****

**UNIVERSIDAD AUTONOMA DE ENCARNACION**

**FACULTAD DE CIENCIAS, ARTE Y TECNOLOGIA**

**Documento de Especificación del Sistema**

**DeliverYApp**

**Índice**

Contenido

[**1.** **Introducción** 5](#_Toc459392240)

[**1.1** **Propósito del Documento** 5](#_Toc459392241)

[**1.2** **Alcance del Producto** 6](#_Toc459392242)

[**1.3** **Definiciones, acrónimos y abreviaturas** 6](#_Toc459392243)

[1.3.1 Del Negocio 6](#_Toc459392244)

[1.3.2 Del Sistema 6](#_Toc459392245)

[1.3.3 De Tecnología 6](#_Toc459392246)

[**1.4** **Referencias** 7](#_Toc459392247)

[**1.5** **Visión General del documento** 7](#_Toc459392248)

[**2.** **Descripción General** 7](#_Toc459392249)

[**2.1** **Perspectiva del producto.** 7](#_Toc459392250)

[**2.2** **Funciones del Producto**. 7](#_Toc459392251)

[**2.3** **Características de los usuarios**: 8](#_Toc459392252)

[**2.4** **Restricciones** 8](#_Toc459392253)

[**2.5** **Suposiciones y Dependencias** 8](#_Toc459392254)

[**3.** **Requisitos Específicos** 9](#_Toc459392255)

[**3.1** **Interfaces Externas** 9](#_Toc459392256)

[**3.2** **Funciones del Sistema** 9](#_Toc459392257)

[**3.3** **Rendimiento del sistema** 9](#_Toc459392258)

[**3.4** **Requerimientos lógicos del sistema de base de datos** 9](#_Toc459392259)

[**3.5** **Propiedades emergentes del Sistema** 9](#_Toc459392260)

[**4.** **Necesidad de los diagramas UML** 9](#_Toc459392261)

[Capítulo I 11](#_Toc459392262)

[Capítulo II 17](#_Toc459392263)

[Modelado de Requerimiento del Sistema 17](#_Toc459392264)

[Descripción Escrita del modelado del Sistema 17](#_Toc459392265)

[Capítulo III 32](#_Toc459392266)

[**Estudio de Viabilidad** 33](#_Toc459392267)

[Capítulo IV 34](#_Toc459392268)

[Calendarización 35](#_Toc459392269)

[Capítulo V 36](#_Toc459392270)

[**COCOMO2** 37](#_Toc459392271)

[Características 37](#_Toc459392272)

[Objetivos para la construcción de COCOMO II 37](#_Toc459392273)

[Formula 38](#_Toc459392274)

[Modelos de COCOMO II 38](#_Toc459392275)

[Ventajas 38](#_Toc459392276)

[Desventajas 38](#_Toc459392277)

[Conclusión 38](#_Toc459392278)

[Capítulo VI 39](#_Toc459392279)

[Plan de riesgos 40](#_Toc459392280)

[Introducción 40](#_Toc459392281)

[Propósito 40](#_Toc459392282)

[Alcance 40](#_Toc459392283)

[Análisis de Riesgos 41](#_Toc459392284)

[Capítulo VII 42](#_Toc459392285)

[Plan de Gestión de Configuración 43](#_Toc459392286)

[Introducción 43](#_Toc459392287)

[Propósito 43](#_Toc459392288)

[Alcance 43](#_Toc459392289)

[Gestión de Configuración 43](#_Toc459392290)

[Herramientas, Entorno e Infraestructura 44](#_Toc459392291)

[Herramientas 44](#_Toc459392292)

[Programa de la Gestión de Configuración 44](#_Toc459392293)

[Identificación de la Configuración 44](#_Toc459392294)

[Elementos de Configuración 44](#_Toc459392295)

[Nomenclatura de Elementos 44](#_Toc459392296)

[Elementos de la Línea Base del Proyecto 50](#_Toc459392297)

[Control de Configuración 50](#_Toc459392298)

[Solicitud de Cambios 50](#_Toc459392299)

[Aprobación de Cambios 50](#_Toc459392300)

[Implementación de Cambios 50](#_Toc459392301)

[Estado de la Configuración 51](#_Toc459392302)

[Informes y Auditorías 51](#_Toc459392303)

[Calendario 51](#_Toc459392304)

[Capacitación y Recursos 51](#_Toc459392305)

[Mantenimiento del Plan de Gestión de la Configuración 51](#_Toc459392306)

[Capítulo VIII 52](#_Toc459392307)

[Descripción de Roles 53](#_Toc459392308)

[Líder del Proyecto: 53](#_Toc459392309)

[Documentador: 53](#_Toc459392310)

[Desarrollador: 53](#_Toc459392311)

[Tester: 54](#_Toc459392312)

[Gestor de Riesgo: 54](#_Toc459392313)

[Gestor de Configuración: 54](#_Toc459392314)

[Capítulo IV 55](#_Toc459392315)

[Plan de SQA 56](#_Toc459392316)

[1. Propósito 56](#_Toc459392317)

[2. Referencias 56](#_Toc459392318)

[3. Gestión 56](#_Toc459392319)

[*3.1.* Organización 57](#_Toc459392320)

[3.2. Actividades 57](#_Toc459392321)

[3.2.1. Ciclo de vida del software cubierto por el Plan 57](#_Toc459392322)

[3.2.2. Actividades de calidad a realizarse 57](#_Toc459392323)

[Técnica utilizada 59](#_Toc459392324)

[Problemas detectados 59](#_Toc459392325)

[Problema 1 59](#_Toc459392326)

[Sugerencia de corrección 59](#_Toc459392327)

[Problema 2 59](#_Toc459392328)

[Sugerencia de corrección 59](#_Toc459392329)

[Evaluación 60](#_Toc459392330)

[Estado actual del Producto 60](#_Toc459392331)

[Acciones a tomar 60](#_Toc459392332)

[Próxima Revisión del Producto 60](#_Toc459392333)

[3.3. Responsables 60](#_Toc459392334)

[4. Documentación 61](#_Toc459392335)

[4.1 Propósito 61](#_Toc459392336)

[4.2 Documentación mínima requerida 61](#_Toc459392337)

[4.2.1 Especificación de requerimientos del software 61](#_Toc459392338)

[4.2.2 Descripción del diseño del software 62](#_Toc459392339)

[4.2.3 Plan de Verificación & Validación 63](#_Toc459392340)

[Verificar que las vistas del sistema cumplan con el requerimiento citado en el documento del diseño, una vez verificado se procede a la validación de los mismos. 63](#_Toc459392341)

[5. Estándares, prácticas, convenciones y métricas 63](#_Toc459392342)

[5.1 Estándar de documentación 63](#_Toc459392343)

[5.2 Estándar de verificación y prácticas 63](#_Toc459392344)

[6. Revisiones y auditorías 64](#_Toc459392345)

[6.1 Objetivo 64](#_Toc459392346)

[6.2 Requerimientos mínimos 64](#_Toc459392347)

[6.2.1 Revisión de requerimientos 64](#_Toc459392348)

[6.2.2 Revisión de diseño preliminar 64](#_Toc459392349)

[6.2.3 Revisión de diseño crítico 64](#_Toc459392350)

[6.2.4 Auditoria Funcional 64](#_Toc459392351)

[6.2.5 Auditoria Física 64](#_Toc459392352)

[6.2.6 Auditorías internas al proceso 64](#_Toc459392353)

[6.2.7 Revisiones de gestión 64](#_Toc459392354)

[6.2.8 Revisión del Plan de gestión de configuración 64](#_Toc459392355)

[6.2.9 Revisión Post Mortem 64](#_Toc459392356)

[6.2.10 Agenda 65](#_Toc459392357)

[7. Verificación 65](#_Toc459392358)

[8. Reporte de problemas y acciones correctivas 65](#_Toc459392359)

[9. Herramientas, técnicas y metodologías 65](#_Toc459392360)

[10. Gestión de riesgos 65](#_Toc459392361)

[Anexos 66](#_Toc459392362)

[Formulario de Pedidos y Detección de Cambios 66](#_Toc459392363)

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, la recepcionista recibe las notificaciones y se genera un nuevo pedido con estado pendiente, una vez que el menú se encuentra listo para su entrega, se efectuara el envió del pedido y este pasa a estado enviado, y por ultimo una vez cobrado se finaliza el pedido y este pasa a estado pagado.

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 del restaurante.

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. **Pedido/s:** Proceso en el que el cliente confirma su pedido del menú.
2. **Cancelación de Pedido:** Proceso en el cual el cliente cancela su pedido.
3. **Recepcionista:** encargado de recibir los pedidos y procesarla.
   * 1. **Del Sistema**
4. **Administrar:** acción de agregar, modificar, eliminar y consultar la información de un determinado objeto o persona.
5. **Administrador del Sistema:** encargado del soporte técnico y operativo del Sistema.
6. **Registrarse:** Proceso en el cualel administrador da de alta a los usuarios del sistema, delegando a cada uno sus respectivos roles.
7. **Usuario:** persona que puede ingresar a la parte privada del Sistema utilizando su usuario y contraseña.
8. **Rol:** conjunto de permiso que pueden asignarse a un usuario.
9. **Cliente:** persona encargada de realizar los pedidos.
10. **Pruebas:** Proceso mediante el cual se realizan pruebas para el perfecto funcionamiento del Sistema.
    * 1. **De Tecnología**
11. **Servidor web IIS:** es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que llamamos hipertextos, pagina Web o páginas HTML.
12. **HTTP:** Protocolo de transferencia de hipertexto (HiperText Transfer Protocol) es el protocolo usado en cada transacción.
13. **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, la recepcionista y las aplicaciones que la utilizan.
14. **Base de Datos:** es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
15. **Aplicación:** es un programa informático para facilitar al usuario la realización de un determinado trabajo.
16. **SQL SERVER:** Motor de bases de datos.
17. **C#:** Lenguaje de programación usado para la creación de aplicación.
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.
19. **Balsamiq mockups:** es una aplicación para crear maquetas para interfaces gráficas para usuario. Le permite al diseñador diagramar widgets pre construidos.
20. **ArgoUML:** es una aplicación de diagramado de UML escrita en Java y publicada bajo la Licencia BSD. Dado que es una aplicación Java, está disponible en cualquier plataforma soportada por Java.
21. **ENTITY FRAMEWORK**: genera un conjunto de objeto que están directamente ligados a una base de datos.
    1. **Referencias**

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

* 1. **Visión General del documento**

En este documento se encontrará las características del sistema, interfaces del sistema, 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 C#, 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, C# 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, C# 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 la recepcionista 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, etc.).
* 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. **Necesidad de los diagramas UML**

UML (Lenguaje unificado de Modelado) es un conjunto de normas y estándares gráficos respecto a cómo se deben representar los esquemas relativos al software.

