con los recursos humanos preparados y la infraestructura necesarias para desarrollar dicho sistema.

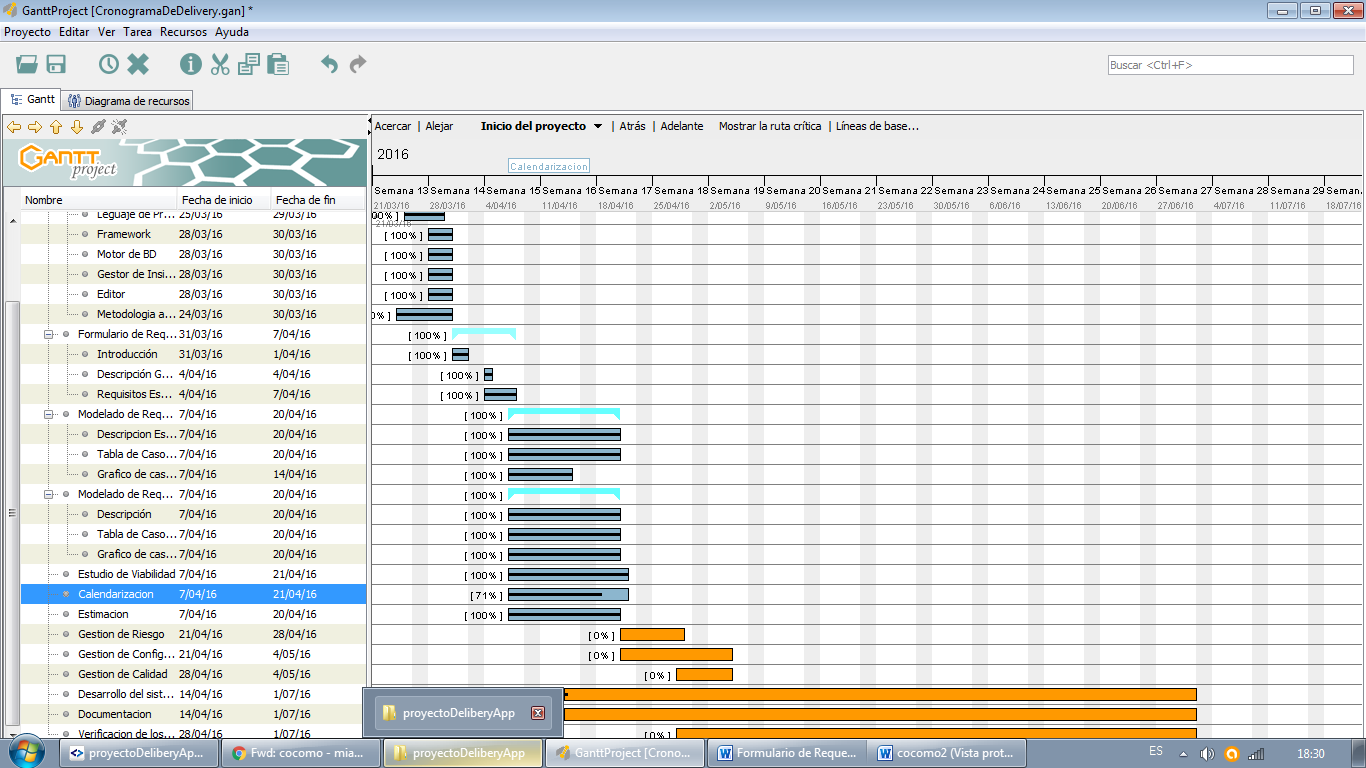
De acuerdo al análisis realizado hemos comprobado que el sistema de “delivery” a desarrollar es viable por el hecho de que les facilita a los usuarios realizar sus pedidos de forma más rápida e intuitiva.

Después de realizar el análisis se concluye que es viable el desarrollo e implementación del Sistema “DeliverYApp ya que se dispone de los recursos necesarios para llevarla a cabo.

**Capítulo III**

**Calendarización**





**Capítulo IV**

**Estimación de Costos**

**COCOMO2**

El Modelo Constructivo de Costos (COCOMO, por su acrónimo del inglés COnstructive COst MOdel)

COCOMO ll es un modelo (algoritmo) que permite estimar el coste, esfuerzo y tiempo   
cuando se planifica una nueva actividad de desarrollo software.

El CocoMo surgió para medir y calcular el coste y el tiempo de un determinado proyecto basándose fundamentalmente en las líneas de código y algunas constantes.

Surge como una alternativa para incluir componentes de incerteza en las estimaciones, conforme al nivel de información disponible. Este es un modelo paramétrico que establece ecuaciones matemáticas para describir las relaciones entre el tamaño del software - factor primario de costo usualmente representado en términos de puntos de función - y otros factores secundarios que buscan capturar particularidades de producto, proceso, personas y plataforma.

EI primer modelo CocoMo se publicó en 1981 por Barry Boehm. En los 15 años siguientes las técnicas de desarrollo de software cambiaron drásticamente. A raíz de esto surgió el CocoMo II, publicado por primera vez en 1997.

**Características**

* Es una herramienta basada en las líneas de código la cual la hace muy poderosa para la estimación de costos y no como otros que solamente miden el esfuerzo en base al tamaño.
* Representa el más extenso modelo empírico para la estimación de [software](http://www.ecured.cu/Software).
* Existen herramientas automáticas que estiman costos basados en COCOMO como ser: Costar, COCOMO 81.

**Objetivos para la construcción de COCOMO II**

* Desarrollar un modelo de estimación de costo y cronograma de proyectos de [software](http://www.ecured.cu/Software) que se adaptara tanto a las prácticas de desarrollo de la década del 90 como a las futuras.
* Construir una [base de datos](http://www.ecured.cu/Base_de_datos) de proyectos de software que permitiera la calibración continua del modelo, y así incrementar la precisión en la estimación.
* Implementar una herramienta de [software](http://www.ecured.cu/Software) que soportara el modelo.
* Proveer un marco analítico cuantitativo y un conjunto de herramientas y técnicas que evaluaran el impacto de las mejoras tecnológicas de software sobre los costos y tiempos en las diferentes etapas del ciclo de vida de desarrollo.

**Formula**

E=a(KI)° · m(X) donde:

"E" es Salario/mes (Media).

"a" y "b" son constantes según el modo (Orgánico, Semilibre o Rígido).

"KI" es la cantidad de líneas de código (En miles).

"m(X)" es el multiplicador que depende de 15 atributos constantes.

**Modelos de COCOMO II**

Los tres modelos de COCOMO II se adaptan tanto a las necesidades de los diferentes sectores, como al tipo y cantidad de información disponible en cada etapa del ciclo de vida de desarrollo, lo que se conoce por granularidad de la información. Estos tres modelos son:

* Modelo de composición de aplicación. Utilizado durante las primeras etapas de la [Ingeniería del software](http://www.ecured.cu/Ingenier%C3%ADa_de_software), donde el prototipado de las interfaces de usuario, la interacción del sistema y del software, la evaluación del rendimiento, y la evaluación de la madurez de la tecnología son de suma importancia.
* Modelo de fase de diseño previo. Utilizado una vez que se han estabilizado los requisitos y que se ha establecido la arquitectura básica del software.
* Modelo de fase posterior a la arquitectura. Utilizado durante la construcción del software.

**Ventajas**

* Es fácil de realizar y de interpretar.
* Tiene pocas variables.
* Se acerca a la realidad en la mayoría de los casos.

**Desventajas**

* No saca resultados fiables en proyectos demasiado pequeños.
* La elección de las variables es muy subjetiva y depende de la persona que realiza el estudio.

**Conclusión**

Cocomo 2 es una buena opción para proyectos medianos/grandes aunque para su realización es necesario conocer cuál va a ser el equipo de desarrolladores, el equipo Informático, el entorno etc.

**Capítulo V**

**Descripción de Roles**

**Líder del Proyecto:**

Objetivos:

* Coordinar el quipo
* Asegurar que todos cumplan con sus trabajos.