Mediante UML se pueden establecer los requerimientos y la estructura necesaria para desarrollar un sistema antes de codificarlo (escribirlo).

UML está compuesto de diferentes diagramas las cuales reflejan las diferentes etapas del desarrollo del sistema. Estos ayudan a detectar las dependencias y dificultades con la que contara el sistema y a reducir el margen de error a la hora de codificar.

# Capítulo I

**Modelado de Requerimiento del Negocio**

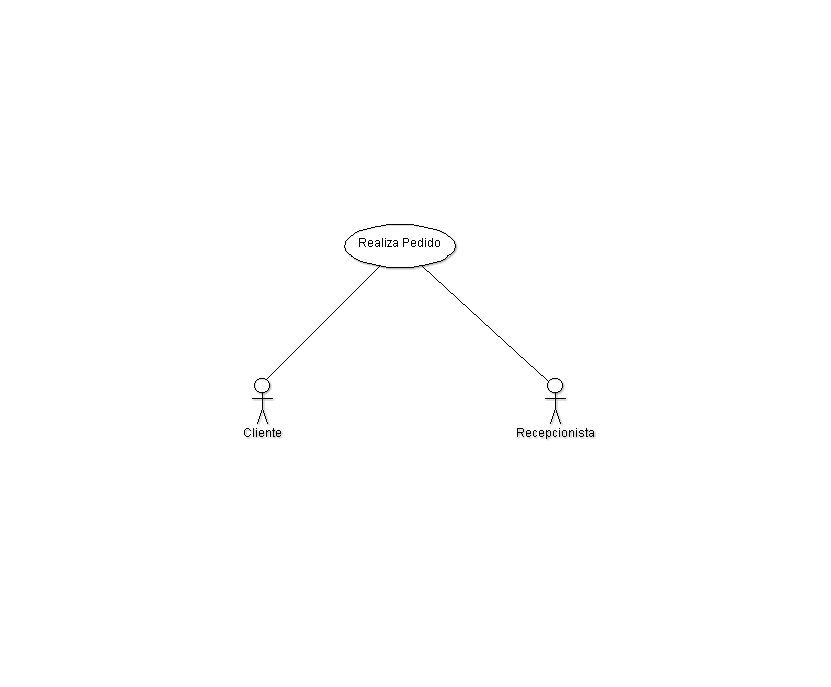
**Descripción Escrita del Modelado de negocio**

El cliente llama al negocio, la recepcionista atiende la llamada y le consulta sobre su necesidad, el cliente si no conoce el menú o lo que le gustaría consumir pregunta sobre el menú vigente, la recepcionista le dicta el menú, posteriormente el cliente elige según su preferencia, la recepcionista pregunta si está satisfecho con su pedido o si le gustaría agregar algo más, el cliente pide algo mas o solo confirma el pedido, la recepcionista solicita los datos personales del cliente como: nombre, número de teléfono y dirección, una vez obtenido los datos la recepcionista le comunica el monto total y pregunta si tiene cambio, el cliente le contesta y finaliza la llamada, el pedido es pasado de la recepción a la cocina en donde se prepara, luego de elaborar completamente el pedido, se le entrega al repartidor para enviarle al cliente su pedido y cobrar.

****

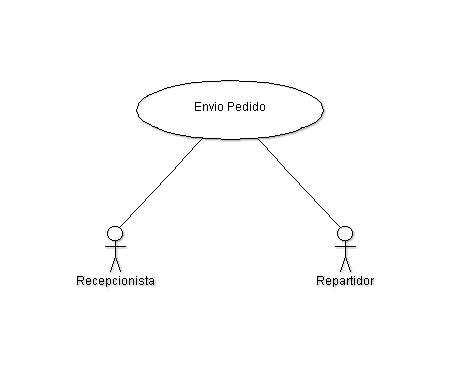
**Tabla de Caso de Uso**

|  |  |
| --- | --- |
| **Autores** | DeliverYapp. |
| **Actores:** | Cliente, Recepcionista, repartidor |
| **Fecha:** | 04-06-2016. |
| **Descripción:** | Proceso de Pedido de Delivery. |
| **Precondiciones** | El cliente debe contar con un teléfono, celular (con crédito) para la realización de llamada. |
| **Flujo normal** | |
| 1. El Cliente llama por teléfono al restaurante 2. La recepcionista contesta la llamada y pregunta que se le ofrece. 3. Si no conoce el menú le pregunta a la recepcionista. Si conoce el menú pasa al paso 6. 4. La recepcionista le dicta el menú. 5. El cliente realiza su pedido. 6. La recepcionista anota el pedido. 7. La recepcionista Pregunta si quiere agregar algo más a su pedido. 8. En caso de que el cliente diga “No” solo confirma su pedido. Si el cliente responde “Si” vuelve al paso 5. 9. La recepcionista solicita los datos personales al cliente(nombre, apellido, dirección, número teléfono) 10. La recepcionista comunica el tiempo estimado, monto total y pregunta si tiene cambio. 11. El cliente le contesta y finaliza la llamada. 12. La recepcionista pasa el pedido a la cocina 13. Una vez terminado el pedido, la recepcionista le entrega al repartidor para llevarle el pedido al cliente. 14. El repartidor entrega el pedido al cliente y cobra. | |
| **Flujo alternativo** | |
| 1. Si la recepcionista no contesta, el cliente vuelve a llamar.   En el caso de que se corte la llamada en cualquiera de los ítems del flujo normal se le devuelve la llamada.  En el caso de que el cliente vuelve a llamar para cancelar el pedido, este se da de baja.  13 En caso de que el pedido no llega, el cliente vuelve a llamar y el recepcionista busca la mejor solución. | |
| **Poscondiciones:** Se completa el servicio de delivery. | |

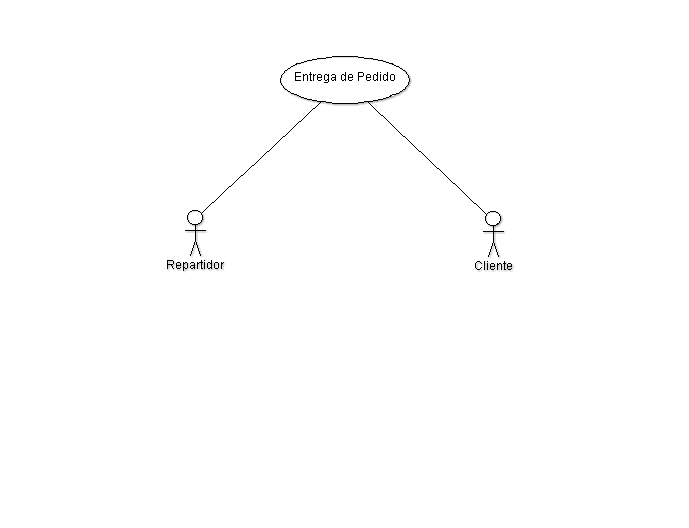
1.1 Realización de pedido

|  |  |
| --- | --- |
| **Autores** | DeliverYapp. |
| **Actores:** | Cliente, Recepcionista |
| **Fecha:** | 02-08-2016. |
| **Descripción:** | El cliente realiza el pedido. |
| **Precondiciones** | El cliente debe contar con un teléfono, celular (con crédito) para realizar la llamada. |
| **Flujo normal** | |
| 1. El cliente llama por teléfono al restaurante 2. La recepcionista contesta la llamada y pregunta que se le ofrece. 3. Si no conoce el menú le pregunta a la recepcionista. Si conoce el menú pasa al paso 6. 4. La recepcionista le dicta el menú. 5. El cliente realiza su pedido. 6. La recepcionista anota el pedido. 7. La recepcionista Pregunta si quiere agregar algo más a su pedido. 8. En caso de que el cliente diga “No” solo confirma su pedido. Si el cliente responde “Si” vuelve al paso 5. 9. La recepcionista solicita los datos personales al cliente(nombre, apellido, dirección, número teléfono) 10. La recepcionista comunica el tiempo estimado, monto total y pregunta si tiene cambio. 11. El cliente le contesta y finaliza la llamada. | |
| **Flujo alternativo** | |
| * 1. Si la recepcionista no contesta, el cliente vuelve a llamar.   2. Si la recepcionista no contesta, el cliente pasa por el local. | |
| **Postcondiciones:** Se generó pedido de delivery. | |

1.2 Envió de pedido



|  |  |
| --- | --- |
| **Autores** | DeliverYapp. |
| **Actores:** | Recepcionista, repartidor |
| **Fecha:** | 02-08-2016. |
| **Descripción:** | Proceso de envió de Delivery. |
| **Precondiciones** | La recepcionista debe contar con un pedido |
| **Flujo normal** | |
| 1. La recepcionista le entrega al repartidor el pedido, el monto a cobrar y la ubicación del cliente.   2 - El repartidor hace la entrega del pedido | |
| **Flujo alternativo** | |
| 1No se realiza la entrega de pedido | |
| **Postcondiciones:** Envio de pedido delivery | |

1.3 Entrega de Pedido

|  |  |
| --- | --- |
| **Autores** | DeliverYapp. |
| **Actores:** | Cliente, repartidor |
| **Fecha:** | 02-08-2016. |
| **Descripción:** | Proceso de entrega de Delivery. |
| **Precondiciones** | El cliente debe haber hecho un pedido. |
| **Flujo normal**   1. El repartidor recibe el pedido y se dirige a la ubicación indicada. 2. El repartidor llega a la ubicación indicada y hace la entrega del pedido al cliente. 3. El cliente le paga al repartidor. 4. El repartidor se retira. | |
| **Flujo alternativo** | |
|  | |
| **Postcondiciones:** Se hizo la entrega del pedido. | |

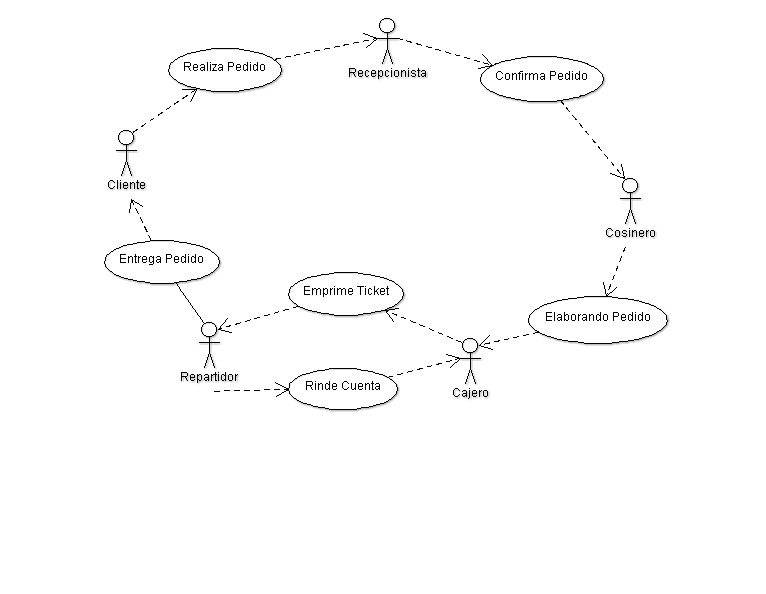
# Capítulo II

## Modelado de Requerimiento del Sistema

### Descripción Escrita del modelado del Sistema

El cliente desde su navegador ingresa a la aplicación “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. 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 mostrará un mensaje de envío correcto. Luego, la recepcionista recibe el pedido y lo verifica (si los datos son coherentes y correctos), en caso de que no se cumplan los requisitos el sistema notificara al cliente que su pedido fue rechazado y que lo intente nuevamente, al confirmar que el pedido cumple los requisitos, se notificará al cliente que su pedido fue aceptado, este ingresara a la cocina con estado pendiente, culminado el menú solicitado, pasa al cajero y cambia el estado a enviado imprimiendo un ticket con los datos del cliente necesarios para el envío, al retornar el repartidor rinde cuenta de los pedidos entregados el cajero ingresa el pago actualizando el estado ha cobrado.

**Grafico**



**Tabla de Caso de Uso General**

|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cliente, Recepcionista, Repartidor |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | Permite hacer el pedido de su restaurant preferido. |
| **Precondiciones** | El Cliente debe contar con un dispositivo con acceso a internet (celulares, computadoras, etc.) |
| **Flujo normal** | |
| 1. El cliente ingresa a la aplicación de DeliverYApp se registra, en caso de que este registrado 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 y se muestran los detalles (fotos, precios, etc.) 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. 7. Si el cliente está de acuerdo, presiona el botón “Confirmar Pedido”, en caso de que no esté de acuerdo, Cancela el pedido. 8. Se muestra un mensaje de confirmación “solicitud de pedido DeliveYApp Enviado “ 9. La recepcionista recibe la notificación de pedido del cliente y verifica los datos, una vez confirmado pasa el pedido a la cocina con el estado pendiente. 10. Culminado el menú solicitado, pasa a la recepcionista y a su vez cambia el estado ha enviado, imprimiendo un ticket con los datos del cliente, para que el repetidor realice el envío. 11. Al retornar, el repartidor rinde cuenta de los pedidos entregados, la recepcionista ingresa el pago actualizando el estado ha cobrado. | |
| **Flujo alternativo** | |
| 10- En caso de que no se haya impreso el ticket, se ingresa a la opción de imprimir el ticket de pedido en cuestión.  Si el cliente se queda sin acceso a internet, no podrá realizar su pedido | |
| **Precondiciones: Se realizó y se entregó el pedido de menú con éxito.** | |

1.1 Realiza Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cliente, Recepcionista |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cliente realiza pedido |
| **Precondiciones** | El Cliente debe contar con un dispositivo con acceso a internet (celulares, computadoras, etc.) |
| **Flujo normal** | |
| 1. El cliente ingresa a la aplicación de DeliverYApp se registra, en caso de que este registrado 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 y se muestran los detalles (fotos, precios, etc.) 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. 7. Si el cliente está de acuerdo, presiona el botón “Confirmar Pedido”, en caso de que no esté de acuerdo, Cancela el pedido. 8. Se muestra un mensaje de confirmación “solicitud de pedido DeliveYApp Enviado “ 9. La recepcionista recibe la notificación de pedido del cliente y verifica los datos | |
| **Flujo alternativo** | |
| Si el cliente se queda sin acceso a internet, no podrá realizar su pedido | |
| **Poscondiciones: Se realizó pedido de menú.** | |

1.2 Confirmación de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Recepcionista, Cocinero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | Confirma pedido |
| **Precondiciones** | La recepcionista debe recibir solicitud de pedido |
| **Flujo normal** | |
| 1. La recepcionista pasa el pedido a la cocina con el estado pendiente. | |
| **Flujo alternativo** | |
| 1 No se realiza el pedido. | |
| **Poscondiciones: Se realizó pedido de menú al cocinero.** | |

1.3 Elaboración de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cocinero, Cajero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cocinero ha elaborado todo el pedido |
| **Precondiciones** | El cocinero debe de tener el pedido. |
| **Flujo normal** | |
| 1. El cocinero elabora el pedido, pasa al cajero 2. El cajero a su vez cambia el estado ha enviado | |
| **Flujo alternativo** | |
|  | |
| **Poscondiciones: El pedido es entregado al cajero** | |

1.4 Impresión de Ticket



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cajero, Repartidor |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cajero imprime el ticket |
| **Precondiciones** | El menú debe de estar elaborado para el envió |
| **Flujo normal** | |
| 1. El cajero imprime un ticket con los datos del cliente. 2. El cajero entrega el ticket al repartidor. | |
| **Flujo alternativo** | |
| 1 De no entregarse el ticket el pedido no es entregado. | |
| **Poscondiciones: Se imprimió el ticket** | |

1.5 Entrega de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Repartidor, Cliente |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El repartidor entrega el pedido al cliente |
| **Precondiciones** | El repartidor debe tener el ticket. |
| **Flujo normal** | |
| 1. El repartidor llega a la dirección indicada por el cliente. 2. El repartidor entrega el pedido al cliente con el ticket 3. El cliente le paga el repartidor | |
| **Flujo alternativo** | |
| 1 El repartidor no llega y llama al cliente. | |
| **Poscondiciones: Se realizó la entrega del pedido al cliente** | |

1.6 El repartidor rinde cuenta



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Recepcionista, Cocinero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El repartidor rinde cuenta al cajero. |
| **Precondiciones** | El repartidor debe haber entregado el pedido. |
| **Flujo normal** | |
| 1. El repartidor retorna y rinde cuenta de los pedidos entregados. 2. La recepcionista ingresa el pago actualizando el estado ha cobrado. | |
| **Flujo alternativo** | |
| 2 - No se actualiza el estado. | |
| **Poscondiciones: El estado cambia de estado enviado a cobrado.** | |

* 1. Realiza Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cliente, Recepcionista |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cliente realiza pedido |
| **Precondiciones** | El Cliente debe contar con un dispositivo con acceso a internet (celulares, computadoras, etc.) |
| **Flujo normal** | |
| 1. El cliente ingresa a la aplicación de DeliverYApp se registra, en caso de que este registrado 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 y se muestran los detalles (fotos, precios, etc.) 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. 7. Si el cliente está de acuerdo, presiona el botón “Confirmar Pedido”, en caso de que no esté de acuerdo, Cancela el pedido. 8. Se muestra un mensaje de confirmación “solicitud de pedido DeliveYApp Enviado “ 9. La recepcionista recibe la notificación de pedido del cliente y verifica los datos | |
| **Flujo alternativo** | |
| Si el cliente se queda sin acceso a internet, no podrá realizar su pedido | |
| **Poscondiciones: Se realizó pedido de menú.** | |

1.2 Confirmación de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Recepcionista, Cocinero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | Confirma pedido |
| **Precondiciones** | La recepcionista debe recibir solicitud de pedido |
| **Flujo normal** | |
| 1. La recepcionista pasa el pedido a la cocina con el estado pendiente. | |
| **Flujo alternativo** | |
| 1 No se realiza el pedido. | |
| **Poscondiciones: Se realizó pedido de menú al cocinero.** | |

1.3 Elaboración de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cocinero, Cajero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cocinero ha elaborado todo el pedido |
| **Precondiciones** | El cocinero debe de tener el pedido. |
| **Flujo normal** | |
| 1. El cocinero elabora el pedido, pasa al cajero 2. El cajero a su vez cambia el estado ha enviado | |
| **Flujo alternativo** | |
|  | |
| **Poscondiciones: El pedido es entregado al cajero** | |

1.4 Impresión de Ticket



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Cajero, Repartidor |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El cajero imprime el ticket |
| **Precondiciones** | El menú debe de estar elaborado para el envió |
| **Flujo normal** | |
| 1. El cajero imprime un ticket con los datos del cliente. 2. El cajero entrega el ticket al repartidor. | |
| **Flujo alternativo** | |
| 1 De no entregarse el ticket el pedido no es entregado. | |
| **Poscondiciones: Se imprimió el ticket** | |

1.5 Entrega de Pedido



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Repartidor, Cliente |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El repartidor entrega el pedido al cliente |
| **Precondiciones** | El repartidor debe tener el ticket. |
| **Flujo normal** | |
| 1. El repartidor llega a la dirección indicada por el cliente. 2. El repartidor entrega el pedido al cliente con el ticket 3. El cliente le paga el repartidor | |
| **Flujo alternativo** | |
| 1 El repartidor no llega y llama al cliente. | |
| **Poscondiciones: Se realizó la entrega del pedido al cliente** | |

1.6 El repartidor rinde cuenta



|  |  |
| --- | --- |
| **Autores** | Grupo DeliverYApp |
| **Actores:** | Recepcionista, Cocinero |
| **Fecha:** | 26-05-2016 |
| **Descripción:** | El repartidor rinde cuenta al cajero. |
| **Precondiciones** | El repartidor debe haber entregado el pedido. |
| **Flujo normal** | |
| 1. El repartidor retorna y rinde cuenta de los pedidos entregados. 2. La recepcionista ingresa el pago actualizando el estado ha cobrado. | |
| **Flujo alternativo** | |
| 2 - No se actualiza el estado. | |
| **Poscondiciones: El estado cambia de estado enviado a cobrado.** | |

# Capítulo III

**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**

El desarrollo no representa un costo económico elevado, teniendo en cuenta que tenemos entorno de desarrollo gratuito y los demás gastos serán absorbidos por los alumnos.

* **Operativamente**

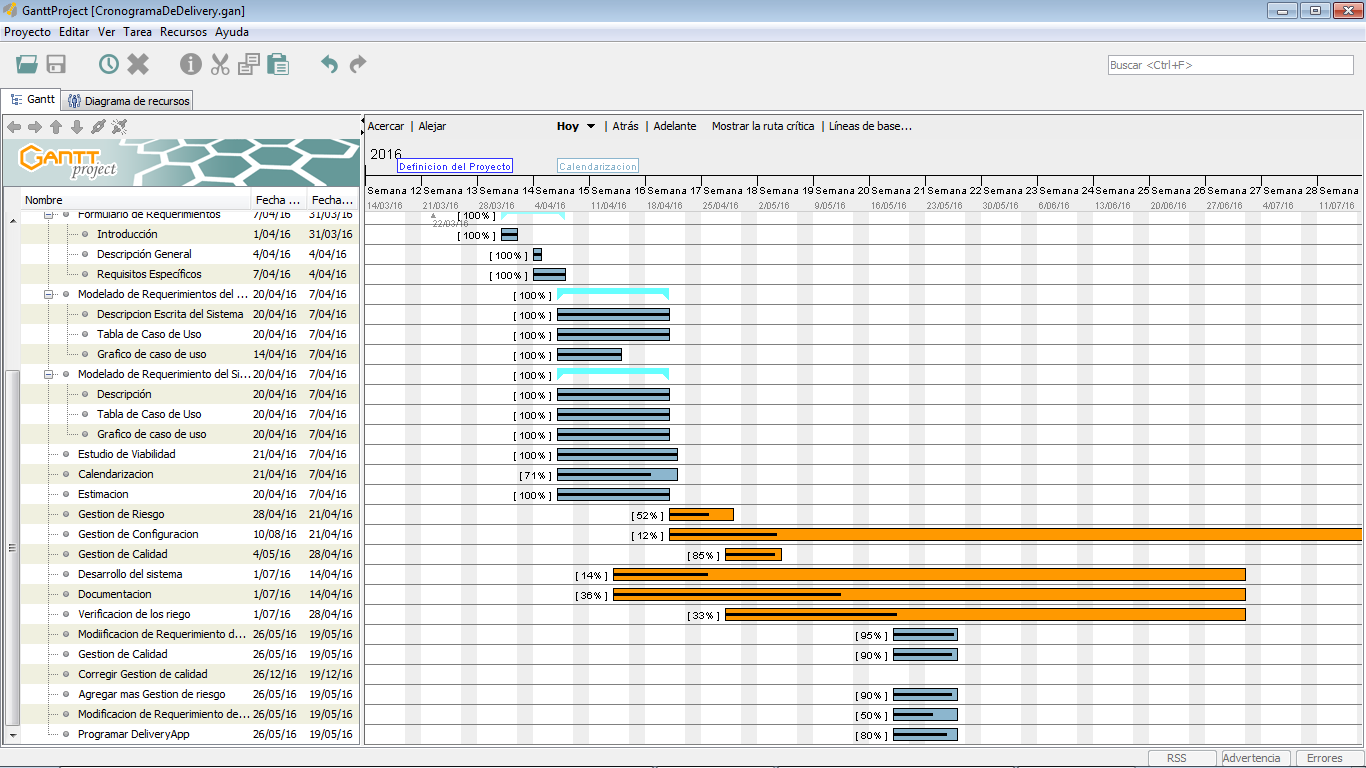
Cuenta con los recursos humanos preparados y la infraestructura necesarias para desarrollar dicho sistema.

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.

# Capítulo IV

## Calendarización





# Capítulo V

**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, Semi libre 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 VI

## Plan de riesgos

### Introducción

En la vida de un proyecto informático pueden provocarse errores por causas humanas, por tardar más en hacer una actividad que creíamos bien estimada, y por no terminar a tiempo un elemento que retrasa una integración con otros elementos, etc. El análisis de riesgo puede adelantarse a algunos problemas, y afirmarnos que si se produce su impacto en los objetivos de proyectos será menor posible.

El Plan de riesgo es una parte del requerimiento del sistema en el cual se plasman los riesgos y posibles soluciones a implementar si estas llegan a ocurrir en el transcurso de la elaboración del proyecto. Se obtiene una lista de los posibles riesgos, luego estas se clasifican y se separan de acuerdo al tipo.

### Propósito

El propósito del Plan de Riesgos es detectar, como dice ya el titulo los futuros riesgos que se podrían presentar durante la elaboración del proyecto y hacer de estas una lista para luego analizar y sacar las posibles soluciones para luego implementarlas si se da ocasión.

### Alcance

El alcance de este documento es fijar un plan de políticas a seguir por parte de los encargados del proyecto, para tener en cuenta ante la aparición de un problema. El proyecto DeliverYapp establece un plan para administrar los productos de trabajo del proyecto, incluyendo tanto los entregables de software como la documentación del proyecto.

**Gestión de Riesgos**

**Identificación de Riesgos**

|  |  |  |
| --- | --- | --- |
| **Riesgo** | **Tipo de riesgo** | **Descripción** |
| Rotación de personal | Proyecto y producto | Personal con experiencia abandona el proyecto antes de que finalice |
| Cambios de requisitos | Proyecto y producto | Existencia de más cambios de requerimientos de los previstos inicialmente |
| Retrasos en la especificación | Proyecto y producto | Retrasos en las especificaciones de interfaces esenciales |
| Subestimación del tamaño | Proyecto y producto | El tamaño del requisito (la ERS, del proceso de IR) se ha subestimado |
| Bajo rendimiento de la herramienta CASE | Producto | Las [herramientas CASE](http://www.monografias.com/trabajos14/herramicase/herramicase.shtml) que ayudan al proyecto no tienen el rendimiento y las funcionalidades esperadas |

### Análisis de Riesgos

|  |  |
| --- | --- |
| **Tipo de riesgo** | **Posibles riesgos** |
| Personal | Imposible contratar personal con los conocimientos requeridos. |
| Organizativos | La organización se reestructura y una nueva [administración](http://www.monografias.com/trabajos36/administracion-y-gerencia/administracion-y-gerencia.shtml) se responsabiliza del proyecto. |
| Herramientas | Las distintas [herramientas](http://www.monografias.com/trabajos11/contrest/contrest.shtml) CASE no están disponibles |
| Requerimientos | Cambios de requerimientos que precisan modificaciones en el [diseño](http://www.monografias.com/trabajos13/diseprod/diseprod.shtml). |
| Estimación | El tamaño del [sistema](http://www.monografias.com/trabajos11/teosis/teosis.shtml) a desarrollar está subestimado. |

**Riegos por efectos**

|  |  |  |
| --- | --- | --- |
| **Riesgo** | **Probabilidad** | **Efectos** |
| Problemas financieros de la organización reducen el presupuesto del proyecto | baja | Catastrófico |
| Imposible contratar personal con los conocimientos requeridos | alta | Catastrófico |
| Personal clave enfermo o no disponible en momentos críticos | moderada | Serio |
| Cambios de requerimientos que precisan modificaciones en la codificación | moderada | Serio |
| El tiempo requerido para desarrollar el proceso de IR está subestimado | alta | Serio |
| Los clientes no comprenden el impacto de los cambios en los requerimientos | moderada | Tolerable |

**Estrategias por riesgos.**

|  |  |
| --- | --- |
| **Riesgo** | **Estrategia** |
| Problemas financieros de la organización | Preparar un documento breve para la [dirección](http://www.monografias.com/trabajos15/direccion/direccion.shtml) de la [empresa](http://www.monografias.com/trabajos11/empre/empre.shtml) que [muestra](http://www.monografias.com/trabajos11/tebas/tebas.shtml) que el proyecto hace contribuciones muy importantes a las metas del negocio |
| Problemas de reclutamiento | Organizar cursos de [capacitación](http://www.monografias.com/trabajos/adpreclu/adpreclu.shtml) para el personal ya existente, investigar la posibilidad de contratar en otras regiones del país |
| Enfermedad del personal | reorganizar el equipo de tal forma que se solapen el trabajo y los miembros comprendan [el trabajo](http://www.monografias.com/trabajos/fintrabajo/fintrabajo.shtml) de los demás |
| Cambios en los requisitos | Rastrear la información para valorar el impacto de los requerimientos, maximizar la información oculta en ellos |
| Tiempo de IR subestimado | Alertar al cliente de las dificultades potenciales y las posibilidades de retraso |

# Capítulo VII

Plan de Gestión de Configuración

Introducción

Planificación de la configuración:

Se describen las actividades de gestión de configuración de software que deben ser ejecutadas durante el proceso de desarrollo del proyecto. Se definen tanto los proyectos que se pondrán bajo control de configuración como los procedimientos que deben ser seguidos por los integrantes del equipo de trabajo.

**Definición de la Línea Base:**

Informar sobre la verificación realizada sobre los objetos de configuración que componen la versión actual para asegurar que se encuentran en estado consistente en la Línea Base del Proyecto.

**Seguimiento de la Línea Base**

Recibir, registrar y mantener todos los productos recibidos a través de todas sus versiones.

**Control de Cambios:**

Seguimiento de un ciclo de vida de un cambio.

**Informe Final de Configuración:**

Realizar el Informe Final de Configuración, el cual contiene un resumen de las actividades realizadas de SCM y sus resultados a lo largo del proyecto.

**Producir la Versión de Producto a Liberar:**

Realizar el Informe Final de Configuración, el cual contiene un resumen de las actividades realizadas de SCM y sus resultados a lo largo del proyecto.

Propósito

Este documento describe las actividades de gestión de configuración de software que deben ser llevadas a cabo durante el proceso de desarrollo del proyecto. Aquí se definen tanto los productos que se pondrán bajo control de configuración como los procedimientos que deben ser seguidos por los integrantes del equipo de trabajo.

Alcance

El ámbito de este documento es el proyecto DeliverYapp y establece un plan para administrar los productos de trabajo del proyecto, incluyendo tanto los entregables de software como la documentación del proyecto.

Gestión de Configuración

Las responsabilidades son:

* Planificación de Configuración
* Definir la Línea Base
* Seguimiento de la Línea Base
* Control de Cambios
* Realización del Informe Final de Configuración
* Producir la Versión de Producto a Liberar

Herramientas, Entorno e Infraestructura

Herramientas

Durante el proceso de gestión de configuración se utilizará la herramienta Github para el control de versiones del proyecto. Esta gestión se hará mediante la herramienta Zenhub para los documentos y Sublime para el código fuente.

El control de cambios, así como la gestión de defectos, se llevará a cabo mediante la plantilla “Solicitud de Cambio”, permitiendo al equipo de desarrollo ir detallando los defectos encontrados para que la persona responsable pueda corregirlos.

Programa de la Gestión de Configuración

Identificación de la Configuración

Elementos de Configuración

Para este proyecto los elementos de configuración se corresponderán con los entregables definidos en el Modelo de Proceso, aunque no necesariamente todos los entregables deben ser elementos de configuración.

La decisión de cuál de los entregables serán elementos de configuración será tomada por el SCMR, quién deberá tomar en cuenta qué productos serán necesarios cuando se quiera recuperar una versión completa del sistema.

Se debe generar una línea base por iteración en cada Fase, de acuerdo a lo siguiente:

* Los eventos que dan origen a la línea base.
* Los elementos que serán controlados en la línea base.
* Los procedimientos usados para establecer y cambiar la línea base.
* La autorización requerida para aprobar cambios a los documentos de la línea base.

Nomenclatura de Elementos

Se especifica cómo se distinguirán las diferentes versiones de cada elemento.

Para todos los elementos de configuración se les deberá agregar, después del nombre del mismo, información acerca del grupo al que corresponde el elemento y la versión del mismo.

El formato para esta nomenclatura es: **NomenclaturaG**X**v**Y.extensión, donde:

· Nomenclatura es la especificada más abajo para cada elemento.

· X es un número de 1 dígito que identifica al grupo.

· Y indica la versión del elemento de configuración o entregable.

· Extensión indica la extensión del elemento de configuración o entregable.

Para los entregables, se deberá identificar a que Fase e iteración corresponden en forma manual. Se indica la siguiente nomenclatura para cada entregable en el modelo de proceso, según la disciplina.

**Requerimientos:**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| RQACT | Acta de Reunión de Requerimientos |
| RQDRQ | Especificación de Requerimientos |
| RQMOD | Modelo de Casos de Uso |
| RQRSU | Requerimientos Suplementarios |
| RQDVC | Documento de Validación con el Cliente |
| RQPIU | Pautas para Interfaces de Usuario |
| RQRCA | Requerimientos Candidatos |
| RQALS | Alcance del Sistema |
| RQGLO | Glosario |
| RQOOMDO | Modelo de Dominio |
| RQOODRP | Documento de Requerimientos para el Prototipo |
| RQGXNOM | Nomenclatura |

**Diseño:**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| DSMDI | Modelo de Diseño |
| DSARQ | Descripción de la Arquitectura |
| DSOOMDA | Modelo de Datos |
| DSOODDP | Documento de Diseño del Prototipo |

**Implementación:**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| IMEDT | Estándar de Documentación Técnica |
| IMEI | Estándar de Implementación |
| IMPR | Prototipo |
| IMIIN | Informe de Integración |
| IMDT | Documentación técnica |
| IMIVU | Informe de Verificación Unitaria |
| IMOOPII | Plan de Integración de la Iteración |
| IMOOMIM | Modelo de Implementación |
| IMOOEJI | Ejecutable de la Iteración |
| IMOORRP | Reporte de Revisión por Pares |
| IMOOCVU | Clases de la Verificación Unitaria de Módulo |
| IMGXICO | Informe de Consolidación |
| IMGXEST | BC Con Estilos |
| IMGXCON | BC Consolidado |
| IMGXNUC | BC Núcleo |
| IMGXMOD | BC Módulo |

**Verificación:**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| VRPVV | Plan de Verificación y Validación |
| VRDAP | Documento de Evaluación y Ajuste del Plan de V & V |
| VRPVI | Plan de Verificación de la Iteración |
| VRMCP | Modelo de Casos de Prueba |
| VRIVD | Informe de Verificación de Documento |
| VRIVI | Informe de Verificación de Integración |
| VRIVS | Informe de Verificación del Sistema |
| VRRPR | Reportes de Pruebas |
| VREV | Evaluación de la Verificación |
| VRIFV | Informe Final de Verificación |

**Implantación (IP):**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| IPMSU | Materiales para Soporte al Usuario  (Se pueden usar sufijos para identificar cada ítem dentro del material Ej. IPMSUMU para Manual de Usuario) |
| IPMCA | Materiales para Capacitación |
| IPPS | Presentación del Sistema |
| IPPLA | Plan de Implantación |
| IPVPR | Versión del Producto |
| IPOOEDU | Estándar de Documentación de Usuario |
| IPOORFPA | Reporte Final de Pruebas de Aceptación |

**Gestión de Configuración y Control de Cambios (SCM):**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| SCMPLA | Plan de Configuración |
| SCMMAC | Manejo del Ambiente Controlado |
| SCMGC | Gestión de Cambios |
| SCMRV | Registro de Versiones |
| SCMILB | Informe de la Línea Base del Proyecto |
| SCMIF | Informe Final de SCM |

**Gestión de Calidad (SQA):**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| SQAPLA | Plan de Calidad |
| SQADAP | Documento de Evaluación y Ajuste del Plan de Calidad |
| SQARTF | Informe de RTF |
| SQAES | Entrega Semanal de SQA |
| SQAIR | Informe de Revisión de SQA |
| SQADV | Descripción de la Versión |
| SQANV | Notas de la Versión |
| SQAIF | Informe Final de SQA |

**Gestión de Proyecto (GP):**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| GPPLA | Plan de Proyecto |
| GPISP | Informe de Situación del Proyecto |
| GPEM | Estimaciones y Mediciones |
| GPDRI | Documento de Riesgos |
| GPRAC | Registro de Actividades |
| GPIFP | Informe Final de Proyecto |
| GPARE | Acta de la Reunión de Equipo |
| GPPIT | Plan de la Iteración |
| GPPDE | Plan de Desarrollo |
| GPICF | Informe de Conclusiones de la Fase |
| GPPDP | Presentación al Director del Proyecto |
| GPARD | Acta de la Reunión con el Director del Proyecto |
| GPOODAP | Documento de Evaluación y Ajuste al Plan de Proyecto |
| GPIARI | Acta de la Reunión de Integración |

**Comunicación (COM):**

|  |  |
| --- | --- |
| Nomenclatura | Entregable |
| COMDI | Documento Informativo |
| COMENS | Encuesta de Satisfacción del Cliente |
| COMEVS | Evaluación de Satisfacción del Cliente |

Elementos de la Línea Base del Proyecto

N/A

Control de Configuración

El procedimiento que se describe a continuación es el que se utilizará cada vez que se precise introducir un cambio al sistema.

Se entiende por cambio al sistema, las modificaciones que afecten a la línea base del sistema, como pueden ser:

* Cambios en los Requerimientos.
* Cambios en el Diseño.
* Cambios en la Arquitectura.
* Cambios en las herramientas de desarrollo.
* Cambios en la documentación del proyecto.

Solicitud de Cambios

Cuando se realiza la solicitud de un cambio, se actualiza el documento de “Solicitud de cambio” para registrar esta solicitud.

Aprobación de Cambios

Se debe formar el “Comité de Control de Configuración” y determinar su autoridad para la aprobación de cambios.

La composición de este comité puede variar según el tipo de cambio y las líneas de trabajo involucradas en él.

Se sugieren como posibles integrantes:

* Administrador (obligatorio)
* Arquitecto (opcional)
* Analista (opcional)
* Implementador (opcional)
* SCM (obligatorio)
* Cliente (opcional)

Se define un comité de Control de Configuración de nivel superior, compuesto por el Gerente de proyecto, al cual se elevarán las solicitudes de cambios cuya aprobación o desaprobación no se pueda resolver por el primer comité.

Implementación de Cambios

Una vez realizada la evaluación del cambio, se decide en qué momento implementarlo. Esta etapa involucra los procesos necesarios para implementar la solicitud y monitorear el progreso del trabajo.

Además se especificará el momento de liberación del cambio; así como también los responsables de las actividades que involucra el cambio.

Estado de la Configuración

N/A

Informes y Auditorías

Cada cierto tiempo, el gestor de configuración y cambios realizará un informe para el jefe de proyecto con el fin de revisar la evolución de los defectos que se vayan registrando en el mismo.

En este informe (“Informe de Evolución de Defectos”) se detallará los defectos detectados, sus prioridades, responsables de corregirlos, su estado y el procedimiento que se ha seguido o se va a seguir a la hora de resolverlos.

Este nos permitirá obtener una serie de indicadores para determinar la calidad del producto que se está desarrollando, permitiendo al jefe de proyecto informar al cliente sobre este aspecto.

Las auditorías que se llevarán a cabo para comprobar si los cambios se han realizado correctamente serán al final de cada iteración, pero antes de que se cree una línea base. En éstas se revisarán tanto los requisitos funcionales y de rendimiento, como que el producto cumpla con las especificaciones detalladas en las que se define. Tomarán parte en éstas el cliente, el jefe de proyecto y el gestor de configuración.

Calendario

La definición de la línea base será determinada al comienzo de cada semana de la siguiente manera: https://github.com/francben/proyectoDeliberyApp#burndown?milestoneId=x, donde la X corresponde a la semana corriente.

Todavía no se cuenta con una implementación del Control de Cambios.

Capacitación y Recursos

Para el manejo de las versiones se utiliza Git.

Mantenimiento del Plan de Gestión de la Configuración

El responsable de monitorear el Plan de SCM es el responsable de SCM.

Para la frecuencia de las modificaciones no hay un tiempo estipulado, serán las necesarias, siempre limitadas al tiempo de finalización de la iteración.

Este Plan deberá ser revisado al inicio de cada fase, modificado de acuerdo a lo necesario, aprobado y distribuido al equipo de proyecto.

# Capítulo VIII

## 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.

# Capítulo IV

Plan de SQA

1. Propósito

El propósito del documento es determinar el correcto funcionamiento del software y la calidad del mismo.

Concordancia con los requisitos funcionales y de rendimientos explícitamente establecidos, y con los estándares de desarrollo explícitamente documentados, y con las características que se espera de todo software desarrollado profesionalmente.

* Prácticas de Aseguramiento de la calidad:

Para el buen funcionamiento del sistema se utilizó la metodología RUP con el tipo de proceso espiral y para las revisiones se ha utilizado la herramienta colaborativa GibHub.

* Evaluación de requerimientos:

El requerimiento se realizó en base a las necesidades del Cliente, según la entrevista realizada.

* Evaluación del diseño: fue confeccionado de acuerdo al requerimiento hecho en la entrevista al cliente
* Evaluación de la codificación: Controlar que se cumpla con los estándares de codificación y evaluar la correctitud técnica del producto.
* Evaluación de los procesos de integración y pruebas: Controlar que se esté cumpliendo con el Plan de Testing.
* Recolección de métricas.

1. Referencias

[ANSI/IEEE Std 730.1-1989, IEEE Standard for Software Quality Assurance

Documento plan de gestión de riesgos

Documento plantilla de Revisión técnica formal RTF]

1. Gestión

Se trata de un grupo de alumnos dentro de la materia ingeniería del software, y las actividades fueron concierne a lo que contempla la metodología RUP, rotando los roles a lo largo del desarrollo del producto.

* 1. Organización

En nuestra organización debido a fines educativos fuimos rotando de roles entre las cuales está el líder, testing, desarrollador, analista de riesgo, documentador, gestor de configuración. Y a pesar de que cada persona involucrada en el proyecto debería chequear y revisar su propio trabajo como parte del mismo, el líder del proyecto es el máximo responsable de la calidad del proyecto como un todo.

* 1. Actividades
     1. Ciclo de vida del software cubierto por el Plan

Planeación (estimación, programación, análisis de riesgo): está en constante modificación de acuerdo al avance del sistema.

Modelado (análisis, diseño)

Etapa de Construcción (Código prueba): el software está en proceso de desarrollo con sus respectivas verificaciones para el control de calidad.

Despliegue (entrega, retroalimentación): este será uno de los puntos que se tendrá en cuenta ara las revisiones de calidad.

Comunicación

* + 1. Actividades de calidad a realizarse

El responsable de detectar e informar los defectos del producto es el testeador del sistema

1. Este numeral se evalúa con la Revisión del Manual de Calidad.

2. ¿Cuáles son los procesos de la organización?

3. ¿Se encuentran identificados los procesos del sistema?

4. ¿Se ha determinado la secuencia e interacción de estos procesos?

5. ¿Existe mapa o diagrama de procesos, que represente la forma como se realizan los productos o servicios?

6. ¿Se determinan los métodos y criterios requeridos para asegurar que la operación como el control de los procesos sean eficaces?

7. ¿La organización, asegura la disponibilidad de recursos e información necesarios para apoyar el funcionamiento y el seguimiento de los procesos?

8. ¿La organización, mide, realiza el seguimiento y analiza estos procesos?

9. ¿La organización ha implantado las acciones necesarias para alcanzar los resultados previstos y la Mejora continua de estos procesos?

En el caso que la organización tenga contratado externamente cualquier proceso que afecte a la conformidad del producto con los requisitos, aplican entre otras las siguientes preguntas.

10. ¿Cuáles procesos son contratados con terceros?

11. ¿Cómo se gestionan esos procesos?

12. ¿Cómo se controlan los procesos subcontratados externamente?

13. ¿El control de dichos procesos contratados externamente están identificados en el sistema de gestión de la calidad?

14. ¿El control se ejerce a través de documentos relacionados con el numeral 7.4 Compras? (por ejemplo: especificaciones técnicas)

* + 1. Revisar cada producto

1. Funcionalidad del Menú Principal del Sistema:

* Posee el acceso a todos los lugares necesarios.
* Está correctamente diseñado.

1. Interfaz restaurant(Según la asignación que le corresponda):

* Cumple con todos los campos de acuerdo a la información relevada.
* Guarda y modifica la información del restaurante.
* El diseño cumple con los estándares.
* El proceso de utilización de la interfaz es intuitivo.
* Las reglas de negocio están bien definidas.

1. Interfaz Plato:

* Cumple con todos los campos de acuerdo a la información relevada.
* Guarda y modifica la información del plato
* El diseño cumple con los estándares.
* El proceso de utilización de la interfaz es intuitivo.
* Las reglas de negocio están bien definidas.

1. Interfaz Cliente:

* Cumple con todos los campos de acuerdo a la información relevada.
* Guarda y modifica la información del plato
* El diseño cumple con los estándares.
* El proceso de utilización de la interfaz es intuitivo.
* Las reglas de negocio están bien definidas.

1. Interfaz de Pedido:

* Cumple con los estándares de diseño aprendidos.
* Es correcto y homogéneo el diseño con respecto al todo.
* Es fácil de utilizar, predecible, intuitivo.
* Utiliza validaciones.
* Contiene los componentes necesarios para ejecutar la transacción.
* Correcto funcionamiento de la reglas de negocio.
* Informa en transacciones correctas o exitosas.
* Permite imprimir una factura.

* + 1. Realizar Revisión Técnica Formal (RTF)

|  |  |
| --- | --- |
| **Participante** | **Rol** |
| Francisco Benítez | Líder |
| Ricardo Maciel | Desarrollador |
| Sebastián Kazlauskas | Documentador |
| Myriam Medina | Analista De Riesgo |
| Melanie Florentín | Encargada de Configuración |
| Orlando Cardozo | Testeador |

Técnica utilizada

Se utilizó la checklist teniendo en cuenta el análisis y diseño del sistema elaborados en la metodología RUP.

Problemas detectados

Problema 1

Al comienzo del desarrollo fueron mal elaborados los casos de uso del negocio primeramente, por lo cual este también era necesario para después hacer el del sistema.

Sugerencia de corrección

Actualizar el documento de forma a cubrir las falencias que tenían los casos de uso.

Problema 2

La elaboración de los Diagramas nos llevó más tiempo de lo pensado ya que carecíamos de conocimiento de las herramientas utilizadas para el efecto.

Sugerencia de corrección

Realizar todos los diseños de los diagramas para que se tenga un panorama de como funcionaria el sistema.

Evaluación

El sistema es implementable, poseen atributos de calidad en todas sus funcionalidades.

Sin embargo se constató que ciertas propiedades no estaban reflejadas en el documento

Estado actual del Producto

Tanto el documento sobre la descripción de la aplicación y el diseño deben completarse en base a los diferentes diagramas elaborados.

Acciones a tomar

El documento será actualizado tomando en cuenta lo discutido en la reunión ya que todos los integrantes aportaron ideas.

Próxima Revisión del Producto

Se hará una próxima revisión antes de seguir el desarrollo.

* + 1. Relaciones entre las actividades de SQA y la planificación

|  |  |
| --- | --- |
| **Actividad** | **Semana** |
| Definición del proyecto | Semana 1 |
| Modificación Modelado de Negocio | Semana 2 |
| Modificación de Modelado del Sistema | Semana 2 |
| Modificación de Diagramas | Semana 3 |
| Modificación de Modelado de Requerimiento | Semana 3 y4 |
| Elaboración de Plan de Riesgo | Semana 5 |
| Elaboración de Plan de Configuración | Semana 6 |
| Modificación de Plan de Calidad | Semana 6 y 7 |

* 1. Responsables

Como los roles son cambiantes las actividades no tenían un responsable permanente, si no que era constante de forma que las actividades no eran específicas de cada persona.

Para la puesta en marcha de estas actividades se deberá seguir el siguiente ciclo de prevención:

* Ejecutar una tarea
* Realizar un control de revisiones, para decidir la aceptación o necesidad de corrección de dicha tarea.
* En caso de que en la revisión se presenten errores se realizara un análisis causal para determinar el motivo de estos. Se analiza un determinado error, se establece una hipótesis de su posible causa, se trata de deducir en qué momento se produjo y por qué. Luego se deberá realizar la corrección del mismo y tomar una acción correctiva con el fin de eliminar la causa del problema.
* El resultado del análisis causal es ingresado a una base de datos para mantener un registro y poder obtener métricas.
* Se comienza nuevamente el ciclo ejecutando la tarea.

1. Documentación
   1. Propósito

Documentar el proceso en el cual se va desarrollando el Sistema de acuerdo al modelado de requerimiento, si en el mismo se encuentran problemas el siguiente paso es anotar esos errores con sus definiciones y la corrección que se le dio de acuerdo al Plan de Configuración y teniendo en cuenta los posibles riesgos citados en el Plan de Riesgo.

* 1. Documentación mínima requerida
     1. Especificación de requerimientos del software

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.
9. **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, etc.).
* 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.

**Funcionalidad**

Adecuación a las necesidades

Precisión de los resultados

Seguridad de los datos

**Confiabilidad**

Madurez

Tolerancia a faltas

Recuperabilidad

**Usabilidad**

Comprensible

Aprendible

Operable

Atractivo

**Eficiencia**

Comportamiento respecto al tiempo

Utilización de recursos

**Mantenibilidad**

Analizable

Modificable

Estable, no se producen efectos inesperados luego de modificaciones

Verificable

**Portabilidad**

Adaptable

Instalable

Co-existencia

Reemplazante

* + 1. Descripción del diseño del software

Login de administrador y usuario

Creación y asignación de roles

Crear menús

Manejo de pedidos (recepción, cocina y caja )

Login y Registro del cliente

Menú de Restaurante

Menú comida

Formulario de Pedido

* + 1. Plan de Verificación & Validación

Verificar que las vistas del sistema cumplan con el requerimiento citado en el documento del diseño, una vez verificado se procede a la validación de los mismos.

1. Estándares, prácticas, convenciones y métricas 
   1. Estándar de documentación

Como estándares de documentación se definirá el siguiente documento:

* Estándar de documentación técnica

La documentación técnica del producto debe:

* Ser adecuada para que un grupo independiente del desarrollo pueda encarar el mantenimiento del producto.
* Incluir fuentes, Modelos de Casos de Uso, Objetos de diseño.

Para la escritura de documentos se han definido plantillas para ser utilizadas en la elaboración de entregables.

En estas plantillas se definen:

* Encabezado y pie de página.
* Fuente y tamaño de fuente para estilo normal.
* Fuente y tamaño de fuente para los títulos a utilizar.
* Datos mínimos que se deben incluir: fecha, versión y responsables.
  1. Estándar de verificación y prácticas

Se utilizan las prácticas definidas en el Plan de Verificación y Validación.

Como estándar se utiliza el documento de:Std 1012-1986 IEEE Standard for Software Verification and Validation Plans.

1. Revisiones y auditorías
   1. Objetivo

Las auditorías de calidad son aquellas en las que se evalúa la eficacia del sistema de gestión de calidad de la organización. Y las revisiones consisten en hacer un seguimiento de los ítems de los requerimientos para seguir el proceso de desarrollo.

* 1. Requerimientos mínimos
     1. Revisión de requerimientos

Se especifican las revisiones y auditorías que deben realizarse como mínimo, así como la agenda para la realización de las mismas.

* + 1. Revisión de diseño preliminar

Esta revisión se realiza para asegurar que se ha cumplido con los requerimientos especificados por el Cliente.

* + 1. Revisión de diseño crítico

Esta revisión se realiza para asegurar la consistencia del diseño detallado con la especificación de requerimientos.

* + 1. Auditoria Funcional

Esta auditoría se realiza previa a la liberación del software, para verificar que todos los requerimientos especificados en el documento de requerimientos fueron cumplidos.

* + 1. Auditoria Física

Esta revisión se realiza para verificar que el software y la documentación son consistentes y están aptos para la liberación.

* + 1. Auditorías internas al proceso

Estas auditorías sirven para verificar la consistencia: del código versus el documento de diseño, especificaciones de interfases, implementaciones de diseño versus requerimientos funcionales, requerimientos funcionales versus descripciones de testeo.

* + 1. Revisiones de gestión

Estas revisiones se realizan periódicamente para asegurar la ejecución de todas las actividades identificadas en este Plan. Deben realizarse por una persona ajena al grupo de trabajo.

* + 1. Revisión del Plan de gestión de configuración

Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de gestión de configuración.

* + 1. Revisión Post Mortem

Esta revisión se realiza al concluir el proyecto para especificar las actividades de desarrollo implementadas durante el proyecto y para proveer recomendaciones.

* + 1. Agenda

En esta sección se deberá especificar la agenda para las revisiones y auditorías detalladas anteriormente.

1. Verificación

Se debe identificar todas las verificaciones que no fueron identificadas en el Plan de V & V para el software y debe especificar los métodos a ser usados.

1. Reporte de problemas y acciones correctivas

Esta sección debe incluir: Descripción de las prácticas y procedimientos que se seguirán para el reporte, seguimiento, y resolución de los problemas surgidos en el desarrollo de software; especificar los responsables comprometidos con la implementación de estas acciones correctivas.

1. Herramientas, técnicas y metodologías

Se deben identificar herramientas de software, técnicas, y metodologías de soporte para las actividades de aseguramiento de calidad.

1. Gestión de riesgos

Se deben especificar los métodos y procedimientos utilizados para especificar, monitorear, y controlar las áreas de riesgo durante el proyecto.

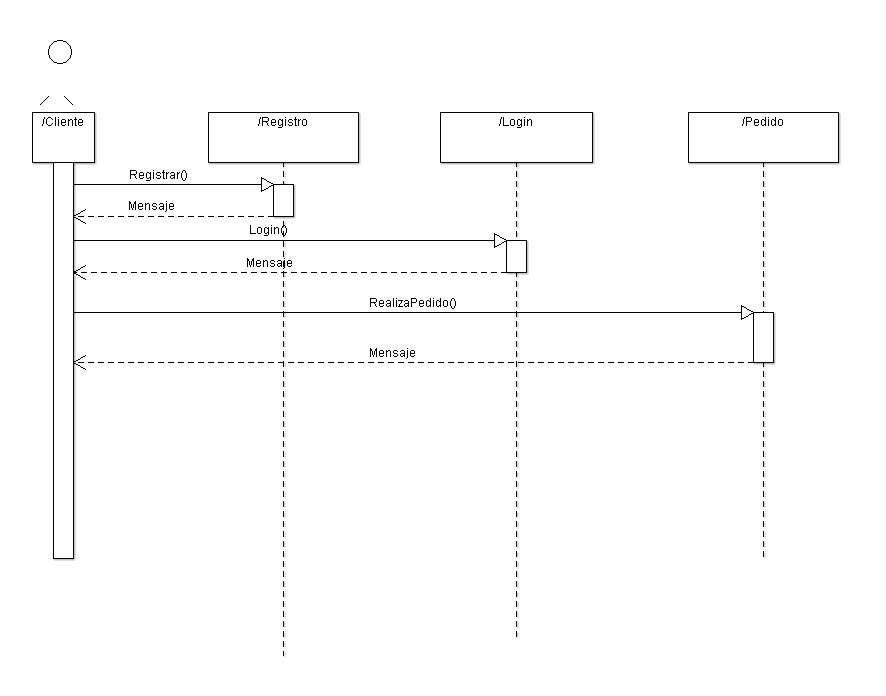
Los riesgos identificados, la estrategia de mitigación, monitoreo y plan de contingencia a ser llevados a cabo, serán descritos en el Documento de Gestión de Riesgos, con lo cual se podrá hacer referencia a él.

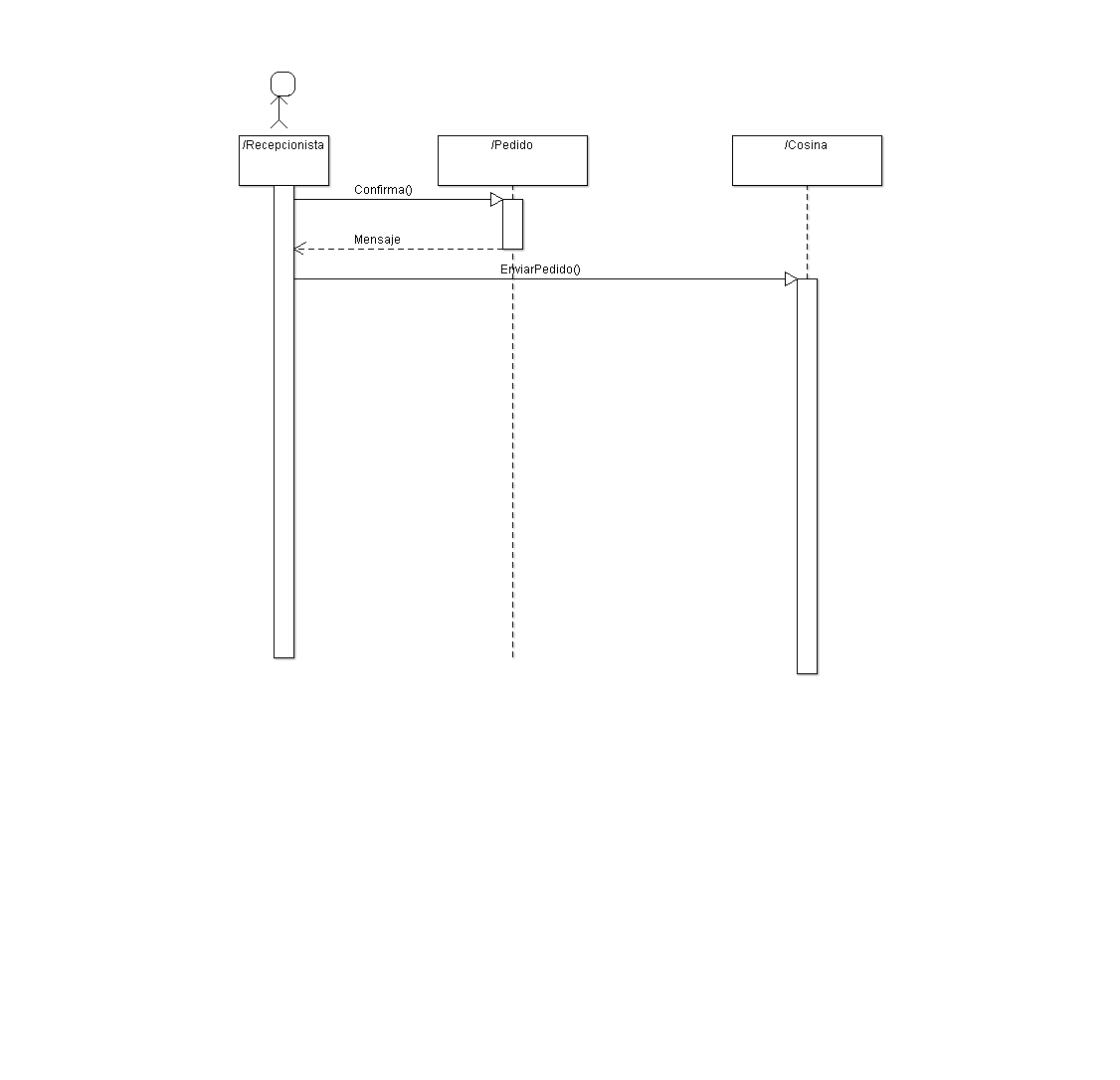
Anexos

Formulario de Pedidos y Detección de Cambios

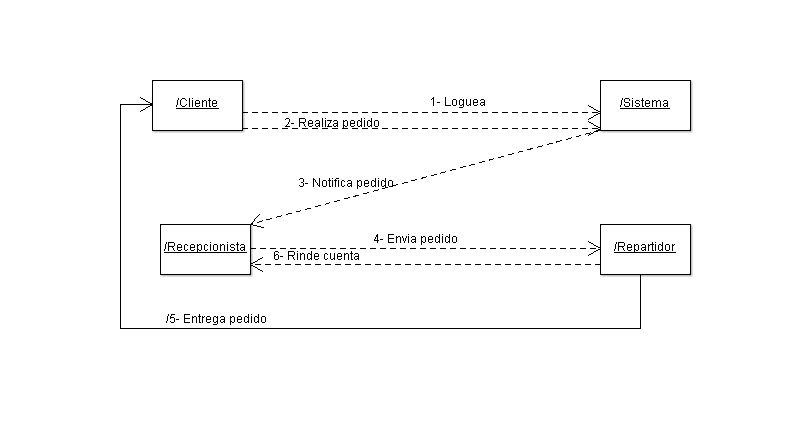
|  |  |
| --- | --- |
| Formulario de Pedidos y Detección de Cambios | |
| Fecha de Petición: |  |
| Nombre y Versión del Elemento |  |
| Nombre del Solicitante: |  |
| Necesidad de Cambio: |  |
| Descripción del cambio pedido: |  |
| Prioridad: |  |
| Estado: |  |
| Fecha del cambio: |  |
| Identificador de la nueva versión: |  |
| Que fue afectado por este cambio |  |

**Diagrama de Secuencia**

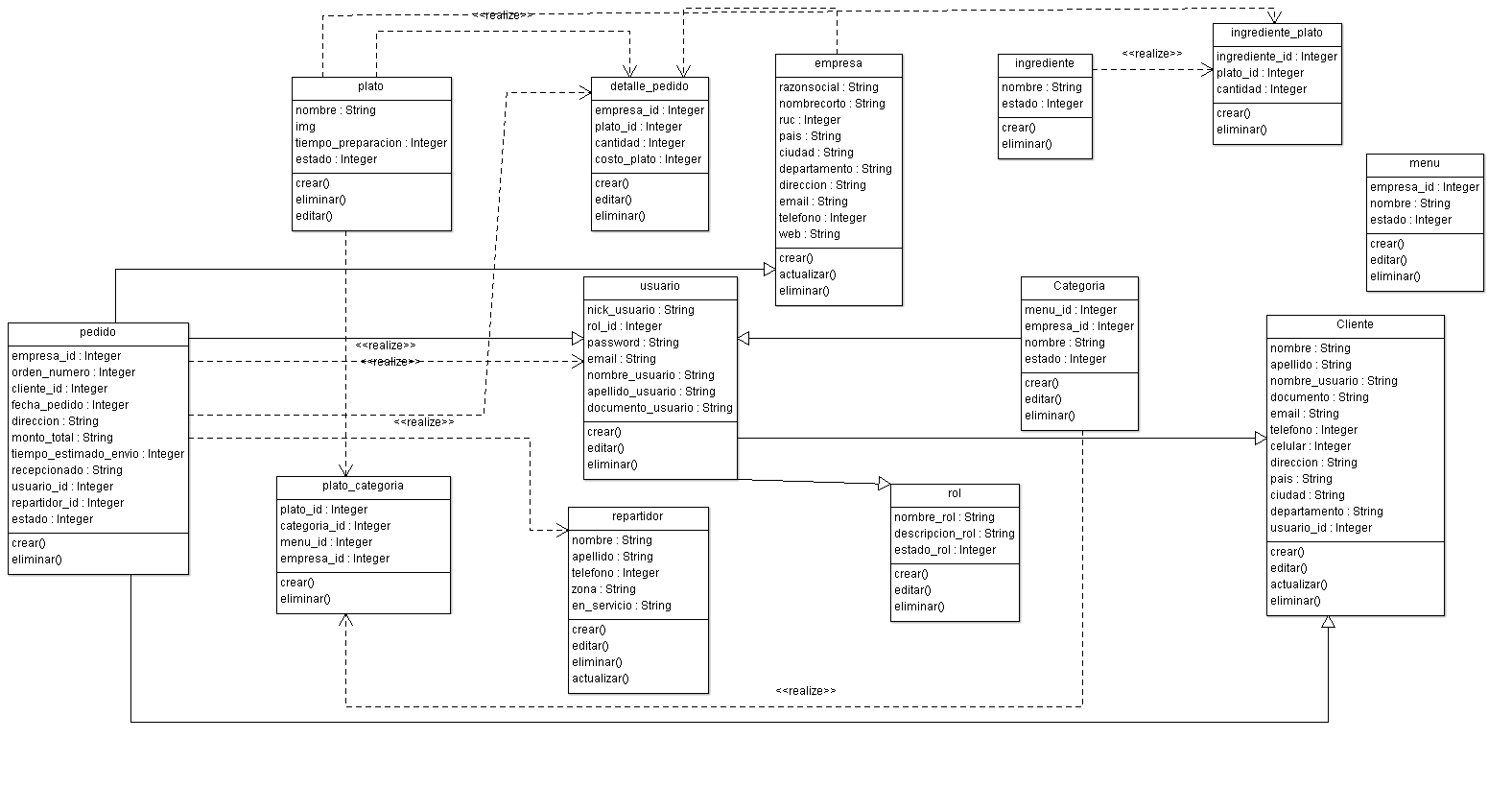
****

****

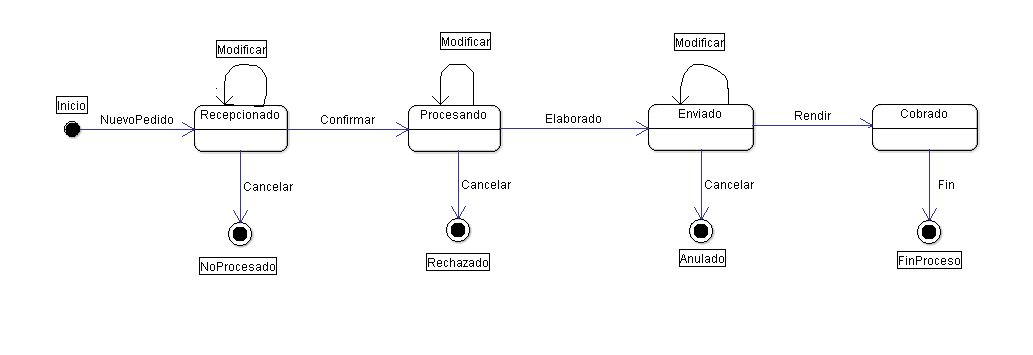
**Diagrama de Colaboración**

****

**Diagrama de Clases**

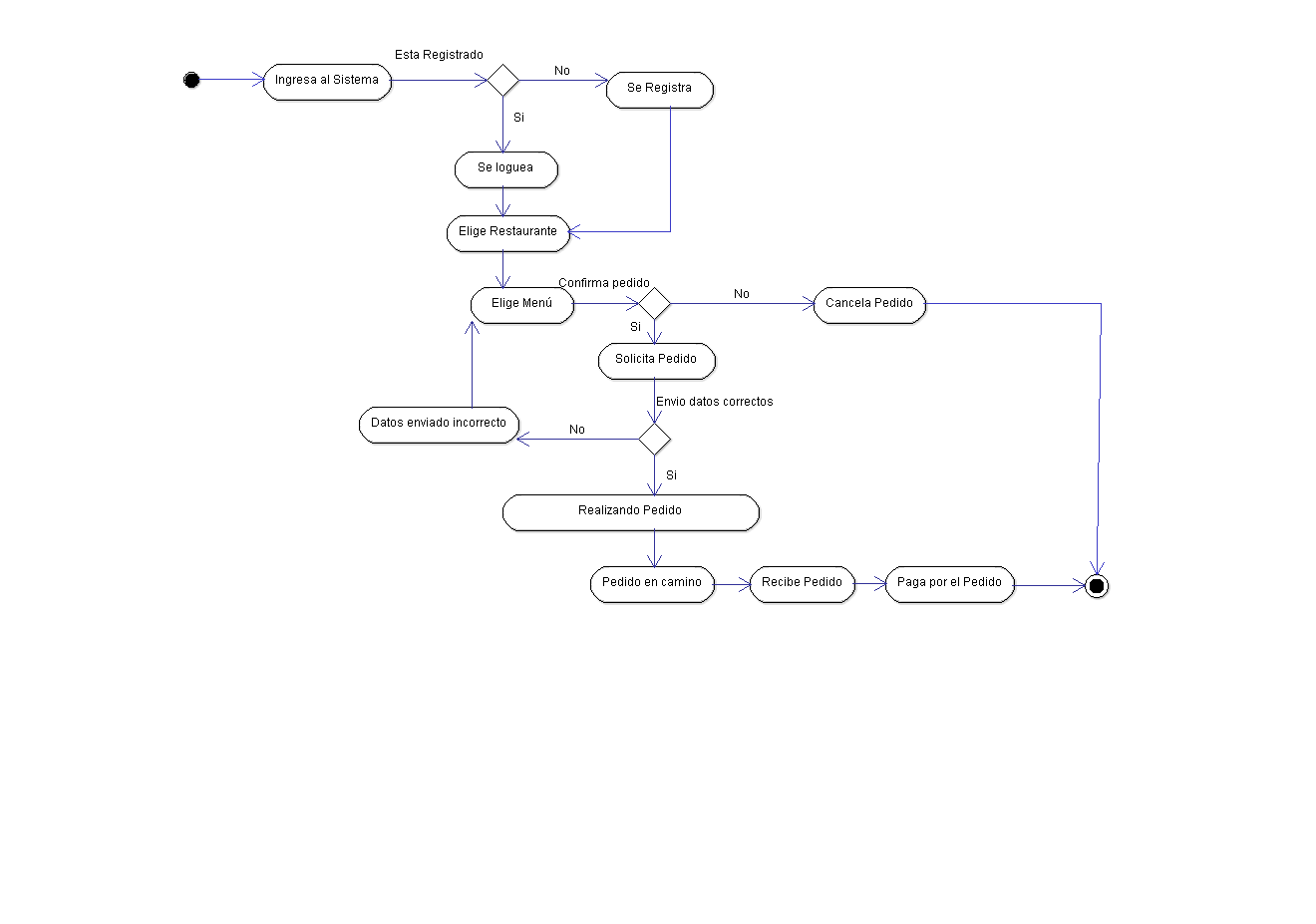
****

**Diagrama de Estados**

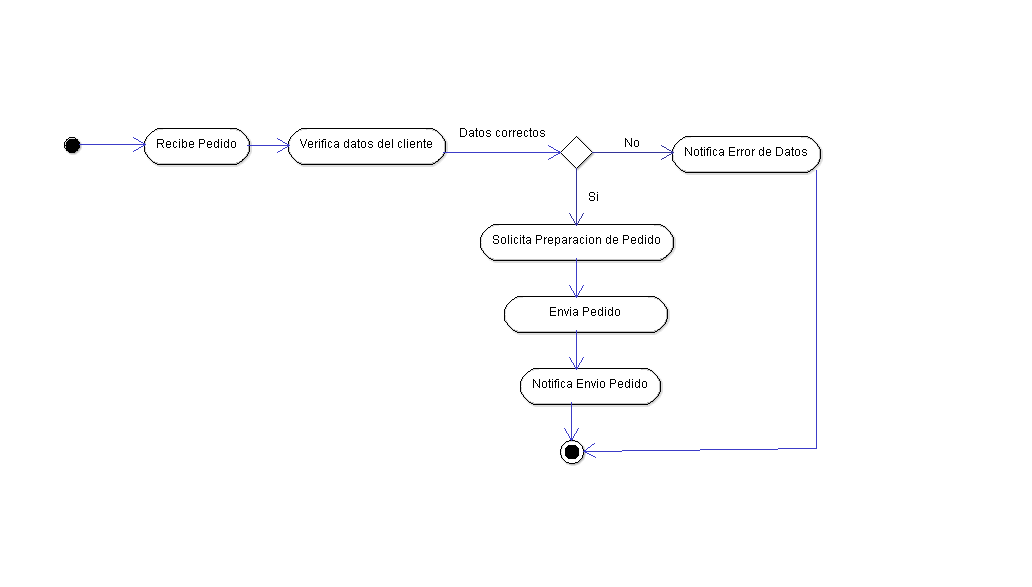
****

**Diagrama de Actividades**

Cliente

****

**Recepcionista**

****

**Diagrama de componente**