Responsabilidad:

* Desarrollo del plan del proyecto.
* Identificación de los requerimientos y el alcance del proyecto.
* Comunicación.
* Administración de los recursos humanos y materiales.
* Control de tiempos.
* Identificación y control de riesgos.
* Administración de los costos/presupuesto, el aseguramiento de la calidad.
* El reporte y evaluación del desempeño del proyecto.
* El líder de proyecto debe mantener su foco en asegurar que el proyecto se termine en el tiempo y presupuesto planeado, y muy frecuentemente con tiempos limitados.

**Documentador:**

El objetivo principal del documentador es el de actuar como medio de comunicación entre los miembros del equipo.

* Mantener la información generada durante el proceso de desarrollo con un adecuado procesamiento que permita la calidad en el mantenimiento de la misma.
* Organizar y garantizar el almacenamiento y recuperación de la documentación de los procesos y productos más recientes durante el desarrollo, manteniendo así la información al día.
* Mantener la consistencia en la apariencia y estructura de los documentos, facilitando su almacenamiento, recuperación e intercambio, no permitiendo el almacenamiento de documentos con formatos diferentes al establecido.
* Generar plantillas que permitan una adecuada estructura y uniformidad en los documentos que lo requieran.
* Elaborar, almacenar y permitir la recuperación de la documentación que se genera como parte de la gestión del proyecto.
* Construir documentación de cara al usuario que contempla los aspectos que son de utilidad para los usuarios finales del sistema y proveer una adecuada capacitación.

La documentación es necesaria para informar a todos los miembros del equipo de desarrollo acerca del estado y evolución del proyecto.

**Desarrollador:**

Debe convertir las especificaciones del sistema en código fuente ejecutable utilizando uno o más lenguajes de programación, así como herramientas de software de apoyo a la programación.

* Creación del código ejecutable.
* Creación de la documentación de operación.
* Integración del entorno.
* Conducir revisiones del software.
* Realizar mantenimiento y actualizaciones del sistema existentes, las cuales serán solicitados a través del formulario de requerimiento de desarrollo.
* Realizar pruebas del programa desarrollado con los usuarios solicitantes.

**Tester:**

El objetivo principal del tester es el de diseñar test que en forma sistemática, pueda eliminar diferentes casos de errores. Realizando esto con la mínima cantidad de tiempo y esfuerzo.

* Construir y aplicar los planes de prueba unitarios de modulo, de sistema, de aceptación parcial, manteniéndolos actualizados durante el proyecto.
* Velar por la amplitud y la exactitud de todos los documentos del proyecto.
* Velar por el estándar adoptado para el desarrollo
* Velar por la calidad del producto final.

**Gestor de Riesgo:**

* Documentos de requisitos de Usuario y Software.
* Plan de administración del proyecto.
* Plan de testeo.
* Fase de diseño arquitectónico.
* Fase de diseño detallado.
* Políticas de control de cambios, control de errores y control de la configuración.
* Documentación.
* Descubrir errores en funciones, lógica e implementación en cualquiera de las representaciones del software.
* Verificar que el software bajo revisión cumple con los requisitos.
* Asegurarse que el software ha sido representado de acuerdo al estándar en uso.
* Alcanzar software que es desarrollado en forma uniforme.
* Hacer el proyecto más manejable.

**Gestor de Configuración:**

* Gestionar la planificación, identificación, control, seguimiento y auditoría de todos los elementos de configuración en la base de datos de configuración.
* Desarrollar el plan de gestión de configuración.
* Promover el uso efectivo de la CMDB.( configuration management database)
* Monitorizar y reportar los cambios no autorizados sobre los CIs. (Configuration Items)
* Asegurar la consistencia e integridad de los datos de la CMDB a través de la ejecución de procedimientos de verificación y auditoría.
* Revisar tipos de elementos de configuración, relaciones, atributos y valores asociados, estructura de la base de datos, derechos de acceso.
* Aprobar cambios estructurales en la CMDB.
* Coordinador de configuración.
* Reportar cualquier discrepancia o no conformidad en los CIs al gestor de configuración.
* Participar en la mejora continua del proceso de gestión de configuración.

**RIESGOS**

|  |  |  |
| --- | --- | --- |
| **Riesgo** | **Tipo** | **Descripción** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |