Universidad Nacional de Ingeniería

UNI - RUACS

Facultad de Ciencias y Sistemas

Ingeniería de Sistemas.

Ingeniería de Software I

Continuación para Proyecto de Software II.

**Autores.**

* Br. Gagdiel Isaac Herrera Morales. Carnet: 2019-0104N
* Br. Axel Roberto Talavera Briones. Carnet: 2018-0376N
* Br. Oscar Eduardo Arauz Nuila Carnet:2019-0174N
* Br. Greyvis José Merlo Arauz Carnet: 2019-0244N

Grupo: 4T1-IS

**Facilitador/a.**

* Ing. Juan José Zeledón Benavidez.

índice

[**1-** **Introducción.** 3](#_Toc101255504)

[**2-** **Justificación.** 5](#_Toc101255505)

[**3-** **Objetivos.** 6](#_Toc101255506)

[**3.1- Objetivo General.** 6](#_Toc101255507)

[**3.2- Objetivos Específicos.** 7](#_Toc101255508)

[**4-** **Marco Teórico.** 7](#_Toc101255509)

[**4.1- Sistemas de información.** 7](#_Toc101255510)

[**4.2- Software de sistema.** 10](#_Toc101255511)

[**4.3- Sistemas de Ventas.** 10](#_Toc101255512)

[**4.3.1- Sistemas de Ventas local.** 11](#_Toc101255513)

[**4.3.2- Modelo Cliente / Servidor.** 11](#_Toc101255514)

[**4.3.3- Diferencia entre cliente y servidor** 13](#_Toc101255515)

[**4.3.4- Tipos de arquitecturas cliente servidor** 14](#_Toc101255516)

[**4.4- Viabilidad de un proyecto.** 16](#_Toc101255517)

[**4.5- Estudio de Viabilidad de Software.** 16](#_Toc101255518)

[**4.5.1- Viabilidad Operativa.** 16](#_Toc101255519)

[**4.5.2- Viabilidad Técnica.** 17](#_Toc101255520)

[**4.5.3- Viabilidad Económica.** 17](#_Toc101255521)

[**4.5.4- Viabilidad Financiera*.*** 17](#_Toc101255522)

[**4.5.5- Viabilidad Legal*.*** 17](#_Toc101255523)

[**4.6- Modelos de Estimación de costos de software.** 18](#_Toc101255524)

[**4.7- Especificación de requerimientos.** 19](#_Toc101255525)

[**4.8- Ciclo de desarrollo del software.** 20](#_Toc101255526)

[**4.9- Soluciones de ciclo de vida de desarrollo de software.** 22](#_Toc101255527)

[**4.10- Metodología RUP.** 23](#_Toc101255528)

[**4.11- Metodología Scrum.** 23](#_Toc101255529)

[**4.12- Sprint.** 25](#_Toc101255530)

[**4.13- Prácticas Recomendadas para la Especificación de Requisitos de Software** 26](#_Toc101255531)

[**4.14- Prácticas recomendadas para una buena ERS.** 26](#_Toc101255532)

[**4.15- Lenguaje de Modelado Unificado UML.** 27](#_Toc101255533)

[**4.16- Modelos de Proceso de Desarrollo del Software.** 28](#_Toc101255534)

[**4.17- Herramientas De Elaboración De Diagramas.** 29](#_Toc101255535)

[**4.17.1- Enterprise Architect.** 29](#_Toc101255536)

[**4.18- Herramientas para el desarrollo del software.** 29](#_Toc101255537)

[**4.18.1- Sistema gestor de base de datos.** 29](#_Toc101255538)

[**4.18.2- C#.** 32](#_Toc101255539)

[**5-** **Metodología de desarrollo.** 33](#_Toc101255540)

[**6-** **Primer capítulo. Estudio de Viabilidad.** 34](#_Toc101255541)

[**6.1- Viabilidad Técnica.** 34](#_Toc101255542)

[**6.2- Viabilidad Operativa** 38](#_Toc101255543)

[**6.3- Viabilidad Económica.** 39](#_Toc101255544)

[**6.4- Viabilidad Financiera.** 45](#_Toc101255545)

[**6.5- Viabilidad Legal.** 46](#_Toc101255546)

[**7-** **Segundo Capítulo. Ingeniería de Requerimientos (REM)** 49](#_Toc101255547)

[**7.1- Scrum.** 58](#_Toc101255548)

[**8-** **Tercer capítulo. Diseño y desarrollo.** 60](#_Toc101255549)

[**1.** **Recomendaciones.** 82](#_Toc101255550)

[Bibliografía 83](#_Toc101255551)

[Diagramas 1 Caso de uso gestión de usuario 65](#_Toc101367093)

[Diagramas 2 Diagrama de actividad validar usuario 67](#_Toc101367094)

[Diagramas 3 Diagrama de actividad agregar usuar 69](#_Toc101367095)

[Diagramas 4 Diagrama de clase Gestión de usuario 74](#_Toc101367096)

[Diagramas 5 Diagrama de estado agregar usuario 75](#_Toc101367097)

[Diagramas 6 Diagrama de estado editar usuario 76](#_Toc101367098)

[Diagramas 7 Diagrama Estado usuario 77](#_Toc101367099)

[Diagramas 8 Diagrama de secuencia agregar usuario 78](#_Toc101367100)

[Diagramas 9 Diagrama de secuencia editar usuario 78](#_Toc101367101)

[Diagramas 10 Diagrama caso de uso gestión de proveedor 80](#_Toc101367102)

[Diagramas 11 Diagrama de actividad agregar proveedor 81](#_Toc101367103)

[Diagramas 12 Diagrama de actividad editar proveedor 82](#_Toc101367104)

[Diagramas 13 Diagrama de clase gestión de proveedor 85](#_Toc101367105)

[Diagramas 14 Diagrama de estado agregar proveedor 86](#_Toc101367106)

[Diagramas 15Diagrama de estado editar proveedor 87](#_Toc101367107)

[Diagramas 16 Diagrama de sescuncia agregar proveedor 88](#_Toc101367108)

[Diagramas 17 Diagrama de secuencia editar proveedor 88](#_Toc101367109)

[Ilustración 1 Registro de usuario 80](#_Toc101371832)

[Ilustración 2 usuarios registrados 80](#_Toc101371833)

[Ilustración 3 Bitácora de inicio de sesión 81](#_Toc101371834)

[Ilustración 4 Ver proveedores, Editar proveedores 91](#_Toc101371835)

[Ilustración 5 Agregar proveedor 91](#_Toc101371836)

# **Introducción.**

Jamen’s Pollos es un negocio de venta de pollos ubicado en la ciudad de Estelí. Actualmente venden pollos enteros y los también famosos medio pollos, de igual manera venden productos extras como lo son: gaseosas, tajadas entre otros productos de consumo. El negocio ha logrado posicionarse en la zona este del país como uno de los centros más frecuentados para comer por lo cual buscan expandirse y abarcar más mercado.

Durante los últimos meses, Jamen´s Pollos ha presentado ciertos inconvenientes en lo que respecta el llevar datos exactos de sus ventas del día a día, de igual manera sus compras de productos y otras actividades que de una u otra manera afecta como tal al negocio.

Para los propietarios de Jamen´s Pollos, es de suma importancia que todas las funciones llevadas a cabo en este negocio sean proveedoras de datos reales para así tener un mejor control de toda la actividad que se presente en cumplimiento de sus deberes del día, ya que todo lo obtenido es registrado en un cuaderno de apunte, lo cual no ha sido de beneficio alguno para el negocio, porque siempre hay datos que no cuadran en todas sus cuentas finales.

Ahora bien, para dar una respuesta de solución ante la problemática que presentan, se propone la implementación de un Sistema de Información que cumpla con diversos requerimientos en función del usuario, tales como: gestión de sus ventas diarias, compras de productos a sus proveedores, presentaciones de combos, manejo de usuarios, entre otras funciones que al igual de las ya mencionadas son de gran utilidad e importancia para la prosperidad notoria del negocio.

Se debe de tener en cuenta que se recolectará la información necesaria mediante entrevistas, se determinará la viabilidad técnica, operativa y financiera para determinar si el proyecto es viable. Por último, el software se desarrollará mediante la metodología Scrum, utilizando el lenguaje orientado a objetos C# y el gestor de base de datos MySQL

Para llevar a cabo el desarrollo de este software, se hará de manera incremental; es decir que por cada semana se harán incrementos diferentes para que así en el tiempo que resulte determinado para entregarlo, se tenga el sistema competo y con toda la documentación necesaria que este requiere. Se prevé que, al implementar ese software al negocio, permita tener datos exactos y reales generados por cada una de las categorías que se presenten dentro de este.

# **Justificación.**

La meta principal de llevar a cabo este proyecto, data a que el negocio de venta de Pollos llamado Jamen´s Pollos, no contaba con un sistema de información como tal, en donde cada uno de los trabajadores y así mismo el administrador pudiera gestionar de manera exitosa los diversos campos necesarios para la recolección de información provenientes de cada uno de ellos, para así llevar un control de todo lo trabajado día a día y así compararlos con años anteriores.

El implementar este Sistema de Información para así tener todos los datos de entradas y salidas del negocio es considerable de mucha y vital importancia, ya que permite a los propietarios de Jamen´s Pollos tener un control absoluto de la información que se vaya recopilando, ya que con todo lo que se pretende desarrollar dentro de este sistema logrará que todo el trabajo realizado por los empleados resulte más fácil de completar.

Teniendo una breve previsualización del sistema, se pretende que este cumpla con las siguientes funciones ya de manera específica:

* Gestión de usuarios para así poder tener acceso al sistema.
* Gestión de venta de productos.
* Gestión de compras.
* Elaboración de combos.
* Etc

El desarrollo e implementación de este Software para Jamen´s Pollos, aumentará la seguridad de todo el negocio, ya que ahora sí podrán tener a su disposición toda la información que los propietarios demandan a sus trabajadores de las ventas y compras realizadas en el día.

# **Objetivos.**

## **3.1- Objetivo General.**

* Implementar un sistema de información de escritorio para la gestión de compra, venta, control de inventario y menús de manera Cliente - Servidor para la empresa Jamen’s Pollos.

## **3.2- Objetivos Específicos.**

* Determinar la viabilidad técnica, legal, operativa, económica de la implementación del sistema de gestión de ventas de la empresa Jamen’s Pollos.
* Identificar los requerimientos funcionales, no funcionales, de usuarios y de sistemas para el sistema a desarrollar.
* Diseñar el sistema de gestión de ventas local utilizando la metodología de proceso unificado Rational.
* Desarrollar el sistema de gestión de ventas utilizando la metodología Scrum y aplicándolo con las herramientas de desarrollo C# y gestor de base de datos MySql.
* Realizar pruebas de caja blanca y pruebas de caja negra a los diferentes requerimientos del sistema.
* Ejecutar pruebas de mantenimiento al sistema en general.
* Proyectar pruebas de riesgos al sistema.

# **Marco Teórico.**

## **4.1- Sistemas de información.**

Según la información de (Peiró, s.f.) un sistema de información está conformado por una serie de datos vinculados entre sí para conseguir un objetivo común.

Un sistema de información tiene como principal objetivo la gestión, y administración de los datos e información que lo componen. Lo importante es poder recuperar siempre esos datos, y que además se tenga un fácil acceso a ellos con total seguridad.

Los componentes del sistema de información, permiten una serie de procesos que consisten en: la entrada de los datos, la gestión y el procesamiento de estos, el almacenamiento y la salida para todos aquellos interesados que deseen tener acceso a este tipo de información.

Los elementos del sistema de información trabajan de manera conjunta y con los mismos objetivos para conseguir el uso y la correcta administración de cualquier información concreta.

**Cómo funciona un sistema de información**

Esta es la metodología habitual:

* La entrada de la información. En este paso se da entrada a la información de manera automática o manual, dependiendo de la técnica que se utilice para incluir los datos. Las manuales las introduce directamente el usuario, las automáticas se gestionan a través de información recibida o proveniente de otro tipo de áreas y módulos. Las formas de entrada más habituales de registros y datos son las siguientes: códigos de barra, el teclado de un ordenador, cajas registradoras, sistemas de voz, un escáner, entre otros.
* El procesamiento de la información registrada. En el sistema existen una serie de operaciones y acciones que previamente se han configurado y establecido. Una vez que se ha producido la entrada de la información se procede a transformar esos datos en información requerida para la toma de decisiones, valoración, investigación y análisis para llevar a cabo un balance o visión general en función de los contenidos que resultan de este proceso.
* El almacenamiento de la información. Esta opción permite que la información quede registrada en el sistema, en un ordenador, por ejemplo. De esta forma, si quiere revisarse o acceder a ella cuando se requiera, siempre estará almacenada y accesible para las tareas que se requieran. En los ordenadores suele almacenarse en discos duros interno, o extraíbles, o incluso en los CDs habituales, aunque su utilización suele ser menor, y en muchos ordenadores se ha dejado de incluir el soporte para ellos.
* La salida de la información. En este caso la información que está incluida en un determinado soporte y previamente procesada, facilita a un usuario a que tenga acceso a ella y pueda sacarla a través de diferentes dispositivos como: usb, impresoras, sistemas de voz, entre otros.

**Los sistemas de información más destacados**

Estos son los principales sistemas que existen:

* Sistemas de información en marketing y comunicación. Su principal objetivo es la venta y promoción de artículos. La creación de información y datos para que se puedan dar a conocer este tipo de artículos y sean de interés para el público.
* El sistema de información administrativa (MIS). Es uno de los más importantes, y se basa en proporcionar la información más destacada y relevante que necesita la empresa para la toma de decisiones, y valoración de datos para llevar a cabo un plan de acción posterior en base a esta recogida de documentos.
* El sistema de procesamiento de transacciones (TPS). En este caso los datos aquí se incluyen son los referidos a las actividades comerciales de la empresa.
* El sistema de apoyo a ejecutivos (EIS). Destinado especialmente a aportar información y datos relevantes para la toma de decisiones estratégicas en concreto.
* El sistema para la toma de decisiones en grupo (GDSS). La información tiene como objetivo ser compartida entre los componentes del grupo para que se faciliten las tareas de trabajos y la cooperación entre todos.
* El sistema de información para expertos (EDSS). En este caso lo que se trata de conseguir es compartir datos de interés entre determinados segmentos y especialistas para valorar esa información destinada a ellos.

## **4.2- Software de sistema.**

En el campo de la computación, un software de sistema (también conocido como software de base) es la forma de referirse a todo sistema informático o programa que se encarga de permitir el manejo y operatividad de un ordenador. Así, este tipo de software es la estructura base que permite que el hardware funcione de manera conjunta a los distintos softwares que se instalen en él.

El software de sistema se caracteriza por tener una estructura sencilla a la que generalmente acceden los programadores o personal capacitado para administrar la manera en la cual los distintos componentes de hardware son utilizados por el sistema operativo del ordenador. Puede decirse que el software de base es el que permite que una computadora pueda ser manipulada y aprovechada por los usuarios convencionales a través de un sistema operativo. A pesar de que un software de sistema y un sistema operativo no son lo mismo, muchos sistemas operativos hacen posible la manipulación del software de sistema.

En la actualidad, todas las computadoras, sin importar que uso, sistema operativo o empresa que la fabrique, emplea un software de sistema que se encuentra alojado en su memoria ROM. De esta manera, se garantiza que siempre sea posible acceder a éste, sin importar si se reemplazan piezas del hardware del ordenador que contengan información necesaria para el inicio del SO del mismo. (ConceptoABC, s.f.)

## **4.3- Sistemas de Ventas.**

Según (Polo, s.f.) define como sistema de ventas a un programa o aplicación que se encuentre completo, para así llevar a cabo una gestión de clientes, proveedores y productos, donde se puede incluir la posibilidad de realizar un registro de ventas de dichos productos y que a la vez genere informe.

Por lo general se sabe que la implementación de un sistema de venta para un negocio es una pieza central para todos los procesos que se lleven a cabo, es decir es el centro de todo el negocio (Ventas, inventarios, gestión de clientes, productos, entre otros).

Se debe conocer los dos elementos existentes dentro de un sistema de ventas, los cuales son de mucha importancia y utilidad dentro de este, estos son:

* Componentes de Software.
* Componentes de Hardware.

Estos componentes al estar en funcionamiento en el sistema de ventas hacen que las funciones se realicen y de tal manera todos los procesos importantes también sean realizados con éxito, en resumidas palabras se podrá llevar el control de las operaciones del día a día del negocio donde se implemente. (Sistema Venta, 2020)

Existen dos tipos de sistemas de ventas los cuales son muy útiles para las diversas necesidades que se puedan presentar en sus negocios, estos son:

* Sistema de Venta Local.
* Sistema de venta en la nube.

### **4.3.1- Sistemas de Ventas local.**

Es una solución administrativa que adquieres al pagar por una licencia. Por ejemplo, con el Software ClassicGes 6 adquieres una licencia anual para poder utilizar todas sus funcionalidades.

Cuando comprar un software de ventas local, simplemente lo instalas en tus equipos, configuras y lo utilizas en toda tu tienda. Algunos proveedores te permiten utilizarlo en un número ilimitado de hardware.

En ocasiones requerirás de un equipo especializado para instalar y configurar correctamente. Casi siempre, las empresas ofrecen este servicio. (Sistemas Ventas, 2020).

### **4.3.2- Modelo Cliente / Servidor.**

La arquitectura cliente servidor tiene dos partes claramente diferenciadas, por un lado, la parte del servidor y por otro la parte de cliente o grupo de clientes donde lo habitual es que un servidor sea una máquina bastante potente con un hardware y software específico que actúa de depósito de datos y funcione como un sistema gestor de base de datos o aplicaciones.

En esta arquitectura el cliente suele ser estaciones de trabajo que solicitan varios servicios al servidor, mientras que un servidor es una máquina que actúa como depósito de datos y funciona como un sistema gestor de base de datos, este se encarga de dar la respuesta demandada por el cliente.

Esta arquitectura se aplica en diferentes modelos informáticos alrededor del mundo donde su propósito es mantener una comunicación de información entre diferentes entidades de una red mediante el uso de protocolos establecidos y el apropiado almacenaje de la misma.

El más claro ejemplo de uso de una arquitectura cliente servidor es la red de Internet donde existen ordenadores de diferentes personas conectadas alrededor del mundo, las cuales se conectan a través de los servidores de su proveedor de Internet por ISP donde son redirigidos a los servidores de las páginas que desean visualizar y de esta manera la información de los servicios requeridos viaja a través de Internet dando respuesta a la solicitud demandada.

La principal importancia de este modelo es que permite conectar a varios clientes a los servicios que provee un servidor y como sabemos hoy en día, la mayoría de las aplicaciones y servicios tienen como gran necesidad que puedan ser consumidos por varios usuarios de forma simultánea.

Componentes

Para entender este modelo vamos a nombrar y definir a continuación algunos conceptos básicos que lo conforman.

* Red: Una red es un conjunto de clientes, servidores y base de datos unidos de una manera física o no física en el que existen protocolos de transmisión de información establecidos.
* Cliente: El concepto de cliente hace referencia a un demandante de servicios, este cliente puede ser un ordenador como también una aplicación de informática, la cual requiere información proveniente de la red para funcionar.
* Servidor: Un servidor hace referencia a un proveedor de servicios, este servidor a su vez puede ser un ordenador o una aplicación informática la cual envía información a los demás agentes de la red.
* Protocolo: Un protocolo es un conjunto de normas o reglas y pasos establecidos de manera clara y concreta sobre el flujo de información en una red estructurada.
* Servicios: Un servicio es un conjunto de información que busca responder las necesidades de un cliente, donde esta información pueden ser mail, música, mensajes simples entre software, videos, etc.
* Base de datos: Son bancos de información ordenada, categorizada y clasificada que forman parte de la red, que son sitios de almacenaje para la utilización de los servidores y también directamente de los clientes.

### **4.3.3- Diferencia entre cliente y servidor**

Como hemos mencionado anteriormente una máquina cliente como servidor se refieren a computadoras que son usadas para diferentes propósitos.

El cliente es un computador pequeño con una estructura al igual a la que tenemos en nuestras oficinas u hogares la cual accede a un servidor o a los servicios del mismo a través de Internet o una red interna. Un claro ejemplo a este caso es la forma en que trabaja una empresa modelo con diferentes computadores donde cada uno de ellos se conectan a un servidor para poder obtener archivos de una base de datos o servicios ya sea correos electrónicos o aplicaciones.

El servidor al igual que el cliente, es una computadora, pero con diferencia de que tiene una gran capacidad que le permite almacenar gran cantidad de diversos de archivos, o correr varias aplicaciones en simultaneo para así nosotros los clientes poder acceder los servicios.

En la actualidad existen varios tipos de servidores como hablamos anteriormente. Los mismos pueden contener y ejecutar aplicaciones, sitios web, almacenaje de archivos, diversas bases de datos, entre muchos más.

Es importante mencionar que un cliente también puede tener una función de servidor ya que el mismo puede almacenar datos en su disco duro para luego ser usados en vez de estar conectándose al servidor continuamente por una acción que quizás sea muy sencilla.

### **4.3.4- Tipos de arquitecturas cliente servidor**

Dentro de la arquitectura cliente servidor existen tres tipos en donde hablaremos brevemente de cómo funciona cada uno de ellos.

* Arquitectura de dos capas

Esta se utiliza para describir los sistemas cliente servidor en donde el cliente solicita recursos y el servidor responde directamente a la solicitud con sus propios recursos. Eso significa que el servidor no requiere de una aplicación extra para proporcionar parte del servicio.

* Arquitectura de tres capas

En la arquitectura de tres capas existe un nivel intermediario, eso significa que la arquitectura generalmente está compartida por un cliente que como hablamos más arriba es el que solicita los recursos equipados con una interfaz de usuario o mediante un navegador web.

La capa del medio es denominada software intermedio cuya tarea es proporcionar los recursos solicitados pero que requiere de otro servidor para hacerlo. La última capa es el servidor de datos que proporciona al servidor de aplicaciones los datos necesarios para poder procesar y generar el servicio que solicito el cliente en un principio.

* Arquitectura N capas

En la arquitectura de tres capas, los servidores dos y tres realizaron una tarea específica por lo tanto un servidor web puede usar los servicios de otros servidores para poder proporcionar su propio servicio.

Por consiguiente, la arquitectura en tres niveles es potencialmente una arquitectura en N capas ya que, así como está contemplado en tres niveles como el caso anterior puede estar compuesto por N servidores donde cada uno de ellos brindan su servicio específico.

**Ventajas y Desventajas**

Este modelo cliente servidor tiene varias ventajas y desventajas las cuales son importantes mencionar y conocer a la hora de establecer si es lo que estamos necesitando o si se acomoda a lo que estamos buscando.

**Ventajas**

Facilita la integración entre diferentes sistemas y comparte información permitiendo por ejemplo que las máquinas ya existentes puedan ser utilizadas mediante una interfaz más amigable para el usuario. De esta manera podemos integrar varias PC con sistemas medianos y grandes sin necesidad de que todos tengan que utilizar el mismo sistema operativo.

Al favorecer el uso de las interfaces de gráficas interactivas, los sistemas construidos bajo este esquema tienen una mayor interacción con el usuario.

La estructura modular facilita de más la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional favoreciendo así la estabilidad de las soluciones.

El modelo cliente servidor permite además proporcionar a las diferentes áreas de una empresa generar un orden de trabajo en donde cada sector puede trabajar en su área, pero accediendo al mismo servidor e información que los demás sin generar conflictos. Esto es de gran utilidad ya que si ponemos como ejemplo una empresa con varios empleados al momento de trabajar es importante que todos puedan hacerlo en simultáneo.

**Desventajas**

Requiere habilidad para que un servidor sea reparado. Por ejemplo, si un problema ocurre en la red, se requiere de alguien con un amplio de esta para poder repararla en su totalidad para así dejar que la información y el correcto funcionamiento siga su flujo.

Otro problema es la seguridad, el hecho que se comparte canales de información entre servidores y clientes requieren que estas pasen por procesos de validación, es decir protocolos de seguridad que pueden tener algún tipo de puerta abierta permitiendo que se generen daños físicos, amenazas o ataques de malware.

Este modelo representa una limitación importante en cuanto a los costos económicos debido a que estos servidores son computadoras de alto nivel con un hardware y software específicos para poder dar un correcto funcionamiento a nuestras aplicaciones. Algo importante a destacar es que no solo es caro a la hora de solucionar problemas como mencionamos antes, sino que también tiene un costo elevado para reemplazar componentes que estén averiados. (Schiaffarino, 2019)

## **4.4- Viabilidad de un proyecto.**

Es un análisis detallado de un proyecto planificado. Incluye un análisis tanto financiero como económico, técnico, legal e institucional destinado a examinar la legitimidad del proyecto. El objetivo principal del estudio de viabilidad de un proyecto es proporcionar toda la información necesaria para verificar si el proyecto tiene la base adecuada para su ejecución. Y, además, si satisface las necesidades de los beneficiarios previstos. Dicho estudio debe brindar todos los datos que sean necesarios para así tomar una decisión de inversión tanto al inversionista como a la institución financiera, que evalúa la inversión con base en el mismo. (S., 2021)

## **4.5- Estudio de Viabilidad de Software.**

Este estudio evalúa si las necesidades de los usuarios pueden estar al nivel de la tecnología actual de hardware y software. La investigación analiza si el sistema propuesto es rentable desde el punto de vista Negocio y si se puede desarrollar dentro del presupuesto disponible ahora.

Existen diferentes tipos de viabilidades a tomar en cuenta para así aplicarlas en el sistema, estas son:

### **4.5.1- Viabilidad Operativa.**

un sistema es operacionalmente viable si será utilizado efectivamente después de su desarrollo. Si los usuarios tienen dificultad con el nuevo sistema, éste no producirá los beneficios esperados. Es necesario preguntarse si: o la gerencia y los usuarios apoyan el proyecto; o el nuevo sistema requiere adiestramiento para los usuarios y la compañía está preparada para ofrecerlo; o el nuevo sistema traerá como consecuencia una reducción del personal; o si los usuarios estarán involucrados en planificar el nuevo sistema desde sus comienzos; o si el nuevo sistema requiere algún cambio en la manera en que se realizan las tareas, entre otras cosas. (Monografias, s.f.)

### **4.5.2- Viabilidad Técnica.**

Hace referencia a aquello que atiende a las características tecnológicas y naturales involucradas en un proyecto. El estudio de la viabilidad técnica suele estar vinculado a la seguridad y al control de lo que vamos a hacer; esto es, a sus características, funcionalidades y propiedades físicas y a cómo lo vamos a hacer. Tendremos que conocer cuál es el proceso de fabricación/realización, los medios técnicos necesarios, los medios humanos que van a intervenir y su cualificación, los materiales necesarios, control de calidad, gestión de residuos, etc. El estudio de viabilidad técnica conlleva resolver la pregunta de si es posible, desde el punto de vista tecnológico, desarrollar eficientemente nuestros productos/servicios.

### **4.5.3- Viabilidad Económica.**

El proyecto es viable, desde el punto de vista económico, si, con los recursos que somos capaces de conseguir, es capaz de generar beneficios y tener una rentabilidad suficiente que compense los riesgos en los que se va a incurrir.

### **4.5.4- Viabilidad Financiera*.***

El proyecto es viable, desde el punto de vista financiero, si no plantea problemas de tesorería y tiene una estructura financiera equilibrada, en cuanto a endeudamiento, solvencia y liquidez. Esto es, si con el dinero que se va a generar (inicialmente, con aportaciones de socios y endeudamiento, y luego con los cobros por las ventas derivadas de nuestra actividad) se puede hacer frente, puntualmente, a los pagos. (UGR Emprendedora, 2017)

### **4.5.5- Viabilidad Legal*.***

Muestra precisamente el respaldo de la ley para llevar a cabo esa acción determinada de un modo correcto. Por el contrario, algo es inviable desde el punto de vista legal cuando no está justificado por las leyes del derecho. (Nicuesa, 2015)

## **4.6- Modelos de Estimación de costos de software.**

El proceso de estimación del coste de un producto software está formado por un conjunto de técnicas y procedimientos que se usan en la organización para poder llegar a una predicción fiable. Éste es un proceso continuo, que debe ser usado y consultado a lo largo de todo el ciclo de vida del proyecto. Se divide en los siguientes pasos:

* Estimación del tamaño.
* Estimación del costo y del esfuerzo.
* Estimación de la programación temporal.
* Estimación de la cantidad de recursos computacionales.
* Asunción de riesgos.
* Inspección y aprobación.

Todas estas estimaciones necesarias están basadas en probabilidades debido a la influencia de factores externos de difícil control. Además de estas probabilidades, es necesario recurrir a información histórica, que debe ser fácilmente accesible y disponible para la organización en cualquier momento.

No siempre se presta la suficiente atención al cuidado de estos datos, por lo que en ocasiones acceder a ellos no resulta sencillo. La cantidad de esfuerzo y tiempo dedicada a la estimación depende del tamaño del proyecto, del equipo de desarrollo y del objetivo a cumplir. La naturaleza del proyecto y el entorno en el que se desarrolla son factores determinantes en esta tarea, y afectan en gran medida al método de estimación que se utilice.

De una manera general podemos afirmar que existen dos maneras diferentes de estimar el presupuesto y el tiempo para un proyecto software:

usando modelos de costo y usando razonamiento basado en similitud. En ambas opciones es necesario recurrir a información histórica y de proyectos anteriores previamente almacenados en bases de datos. (Sites google, s.f.)

El modelo de estimación de costos que por lo general se aplica a los softwares es el COCOMO.

Es un modelo que permite estimar el costo, el esfuerzo, y programar la hora de planificar una nueva actividad de desarrollo de software.

El modelo provee tres «niveles» de aplicación: básico, intermedio y avanzado, basados en los factores considerados por el modelo.

Básico, es un modelo estático simplemente evaluado que calcula el esfuerzo (y costo) del desarrollo del software como función del programa expresado en líneas de código (LDC estimados).

Intermedio, calcula el esfuerzo del desarrollo del software como función del tamaño del programa y un conjunto de «guías de costo» que incluye una evaluación subjetiva del producto, hardware, personal y de los atributos del proyecto.

Avanzado, incorpora todas las características de la versión intermedia con una evaluación del impacto de las vías de costo en cada fase (análisis, diseño, etc) del proceso de la ingeniería de software. (Acevedo, 2008)

## **4.7- Especificación de requerimientos.**

En la Ingeniería de software, un Análisis de Requerimientos es una tarea que cubre el hueco entre la definición del software a nivel sistema y el diseño del mismo.

Tanto el desarrollador como el cliente tienen un papel activo. Juntos definen en detalle los requisitos del sistema a desarrollar y los pasos a seguir. (Romero, s.f.)

Estos pueden dividirse en dos partes, funcionales y no funcionales.

* Funcionales: Condición o capacidad de un sistema requerida por el usuario para resolver un problema o alcanzar un objetivo.
* No Funcionales: Condición o capacidad que debe poseer un sistema para satisfacer un contrato, un estándar, una especificación u otro documento formalmente impuesto.

El análisis de requerimientos es la tarea que plantea la asignación de software a nivel de sistema y el diseño de programas. Facilita al ingeniero de sistemas especificar la función y comportamiento de los programas, indicar la interfaz con otros elementos del sistema y establecer las ligaduras de diseño que debe cumplir el programa. Permite al ingeniero refinar la asignación de software y representar el dominio de la información que será tratada por el programa. (Interpolados, 2018)

## **4.8- Ciclo de desarrollo del software.**

Según la información brindada por (Ungoti, s.f.), el ciclo de vida del desarrollo de software (en inglés: SDLC – Systems Development Life Cycle) es la estructura que contiene los procesos, actividades y tareas relacionadas con el desarrollo y mantenimiento de un producto de software, abarcando la vida completa del sistema, desde la definición de los requisitos hasta la finalización de su uso.

Se trata de evitar los costes de rectificar errores de implementación mediante un método que permita a los programadores adelantarse para mejorar sus resultados finales. Este sistema de desarrollo (o ciclo de vida del proceso de software), necesita de varios pasos imprescindibles para garantizar que los programas ofrezcan una buena experiencia al usuario, seguridad, eficiencia, estabilidad y fiabilidad de uso.

**Fases de SDLC.**

* Comunicación.

Este es el momento en el que un cliente solicita un producto de software determinado. Nos contacta para plasmar sus necesidades concretas y presenta su solicitud de desarrollo de software.

* Planificación y análisis.

El desarrollo de software comienza con una fase inicial de planificación incluyendo un análisis de requisitos. Nos fijamos en los requisitos que piden los clientes para estudiar cuales están poco claros, incompletos, ambiguos o contradictorios. Se indaga en profundidad y se hacen demostraciones prácticas incluyendo a los usuarios clave. Los requisitos se agrupan en requisitos del usuario, requisitos funcionales y requisitos del sistema. La recolección de todos los requisitos se lleva a cabo: estudiando el software actual que tengan, entrevistando a usuarios y desarrolladores, consultando bases de datos o mediante cuestionarios.

* Estudio de viabilidad.

Después de la recolección de requisitos, se idea un plan para procesar el software. Se analiza que parte del software cubre los requisitos de cada usuario. Se investiga la viabilidad financiera y tecnológica. Se utilizan algoritmos para saber si el proyecto de software es factible o no.

* Análisis del sistema.

En este paso el equipo del proyecto asigna recursos y planifica el tiempo de duración del proyecto. Se buscan limitaciones del producto y se identifican los impactos del proyecto sobre toda la organización en su conjunto.

* Diseño.

En esta fase ya se comienza a visualizar la solución con la ayuda de las anteriores fases. Se hace un diseño lógico y otro físico. Se crean metadatos, diagramas o pseudocódigos. La duración de esta fase varía de un proyecto a otro.

* Codificación.

Esta fase también denominada ‘fase de programación’ o ‘fase de desarrollo’ es en la que elige el lenguaje de programación más conveniente, y se desarrollan programas ejecutables y sin errores de manera eficiente. Nuestro enfoque es construir trozos de funcionalidad. Por lo tanto, entregar unidades de funcionalidad concisa. Al final de esta fase se puede obtener un PMV (Producto mínimo viable) o el software completamente desarrollado y listo para implementarse.

* Integración.

El Software puede necesitar estar integrado con bibliotecas, bases de datos o con otros programas. Esta fase del SDLC integra el software con las entidades del mundo exterior.

* Pruebas.

Esta fase junto con la fase de desarrollo entra en un ciclo continuo hasta que se completan el desarrollo y las pruebas. Probamos, probamos y luego volvemos a probar tanto como sea necesario hasta que la funcionalidad sea del 100%.

Además, se hacen evaluaciones para evitar errores, incluyendo la evaluación de módulos, programas, productos, y finalmente evaluación con el cliente final. Encontrar errores y su arreglarlos a tiempo es la clave para conseguir un software confiable y eficiente.

* Implementación.

Aquí se instala el software, se evalúa la integración, la adaptabilidad, la portabilidad y se instalan las configuraciones posteriores necesarias.

* Formación.

Esta es la fase más interesante, ¡La formación! La adopción del usuario es muy importante y para ello ofrecemos capacitación inicial para cada usuario. Es importante comprobar el nivel de uso, la experiencia de usuario y resolver cualquier dificultad que pueda surgir a la hora de enfrentarse a un nuevo sistema o plataforma.

* Mantenimiento y funcionamiento.

Por último, pero no menos importante el mantenimiento es uno de los elementos clave de éxito de cualquier proyecto. En esta fase se minimizan pequeños errores, se confirma el buen funcionamiento del software, su eficiencia y estabilidad. El proyecto ya está completado y necesitamos monitorear y mantener de forma continua para garantizar que el proyecto siga ejecutándose bien.

Si es necesario se dan nuevas formaciones, o se presta documentación sobre como operar y mantener el software en perfecto estado de funcionamiento. Se adaptan entornos del usuario o tecnológicos, dando mantenimiento al software, actualizando el código y configuración.

El software es efectivo cuando se usa de forma apropiada y por eso el mantenimiento y la mejora de los productos de software es crucial para poder corregir defectos que vayan surgiendo o para poder atender a los requisitos del software.

## **4.9- Soluciones de ciclo de vida de desarrollo de software.**

Mediante el soporte de Azure DevOps Server de Microsoft te proporcionamos control de versiones, informes, administración de requisitos, administración de proyectos, compilaciones automatizadas, pruebas y capacidades de administración de versiones. Cubriendo todo el ciclo de vida del software y habilitando las capacidades de DevOps.

## **4.10- Metodología RUP.**

La metodología de desarrollo RUP por sus siglas en inglés o Proceso de Desarrollo Unificado es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Se puede utilizar RUP desde el principio de un nuevo proyecto de software, y puede seguir utilizándolo en los ciclos de desarrollo subsiguientes tiempo después de que el proyecto inicial haya terminado. No obstante, la forma de utilizar RUP varía para ajustarse a sus necesidades. (A.L.EPH, 2021)

## **4.11- Metodología Scrum.**

(SOFTENG, s.f.) nos dice que scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa (ROI). Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación.

Con la metodología Scrum el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Así mismo le permite en cualquier momento realinear el software con los objetivos de negocio de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema.

Esta metódica de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

**Beneficios.**

* Cumplimento de expectativas:

El cliente establece sus expectativas indicando el valor que le aporta cada requisito / historia del proyecto, el equipo los estima y con esta información el Product Owner establece su prioridad. De manera regular, en las demos de Sprint el Product Owner comprueba que efectivamente los requisitos se han cumplido y transmite se feedback al equipo.

* Flexibilidad a cambios:

Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.

* Reducción del Time to Market:

El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.

* Mayor calidad del software:

La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.

* Mayor productividad:

Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.

* Maximiza el retorno de la inversión (ROI):

Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.

* Predicciones de tiempos:

Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.

* Reducción de riesgos:

El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

## **4.12- Sprint.**

Sprint es el nombre que va a recibir cada uno de los ciclos o iteraciones que vamos a tener dentro de dentro de un proyecto Scrum.

Nos van a permitir tener un ritmo de trabajo con un tiempo prefijado, siendo la duración habitual de un Sprint unas cuatro semanas, aunque lo que la metodología dice es que debería estar entre dos semanas y un máximo de dos meses.

En cada Sprint o cada ciclo de trabajo lo que vamos a conseguir es lo que se denomina un entregable o incremento del producto, que aporte valor al cliente.

La idea es que cuando tenemos un proyecto bastante largo, por ejemplo, un proyecto de 12 meses, vamos a poder dividir ese proyecto en doce Sprints de un mes cada uno. En cada uno de esos Sprints vamos a ir consiguiendo un producto, que siempre, y esto es muy importante, sea un producto que esté funcionando.

Una pila o product backlog, que serían todos los requisitos que nos pide el cliente, es decir, el año completo de trabajo. La idea es ir seleccionando esos requisitos en los que tenemos la pila dividida, y los vamos a ir haciendo en diferentes Sprints.

Esto se haría seleccionando esos requisitos realizamos todos los pasos que conforman un Sprint, es decir, la toma de requisitos, diseño, implementación, pruebas y despliegue en el plazo establecido, y así vamos a tener siempre un software que sea válido, un software funcionando.

**Lo que engloba un Sprint.**

Cuando hablamos de Sprint en Scrum técnico, engloba todo el proceso, es decir, desde que decidimos qué vamos a hacer para ese Sprint, hasta que estudiamos cómo hemos trabajando en ese Sprint.

Cuando estamos en un proyecto Scrum y comenzamos un Sprint, el mismo siempre comienza con la reunión de planificación del Sprint. En esa reunión vamos a decidir lo que vamos a hacer y cómo lo vamos a hacer, el número de tareas o de historias de usuario que vamos a realizar en el Sprint.

Después vamos a tener otra serie de reuniones:

* Reuniones de Scrum diario, que van a ser pequeñas reuniones con los miembros del equipo.
* Revisión del Sprint, en la que vamos a aceptar o denegar el Sprint.
* Reunión de retrospectiva, dónde vamos a ver cómo ha trabajado el equipo y qué problemas ha tenido durante el desarrollo y cómo lo podemos corregir.

El Spring engloba todo lo anterior, desde que comienza el mismo hasta que es aceptado o denegado, y el equipo se pregunta cómo ha trabajado. (Mesa, 2018)

**4.13- Prácticas Recomendadas para la Especificación de Requisitos de Software.**

Según la fuente de información (es.what-this.com, 2020), la especificación de requisitos de software es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software. Los casos de uso también son conocidos como requisitos funcionales. Además de los casos de uso, la ERS también contiene requisitos no funcionales. Los requisitos no funcionales son requisitos que imponen restricciones en el diseño o la implementación, como, por ejemplo, restricciones en el diseño o estándares de calidad. Está dirigida tanto al cliente como al equipo de desarrollo. El lenguaje utilizado para su redacción debe ser informal, de forma que sea fácilmente comprensible para todas las partes involucradas en el desarrollo.

## **4.14- Prácticas recomendadas para una buena ERS.**

Las características de una buena ERS son definidas por el estándar IEEE 830-1998. Una buena ERS debe ser:

* Verificable. Debe existir un método finito sin costo para poder probarlo.
* Priorizable. Los requisitos deben poder organizarse jerárquicamente según su relevancia para el negocio y clasificándolos en esenciales, condicionales y opcionales.
* Consistente. Debe ser coherente con los propios requerimientos y también con otros documentos de especificación.
* Correcta. El software debe cumplir con los requisitos de la especificación.
* Trazable. Se refiere la posibilidad de verificar la historia, ubicación o aplicación de un ítem a través de su identificación almacenada y documentada.
* Completa. Todos los requerimientos deben estar reflejados en ella y todas las referencias deben estar definidas.
* Inequívoca. La redacción debe ser clara de modo que no se pueda mal interpretar.
* Modificable. Aunque todo requerimiento es modificable, se refiere a que debe ser fácilmente modificable.

## **4.15- Lenguaje de Modelado Unificado UML.**

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. UML tiene aplicaciones más allá del desarrollo de software, p. ej., en el flujo de procesos en la fabricación.

Es comparable a los planos usados en otros campos y consiste en diferentes tipos de diagramas. En general, los diagramas UML describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene.

UML no es un lenguaje de programación, pero existen herramientas que se pueden usar para generar código en diversos lenguajes usando los diagramas UML. UML guarda una relación directa con el análisis y el diseño orientados a objetos. (Lucidchart, s.f.)

El UML está conformado por diferentes diagramas, entre ellos encontramos:

* Diagrama de Caso de Uso.
* Diagrama de Actividades.
* Diagrama de Clases.
* Diagrama de Secuencias.
* Diagrama de Estado.
* Diagrama de Componentes.

## **4.16- Modelos de Proceso de Desarrollo del Software.**

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente. Este proceso es intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas. Aunque un proyecto de desarrollo de software es equiparable en muchos aspectos a cualquier otro proyecto de ingeniería, en el desarrollo de software hay una serie de desafíos adicionales, relativos esencialmente a la naturaleza del producto obtenido.

El proceso de desarrollo de software no es único. No existe un proceso de software universal que sea efectivo para todos los contextos de proyectos de desarrollo. Debido a esta diversidad, es difícil automatizar todo un proceso de desarrollo de software. (Silverio, s.f.)

A pesar de la variedad de propuestas de proceso de software, existe un conjunto de actividades fundamentales que se encuentran presentes en todos ellos:

* Especificación de software: Se debe definir la funcionalidad y restricciones operacionales que debe cumplir el software.
* Diseño e Implementación: Se diseña y construye el software de acuerdo a la especificación.
* Validación: El software debe validarse, para asegurar que cumpla con lo que quiere el cliente.
* Evolución: El software debe evolucionar, para adaptarse a las necesidades del cliente.

Además de estas actividades fundamentales, Pressman menciona un conjunto de "actividades protectoras", que se aplican a lo largo de todo el proceso del software. Ellas se señalan a continuación:

* Seguimiento y control de proyecto de software.
* Revisiones técnicas formales.
* Garantía de calidad del software.
* Gestión de configuración del software.
* Preparación y producción de documentos.
* Gestión de reutilización.
* Mediciones.
* Gestión de riesgos.

Un modelo de desarrollo de Software es el Scrum, el cual consiste en un conjunto de buenas prácticas que permite el trabajo de los equipos involucrados en proyectos TIC de manera colaborativa, con una fluida relación entre cliente y proveedor

## **4.17- Herramientas De Elaboración De Diagramas.**

Es indispensable el uso de diversas herramientas para lo que es la elaboración de diagramas de UML (Unified Modeling Language), ya que estos son sumamente necesarios para tener una mejor comprensión acerca de las funciones correspondientes del sistema que se vaya a desarrollar y de tal manera realizar la documentación necesaria.

### **4.17.1- Enterprise Architect.**

Enterprise Architect es una herramienta gráfica multiusuario diseñada para ayudar a su equipo a construir sistemas robustos y fáciles de mantener. Incorporando reporting integrado y documentación de alta calidad, usted podrá ofrecer fácilmente una visión compartida y con gran nivel de fiabilidad. (Rambla Informática, s.f.)

Por medio de este software es que podemos darle mayor comprensión a cada detalle que se le vaya añadiendo al sistema al momento de desarrollarlo.

## **4.18- Herramientas para el desarrollo del software.**

Las herramientas que son utilizadas para el respectivo desarrollo de cualquier software, permiten así que los desarrolladores al mando puedan crear, depurar, gestionar y mantener el software durante todo el ciclo de vida del mismo.

### **4.18.1- Sistema gestor de base de datos.**

(Marín, 2019) Un Sistema Gestor de Base de Datos (SGBD) o DataBase Managenent System (DBMS) es un sistema que permite la creación, gestión y administración de bases de datos, así como la elección y manejo de las estructuras necesarias para el almacenamiento y búsqueda de información del modo más eficiente posible.

En la actualidad, existen multitud de SGBD y pueden ser clasificados según la forma en que administran los datos en:

* Relacionales (SQL)
* No relacionales (NoSQL)

**MYSQL.**

MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por Sun MicroSystems en 2008 y esta su vez comprada por Oracle Corporation en 2010, la cual ya era dueña de un motor propio InnoDB para MySQL.

MySQL es un sistema de gestión de bases de datos que cuenta con una doble licencia. Por una parte, es de código abierto, pero por otra, cuenta con una versión comercial gestionada por la compañía Oracle.

Las versiones Enterprise, diseñadas para aquellas empresas que quieran incorporarlo en productos privativos, incluyen productos o servicios adicionales tales como herramientas de monitorización y asistencia técnica oficial.

**Características de Mysql.**

(Robledano, 2019) MySQL presenta algunas ventajas que lo hacen muy interesante para los desarrolladores. La más evidente es que trabaja con bases de datos relacionales, es decir, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente.

Al ser basada en código abierto es fácilmente accesible y la inmensa mayoría de programadores que trabajan en desarrollo web han pasado usar MySQL en alguno de sus proyectos porque al estar ampliamente extendido cuenta además con una ingente comunidad que ofrece soporte a otros usuarios. Pero estas no son las únicas características como veremos a continuación:

* Arquitectura Cliente y Servidor:

MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento.

* Compatibilidad con SQL:

SQL es un lenguaje generalizado dentro de la industria. Al ser un estándar MySQL ofrece plena compatibilidad por lo que si has trabajado en otro motor de bases de datos no tendrás problemas en migrar a MySQL.

* Vistas:

Desde la versión 5.0 de MySQL se ofrece compatibilidad para poder configurar vistas personalizadas del mismo modo que podemos hacerlo en otras bases de datos SQL.

* Procedimientos almacenados.

MySQL posee la característica de no procesar las tablas directamente, sino que a través de procedimientos almacenados es posible incrementar la eficacia de nuestra implementación.

* Desencadenantes.

MySQL permite además poder automatizar ciertas tareas dentro de nuestra base de datos. En el momento que se produce un evento otro es lanzado para actualizar registros u optimizar su funcionalidad.

* Transacciones.

Una transacción representa la actuación de diversas operaciones en la base de datos como un dispositivo. El sistema de base de registros avala que todos los procedimientos se establezcan correctamente o ninguna de ellas.

### **4.18.2- C#.**

Es una evolución que Microsoft realizó de este lenguaje, tomando lo mejor de los lenguajes C y C++, y ha continuado añadiéndole funcionalidades, tomando de otros lenguajes, como java, algo de su sintaxis evolucionada. Lo orientó a objetos para toda su plataforma NET (tanto Framework como Core), y con el tiempo adaptó las facilidades de la creación de código que tenía otro de sus lenguajes más populares, Visual Basic, haciéndolo tan polivalente y fácil de aprender como éste, sin perder ni un ápice de la potencia original de C. En la versión de .NET Core, se ha reconstruido por completo su compilador, haciendo las aplicaciones un 600% más rápidas. (besoftware, s.f.)

**Características presentes en C#.**

Sencillez:

En comparación a los otros lenguajes antecesores de este, C# elimina cierto objetos y atributos innecesarios para que la acción de programar sea más intuitiva.

Modernidad:

Aunque hemos mencionado que su creación esta también enfocada para dar solución a los temas actuales, también el lenguaje C# realiza de manera automática e intuitiva la incorporación de algunos objetos que con el paso de los años han sido necesarios a la hora de programar.

Seguridad:

Desde unas instrucciones para realizar acciones seguras y un mecanismo muy fuerte para la seguridad de los objetos.

Sistemas de tipos unificados:

Todos los datos que se obtienen al programar el lenguaje C# quedan guardados en una base para que puedan ser utilizada posteriormente.

Extensibilidad:

Esta característica es muy positiva, debido a que puedes añadir tipos de datos básicos, operadores y modificadores a la hora de programar.

Versionable:

Dispone la característica de tener versiones, es decir, actualizarse y mejorar constantemente.

Compatible:

Tanto con sus antecesores como con Java y muchos otros lenguajes de programación, #C integra a todos estos para facilidad del programador. (Negocios y estrategias, s.f.)

# **Metodología de desarrollo.**

El sistema el cual desarrollamos es Cliente - Servidor, es decir, que el sistema se podrá usar por medio de conexión a red pata mejor experiencia de uso y de igual manera sea más sencilla las supervisiones que lleguen a dar del trabajo realizado por medio de este, por lo tanto, se utilizó la metodología Srcum, el cual consiste en aplicar un conjunto de buenas prácticas para facilitar el trabajo en equipo, consiguiendo consecuentemente buenos y mejores resultados.

En este sistema se realizan entregas parciales las cuales se denominan sprint, en cada uno de estos se desarrollaron nueve requerimientos: ventas, compra, productos, combo, inventarios, usuarios, proveedores, clientes y sucursales.

Para esto, se realizó en cada categoría correspondiente, una serie de etapas las cuales consisten en: Análisis, diseño, desarrollo y pruebas.

# **Primer capítulo. Estudio de Viabilidad.**

## **6.1- Viabilidad Técnica.**

En estudio de la viabilidad técnica se determinará si la empresa cuenta con los recursos necesarios para implementar el sistema.

A través de la entrevista elaborada a la gerente del negocio se confirmó la existencia de máquinas donde se puede instalar el sistema que se propone desarrollar, falta la confirmación de las características de hardware requeridas. Cabe destacar que el negocio no cuenta con un sistema, todo registro lo llevan en físico causando: retrasos, perdida de información, errores. Por ende, el equipo existente debe de poder cumplir con las especificaciones siguientes:

|  |  |
| --- | --- |
| Computadora | |
| Especificaciones de Hardware. | |
| Tarjeta Madre | Marca: Gigabyte.  Modelo: GA-HB1-D3  Chpset: Intel® HB1 |
| Memoria RAM | Capacidad: 4GB  Descripción: DDR3 DIMM 1333MHz |
| Procesador | Intel® Pentium G3258  Velocidad: 3.2 CHz 133MHz  Cache: 3MB |

|  |  |
| --- | --- |
| Procesador | Intel ® Pentium G3258  Velocidad: 3.2CHz 1333MHz  Cache: 3MB |
| Comunicación de red | Conexión Alámbrica: Puerto Ethernet Gigabit (10/100/1000) |
| Almacenamiento | Disco duro marca Seagate 7200RPM  Capacidad: 250 GB  Espacio libre: 150GB |
| Pantalla | Monitor AOC 17  Entrada de video: VGA |
| Periféricos | Mouse, Teclado y parlantes estéreo |
| Especificación de software | |
| Sistema operativo | Microsoft Windows® 7 SP  Arquitectura: x64(64-bits) |
| Antivirus | ESET NOD32® antivirus |

**Características de la computadora recomendada.**

|  |  |
| --- | --- |
| Computadora HP | |
| Especificación de Hardware | |
| Tarjeta madre | Marca: HP Laptop 15-bs0xx |
| Memoria Ram | Capacidad: 4GB  Descripción: DDR4 SDRAM |
| Procesador | Dual Core Intel Core i3-6006U, 2000MHz(20x10) |
| Comunicación de red | Dirección IP primaria 127.0.0.1  Dirección MAC primaria B2-52-16-8C-DC-54 |
| Almacenamiento | Disco duro 449GB |
| Pantalla | Monitor PnP genérico NoDB |
| Periféricos | Mouse, Teclado |
| Especificación de software | |
| Sistema operativo | [Microsoft Windows 10 Pro](http://www.microsoft.com/windows/)  Arquitectura: x64(64-bits) |
| Antivirus | Windows Defender |

Cabe destacar que es necesario comprar un equipo con la información proporcionada ya que el sistema necesita un equipo con esas especificaciones para no tener inconvenientes algunos en la ejecución.

**Análisis de vida útil.**

El único equipo existente dentro del negocio Jamen´s Pollos tiene un estimado de 2 años en lo que respecta su vida útil, y se estima que el tiempo transcurrido desde la realización de este estudio hasta la actualidad data de seis meses.

Cabe destacar que por lo general este equipo no les dan uso referente a trabajos del negocio.

**Estado del ordenador.**

La actual computadora en donde cual realizamos el análisis nos da como resultado que se encentra en un buen estado, ya que este equipo a no ser tan activo en lo que se refiere a uso de ella recibe su respectivo mantenimiento en tiempo y forma. Esto nos beneficiará ya que ayudará a que el sistema funcione con normalidad, que cumpla con todos los requerimientos propuestos de sus funciones a realizar.

**Requerimientos de herramientas de desarrollo.**

Es vital recopilar los requerimientos de las herramientas que se van a utilizar para el desarrollo del sistema de información, ya que se necesita estimar los requerimientos del sistema para el equipo del cliente.

|  |  |
| --- | --- |
| Herramienta | Microsoft ® Visual Studio 2019 |
| Requerimientos del sistema | * Recomienda 64 bits, ARM no se admite. * Windows 10 versión 1703 * Procesador de 1.8 GHz o superior * 2 GB de Ram; 8 GB de Ram recomendado * Espacio disponible disco duro 800 MB hasta 210 GB |

|  |  |
| --- | --- |
| Herramienta | Microsoft. NET Framework |
| Requerimientos del sistema | * Sistema operativo 8 o 10 * Recomienda 32 o 64 bits, ARM no se admite. * Procesador de 1.8 GHz o superior * 2 GB de Ram; 8 GB de Ram recomendado * Espacio disponible disco duro 800MB hasta 210 GB |

|  |  |
| --- | --- |
| Herramienta | Microsoft. MYSQL. |
| Requerimientos del sistema | * Sistema operativo 8 o 10 * Recomienda 32 o 64 bits, ARM no se admite. * Procesador de 1.8 GHz o superior * 2 GB de Ram; 8 GB de Ram recomendado * Espacio disponible disco duro 800MB hasta 210 GB |

**Requerimientos para la ejecución del sistema.**

Ya una vez analizado los requerimientos de las herramientas que se utilizaran para el desarrollo del sistema, se estima que los requisitos de los ordenadores en los que funcionara son los siguientes.

|  |  |
| --- | --- |
| Componentes | Requerimientos |
| Equipo | Procesador Dual Core Intel Core i3-6006U, 2000MHz(20x10) |
| Memoria RAM | 512 megabytes disponibles, se recomienda 2GB |
| Almacenamiento digital | 200 MB espacio mínimo de disco duro |
| Pantalla | Monitor PnP genérico NoDB |
| Especificaciones de software | |
| Sistema operativo | [Microsoft Windows 10 Pro](http://www.microsoft.com/windows/)  Arquitectura: x64(64-bits), instalado Net Framework |
| Periféricos | Mouse teclado |

## **6.2- Viabilidad Operativa**

Para determinar la viabilidad operativa se analizó cuáles serían los impactos que tendría el sistema de información para controlar los procesos. La idea surge por la necesidad que se logró notar en ver como hacían los procesos y se ve la necesidad de un sistema que agilice y guarde la información de manera más confiable.

Debido a este problema el sistema se enfoca en resolver y satisfacer dichas necesidades y beneficios el cual mantendrá la información guardada de los años pasados y de los próximos años.

Este sistema presentará una interfaz amigable con el usuario de tal manera que sin mucha dificulta podrá ser usado y adaptarse fácilmente para aprovechar al máximo las facilidades que este estar brindando.

Respecto al personal que se verá afectado por el software se notó lo siguiente:

* En el caso de uso de usuario, el sistema les permitirá realizar sus tareas de una forma más segura, rápida y ordenada.
* En el caso del administrador le permitirá agregar nuevos usuarios y darles privilegios según lo que él quiera.
* De igual manera se espera que los usuarios se adapten al sistema ya que el personal actualmente no hace uso de sistemas de información todo se hace de manera tradicional.

En Jamen’s pollos laboran una cantidad total de dos personas, en donde una persona cumple con lo que es la administración (dueña) y la otra persona es quien realiza las ventas de los productos que ofrecen en dicho negocio.

Ambas personas que laboran en Jamen’s poseen un nivel educativo apto para el buen funcionamiento del sistema.

## **6.3- Viabilidad Económica.**

Modelo constructivo de costo (COCOMO)

Este modelo permite realizar estimaciones del software de un conjunto de factores de costos y escala. Los principales factores de costos se describen según los aspectos relacionados con la naturaleza del producto. El conjunto de factores de escala explica las economías producidas a medida que se incrementa un proyecto.

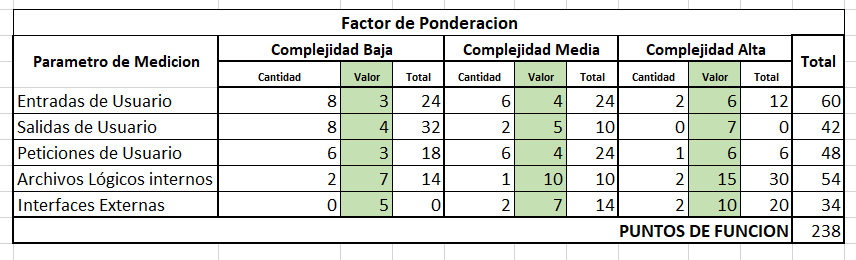
**Valores de ajustes de complejidad.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Valores de ajuste de complejidad (Fi) | | | | | | |
| Escala | 0 | 1 | 2 | 3 | 4 | 5 |
| Valor | Sin Influencia | Incidente | Moderada | Medio | Significativo | Esencial |

|  |  |
| --- | --- |
| Preguntas. | |
| ¿Requiere el sistema copias de seguridad y de recuperación fiables? | 4 |
| ¿Se requiere de comunicación de datos? | 5 |
| ¿Existe funciones de procesamiento distribuidos? | 4 |
| ¿Es crítico el rendimiento? | 5 |
| ¿Se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado? | 2 |
| ¿Requiere el sistema entrada de datos interactiva? | 4 |
| ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones? | 3 |
| ¿Se actualizan los archivos maestros de forma interactiva? | 3 |
| ¿Son complejas las entradas, salidas, archivos o las peticiones? | 2 |
| ¿Es complejo el procesamiento interno? | 4 |
| ¿Se ha diseñado el código para ser reutilizable? | 4 |
| ¿Están incluidas en el diseño la conversión y la instalación? | 3 |
| ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones? | 5 |
| ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario? | 4 |
| TOTAL | 52 |

**Puntos de función sin ajustar (PF)**

L os puntos de función es el que permite la métrica en el número de funcionalidades que tendrá el software desde el punto de vista de los usuarios.



**Punto de función ajustado (PFA).**

Una vez tenido los puntos de función sin ajustar podemos hacer uso de la fórmula para calcular los puntos de función los cuales representan la cantidad de unidades de software del proyecto, con el valor obtenido se pueden hacer estimaciones de plazo, costo, recursos entre otros factores.

𝑷𝑭𝑨 = 𝑷𝑭 ∗ [𝟎. 𝟔𝟓 + (𝟎. 𝟎𝟏 ∗ ∑𝑭𝒊)]

Dónde:

* PFA será el valor de los puntos de función ajustados.
* PF es el total de puntos de función sin ajustar.
* 0.65 y 0.01 son constantes de la formula.
* ∑Fi es la sumatoria de los valores ajustados de complejidad.

Suplementando el valor de la formula se obtuvo un resultado de PFA= **278**.

**Total, de líneas de código.**

Para determinar el esfuerzo nominal usando el modelo COCOMO los puntos funcionales tienen que convertirse en lianas de código, teniendo en cuenta la fuente del lenguaje que se utilizara para programar, en el caso de este sistema se usara el lenguaje de visual C# el que toma un valor de 20 puntos. Una vez que se determina el punto de función del lenguaje en el que se programa se procede a calcular las líneas de códigos totales con la formula.

𝑳𝑫𝑪 = 𝑳𝑷 ∗ 𝑷𝑭A

Dónde:

* LDC será el total de líneas de código del proyecto.
* LP es el valor de función del lenguaje con el que se desarrollará el sistema.
* PFA son los puntos de función ajustado

Haciendo los cálculos correspondientes obtuvimos que el sistema tendrá las siguientes líneas de código.

5069.2 = 5.569 Mf (Este valor ha de cambiar con la revisión de Cocomo)

**Factores de escala.**

Los modelos estimados de costos analizan dos aspectos que incluyen notablemente en los procesos de estimación de la economía de escala. Estos modelos frecuentemente tienen un valor exponencial para considerar la economía, en particular el cocomo captura esos valores.

|  |  |  |
| --- | --- | --- |
| **Factores de Escala.** | | |
| **Nombre** | **Valor** | **Observación.** |
| PREC | 2.48 | Bastante Parecido. |
| FLEX | 2.03 | Acuerdo General. |
| RESL | 1.41 | Se identifican pocos riesgos, 23% del tiempo de desarrollo. |
| TEAM | 1.1 | Interacciones altamente cooperativas. |
| PMAT | 4.68 | Nivel repetible. |
|  | 11.17 |  |

Ya una vez teniendo la sumatoria de los factores de escala, se aplica la fórmula para encontrar el exponente B.

Donde:

* B, será el exponente de escala.
* ∑ 𝑆𝐹, es la sumatoria de los factores de escala.
* 0.91 y 0.01 son constantes de la fórmula

B= 1,027

**Esfuerzo personal al mes.**

Una vez obteniendo el valor del producto se hace una estimación de esfuerzo de personas al mes, las que serán necesarias para desarrollar el proyecto, se estima que se necesitarán 3 personas al mes para desarrollar el proyecto.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | TLDC | B |  |  |  |  |
| Esfuerzo (E) | 2.94 | 5.5692 | 1.027 | 0.3367684 | = | 5.7757431 | Personas / mes |

**Tiempo de desarrollo.**

Cuando ya se tiene el esfuerzo necesario para el desarrollo del sistema se necesita proyectar en cuanto tiempo este estará listo funcionando, el cual es medido en mese, este sistema estará siendo desarrollado en 5 meses

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tdes= |  | E |  |  |  |
| 3.67 | 5.7757431 | 11.7 | = | 6.247955 mes |

**Cantidad de personal necesario.**

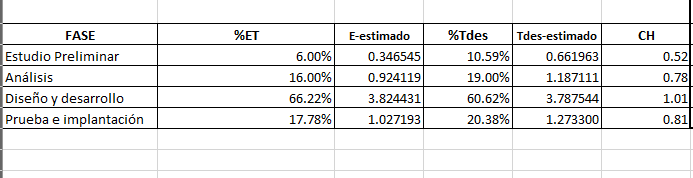
Cuando ya se tiene calculado el esfuerzo y el tiempo de desarrollo necesario para el sistema se calcula la cantidad de personas necesarias que deberán participar en el proyecto.

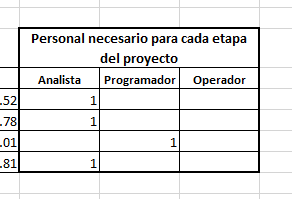
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CH= | E | Tdes | CH |  |  |
| 5.7757431 | 6.247955 | 0.9244214 | = | 1 persona |

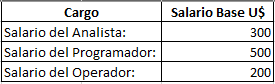
El porcentaje de esfuerzo de cada etapa en el desarrollo del proyecto mide y categoriza los niveles desde el más pequeño al más grande al igual que el nivel de complejidad.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Tamaño.** | **Pequeño.** | **Intermedio.** | **Medio.** | **Grande.** |
| Indicador | Fases. | 2 | 8 | 32 | 128 |
| Esfuerzo. | Estudio Preliminar. | 6 | 6 | 6 | 6 |
| Análisis. | 16 | 16 | 16 | 16 |
| Diseño y desarrollo. | 68 | 65 | 62 | 59 |
| Prueba e implementación. | 16 | 19 | 22 | 25 |
| Tiempo de Desarrollo | Estudio Preliminar. | 10 | 11 | 12 | 13 |
| Análisis. | 19 | 19 | 19 | 19 |
| Diseño y desarrollo. | 63 | 59 | 55 | 51 |
| Prueba e implementación. | 18 | 22 | 26 | 30 |

Según el personal necesario para el desarrollo calculado por medio del cocomo, se estima lo siguiente:







Se ha tomado un salario de $600 para el analista ya que en Nicaragua el salario de un analista de software ronda entre los $270 salario mínimo hasta y los $830, el salario que hemos establecido para el analista es porque contrataremos pernal con experiencia esto para tener una firmeza en el desarrollo. (Tusalario.org/Nicaragua, s.f.)

Como programadores nos hemos puesto un salario de $500 ya que somos desarrolladores con poca experiencia, también se ha decidido establecer ese salario puesto que en Nicaragua el salario mínimo de un programador es de $260 como minimizo y hasta unos $745 esto depende de la experiencia. (Tusalario.org/Nicaragua, s.f.)

**Costos de uso de medios técnicos (CUMT)**

El costo de uso de medios técnicos abarca el costo de utilización de los equipos de trabajo, consumo de energía, costos de mantenimiento preventivo.

|  |  |
| --- | --- |
| **Supuestos** |  |
| Costo del Equipo. | 1,176 |
| Cantidad de Equipos. | 1 |
| Costo del KW/H. | 0.10 |
| Cantidad de Horas Trabajadas. | 4 |
| Hora de Tiempo Maquina. | 499.84 |
| CRM. | 0 |
| PRk. | 17,280 |
| Cek. | 0.04503 |
| Cantidad de Mantenimiento. | 1 |

El costo del equipo es de $1176, se comprar un computador que sea para desarrollo para que al momento de estar trabajando no se tengan dificultas como que se pegue, se caliente, etc,

El costo del kw/h lo tenemos representado a una tarifa de 0.10 ya que la potencia de consumo del computador es de 3.6kw, al hacer los cálculos de consumo obtuvimos esa cantidad de consumo por horas.

**Costo de abastecimiento técnico de materiales (CMAT)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Costo de abastecimiento técnico de materiales (CMAT)** | | | |
| **Cantidad.** | **Descripción.** | **Precio** | **Total** |
| 0.1 | Resma de Papel Carta. | 140 | 14 |
| 3 | Folders Carta. | 2 | 6 |
| 2 | Lapiceros. | 7 | 14 |
| 2 | Lápiz de carbón. | 5 | 10 |
| 1 | Borrador. | 6 | 6 |
| 0.5 | Cartucho de Tinta. | 620 | 310 |
| 1 | Libreta. | 20 | 20 |
| 1 | Engrapadora pequeña. | 15 | 15 |
| 0.2 | Caja de Grapa. | 35 | 7 |
| 1 | Tablas con clan. | 50 | 50 |
|  | | **Sub Total** | **452** |
| **IVA** | **0** |
| **Total** | **452** |

Todos los materiales descritos en la tabla CMAT son de gran utilidad esto para hacer apuntes etc, este material es una base para ir diseñando y anotar cada observación que se haga a futuros cambios a medida que se desarrolla, los costos son bastante bajos, pero si es esencial contra con estas herramientas que serán de gran ayuda.

**Costo del Proyecto (CTP)**

El costo total del proyecto se calcula sumando los costos directos y los costos indirectos. El costo total del sistema vale $ 4,232.23 dólares

|  |  |
| --- | --- |
| Costos Directos. | $4,642.65 |
| Costos Indirectos. | $464.27 |
| **CTP=** | **$5,106.92** |

## **6.4- Viabilidad Financiera.**

Un estudio financiero tiene como objetivo evaluar los recursos económicos que se necesitan para llevar a cabo un proyecto, con la información obtenida durante el estudio de la viabilidad financiera del proyecto esta información será la fuente donde se obtendrán los datos para los análisis de riesgo.

A continuación, se muestra el flujo de efectivo del proyecto.

**Factores de evaluación del proyecto de inversión.**

* **Valor actual neto (VAN)**

El valor actual neto (VAN) es simplemente la suma de todos los flujos proyectados a los que se les resta la inversión inicial, los criterios para evaluar si la inversión es aceptable o no son los siguientes si VAN>0 la inversión es aceptada y si VAN≤0 no se debería aceptar la inversión.

El VAN de este proyecto es de C$7,192.47 este valor es mayor que cero por lo tanto se acepta la inversión.

* **Tasa de rendimiento (TIR)**

La tasa de retorno hace que el descuento de la suma de flujo de efectivos sea igual a la inversión inicial por lo tanto se iguala VAN a cero. El resultado obtenido del TIR es de 47% por lo tanto se concluye que el proyecto es aceptable.

El estado financiero es un instrumento que permite saber con exactitud el estado económico de una empresa con la finalidad de hacer la correcta toma de decisiones y evaluar la factibilidad de proyectos de inversión.

## **6.5- Viabilidad Legal.**

Con el propósito de determinar la viabilidad legal del sistema de información se tomaron en cuenta dos cosas la reservación de derecho de autor y la integridad de la información. El equipo desarrollador guardara confidencialidad y silencio profesional sobre la información facilitada por la empresa para la ejecución del proyecto, el equipo desarrollador está comprometido a proporcionar el instalador del cliente con sus respectivos manuales de uso y de instalación.

Si la empresa pide al equipo desarrollador la implementación estos acudirán con el material necesario, adecuado y actualizado para instalar el producto mejorado.

Según el suscrito director general de Ingresos en uso de las facultades que le confieren el Arto 223 numeral 3 de la Ley 562 Código Tributario de la República de Nicaragua (CTr.) y sus reformas publicado en la Gaceta Diario Oficial No. 177 del 23 de noviembre del 2005, considera:

1. Que es necesario regular y definir el procedimiento para autorización y uso del sistema de facturación computarizado de los contribuyentes y responsables, dedicados a cualquier actividad económica que deseen utilizar este sistema a fin de soportar sus transacciones por ventas de bienes, prestación de servicios e importaciones.
2. Que los contribuyentes o responsables conforme lo establecido en el Arto 81 y 126 numeral 4 del CTr. podrán solicitar autorización para utilizar sistemas en medios electrónicos para emitir sus facturas de las actividades y operaciones que se vinculen con los tributos.

Por ende:

**PRIMERO:**

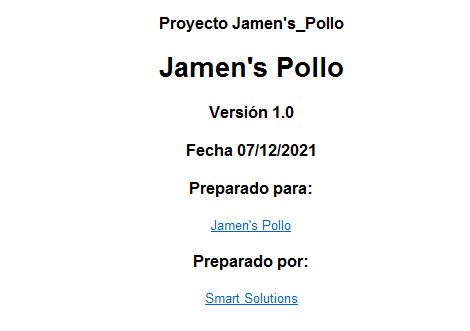
De conformidad al Arto.81 CTr. todo contribuyente o responsable podrá hacer uso de medios electrónicos de información, para facturar sus enajenaciones, servicios en general e internaciones previa autorización avalada por auditores del Departamento de Fiscalización de la Dirección de Grandes Contribuyentes o Administración de Rentas según corresponda.

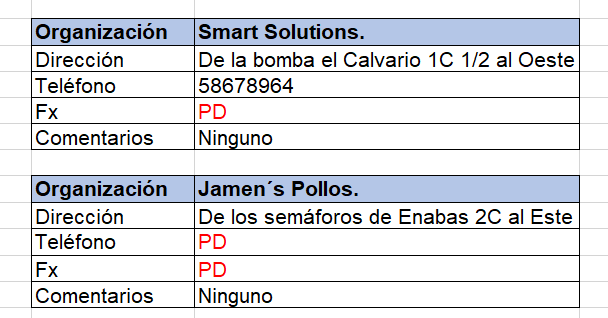
**SEGUNDO:**

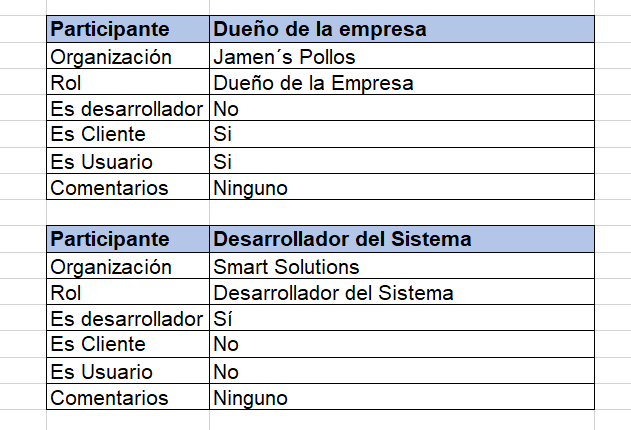
La solicitud de autorización del uso de Sistemas de Facturación Computarizado deberá cumplir con los requisitos siguientes:

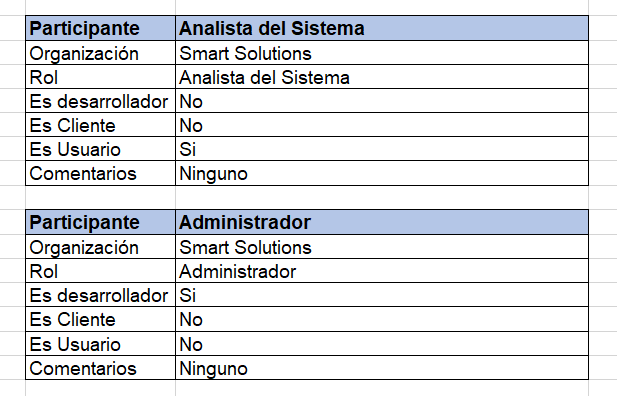
|  |  |
| --- | --- |
| **Primer día del mes en que dará inicio con la aplicación del sistema señalando (día. mes y año)** |  |
| **Nombre del sistema.** | Jamen´s Pollos. |
| **Lenguaje en que fue desarrollado.** | C# y MySql como gestor de base de datos. |
| **Informar si poseen programas fuentes.** |  |
| **Informar si el sistema está integrado a otros módulos, indicando cada uno de ellos.** |  |
| **Nombre del autor y/o proveedor del sistema, número RUC, número de cédula de identidad si es persona natural, dirección, país, teléfono, correo electrónico. El número RUC y cédula de identidad no serán necesarios si el proveedor es extranjero** |  |
| **Fecha en que fue adquirido el Sistema.** |  |
| **Seguridad de Acceso.** | Usuario Mayor:  Nombre de Usuario: Saul  Clave: Artola. |
| **Respaldo de Soportes.** |  |
| **Manuales.** | Si posee un manual de usuario, en donde se especifica cada detalle presente en el sistema y de igual manera los pasos que debe seguir para comprender su funcionamiento. |
| **Notificación de Imprevistos.** |  |
| **Numeración de Facturas.** |  |
| **Formalidades de Facturas.** |  |
| Desglose de Impuestos. |  |
| **Anulación de Facturas.** | Toda factura que se anule por cualquier razón, deberá conservarse la factura original con todas las copias preimpresas sin excepción alguna, estampando en cada una de ellas el sello de anulado. |
| **Cantidad de Copias.** |  |
| **Sucursales o agencias.** |  |
| **Recibos de caja.** |  |
| **Plan de contingencia.** |  |
| **Devoluciones y/o anulaciones.** |  |

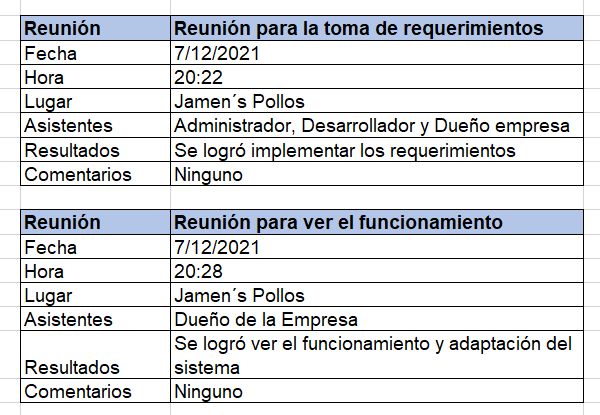
# **Segundo Capítulo. Ingeniería de Requerimientos (REM)**

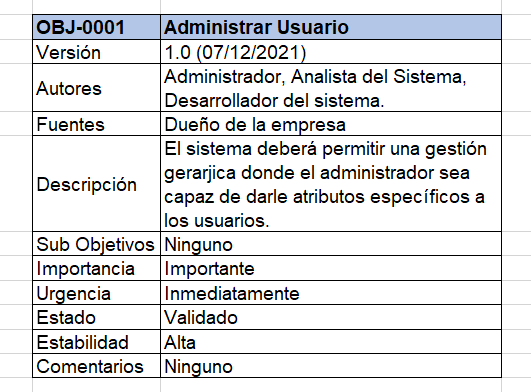
En este capítulo se abordará los requerimientos manera específica para el sistema de ventas de Jamen´s Pollos.

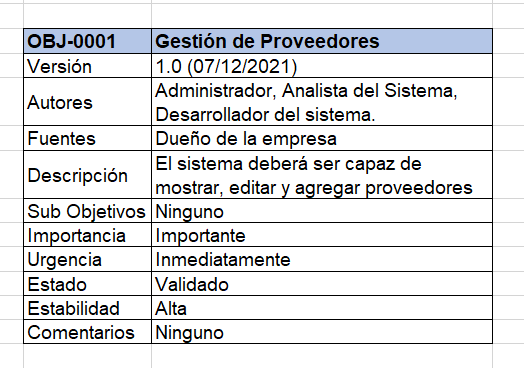


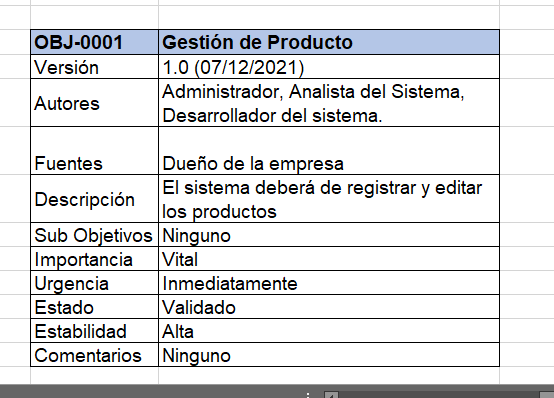


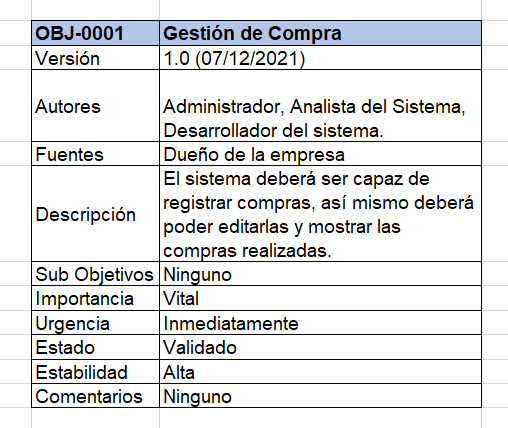


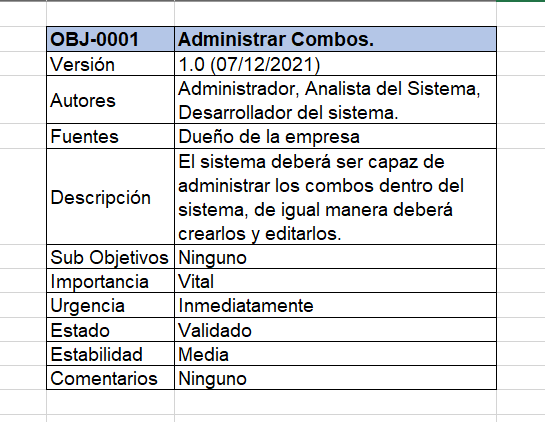


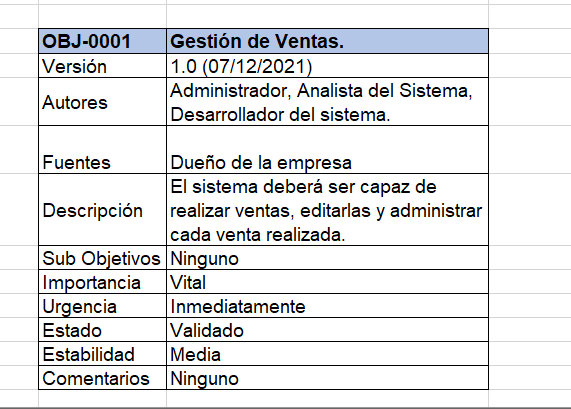


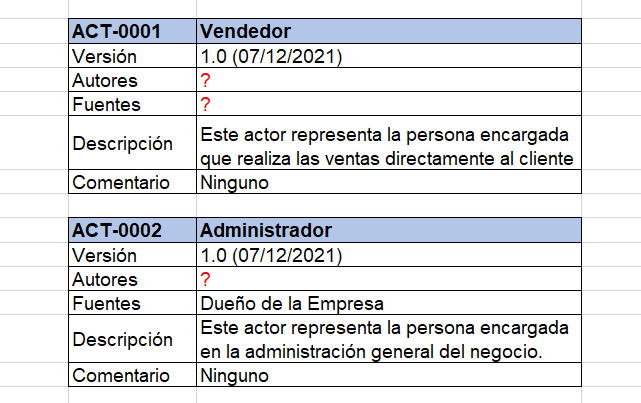


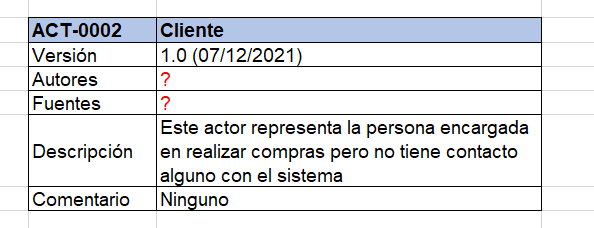


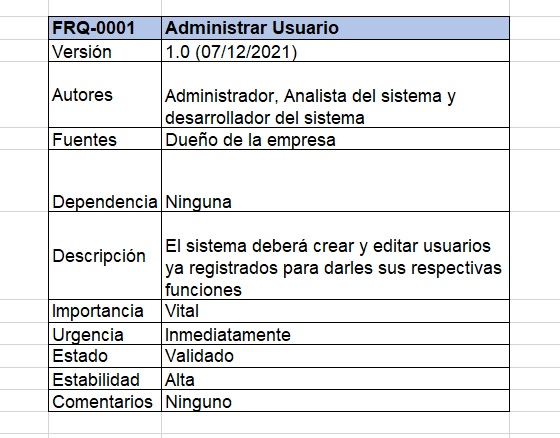


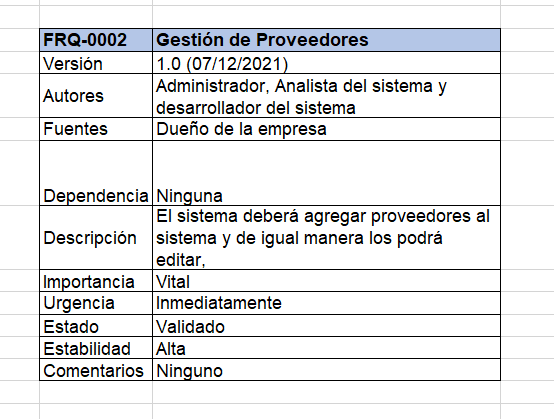


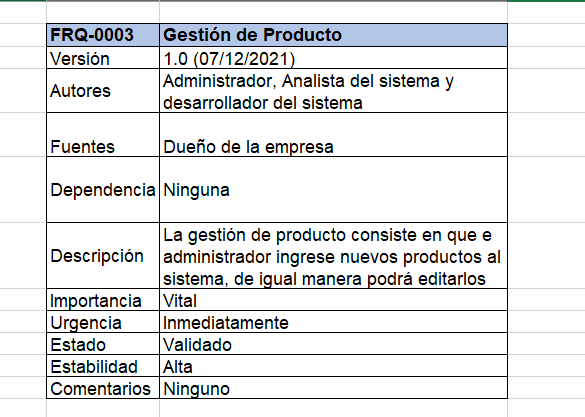


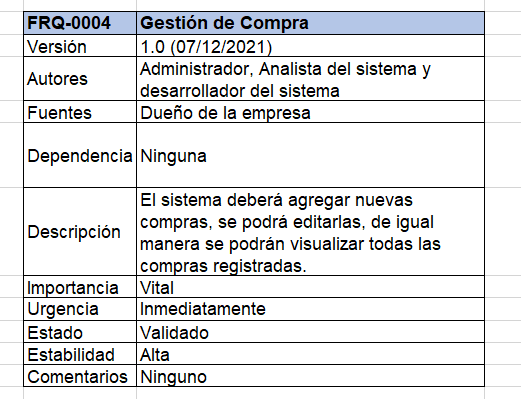


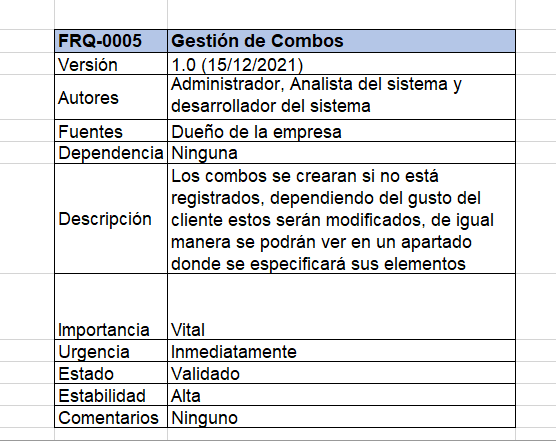


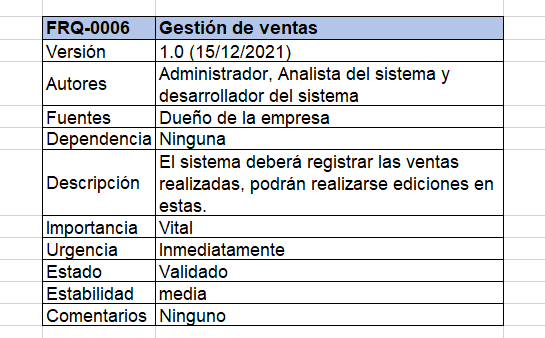












## **7.1- Scrum.**

|  |  |
| --- | --- |
| **Definir que se quiere conseguir.** | Se pretende obtener lo siguiente:   * Un sistema que tenga gestión de Usuarios, proveedores y clientes. * Creaciones de productos y combos. * Que realice ventas de los productos que tenga por medio de los combos. * Que pueda hacer compras a sus sucursales para el abastecimiento de productos. * Gestiones de sucursales, etc   Se pretende que sea fácil de usar para el usuario y multi dinámico. |
| **Organización del Grupo.** | El grupo de desarrollo tendrá los siguientes elementos:   * Desarrollador. * Analista. * Administrador. |
| **Organización de Prioridades.** | El desarrollo del sistema se vendrá presentando por etapas, es decir semana con semana se estará dando las presentaciones de los requerimientos dispuestos a trabajar. |
| **Planificación del primer Sprint.** | Como inicio de cada desarrollo, se elige la implementación inicial del Primer Sprint a:   * Gestión de Usuarios.   En donde se puedan mostrar todas las funciones que esta área tendría. |
| **Desarrollo y Ejecución.** | Se llevarán a cabo reuniones dos días por la semana para ir creando el desarrollo del sistema. |
| **Revisión de Primeros resultados.** | Teniendo ya el primer prototipo del sistema, se ha de revisar si funciona correctamente tal cual era el resultado que habíamos esperado para cada incremento. |
| **Retrospectiva del proyecto** | Según los resultados obtenidos de esta revisión se planteará un esquema de actividades a realizar para implementarlas y así dar mejoras necesarias a algunos detalles que puedan salir más y así mejorarlo al máximo. |
| **Ciclo de Sprints** | Se empezarán a implementar nuevos incrementos al sistema. |

# **Tercer capítulo. Diseño y desarrollo.**

**Diagrama general de casos de uso del sistema.**

Product Backlog.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Prioridad | Título | Descripción |
| PBI01 | 3 | Administración de usuario | El requerimiento consiste en que el administrador principal del sistema sea capaz de añadir, modificar y permitir el acceso a usuarios del sistema. |
| PBIO2 | 3 | Gestión de Proveedores | Esta funcionalidad debe de permitir al usuario agregar y editar proveedores |
| PBIO3 | 4 | Gestión de Producto | La función de este requerimiento es poder ingresar al sistema productos, en este caso de dos tipos: materia prima e insumos, cada uno con sus características propias. Además, se debe de poder editarlo. |
| PBIO4 | 5 | Gestión Compras | El usuario debe de ser capaz de registrar compras de los productos existentes en el sistema, también editarlas por cualquier error, se debe de llevar control preciso de quien y cuando se hicieron las compras |
| PBIO5 | 5 | Administración de Combos | El sistema debe de permitir el ingreso de combos los cuales necesitan ingredientes previamente ingresados en el sistema, así como la edición de estos combos. |
| PBIO6 | 5 | Gestión de ventas | El sistema debe de realizar la venta de los combos ingresados en el sistema, así como los productos extras que se pueden vender de manera individual. También debe de editarse las ventas con un plazo no mayor a 5 días. |
| PBIO7 | 5 | Clientes | El sistema se encargará de registrar a los clientes para un mejor control de información. |
| PBIO8 | 5 | Sucursales | El sistema debe registrar sucursales nuevas para la expansión del negocio. |

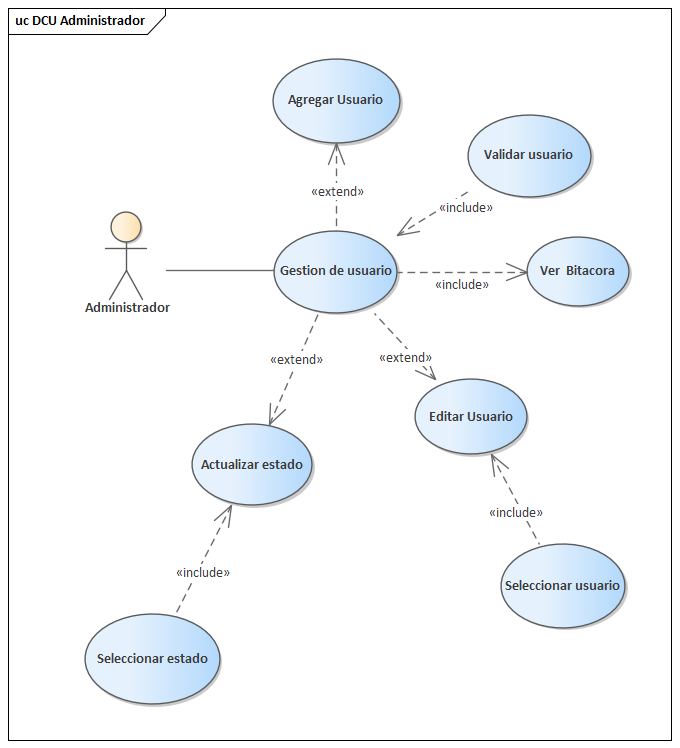
**Incremento #1. [Gestión de usuario]**

La validación de usuarios consiste en poder acceder de manera correcta y satisfactoria al sistema, siempre y cuando cumpla con los requisitos solicitados (Nombre de Usuario y Contraseña) para que de esta manera tenga acceso a todo el sistema, pero en este caso específicamente al ítems de Usuario, en donde podrá añadir usuarios al sistema única y exclusivamente si su rol es administrador, esto va como parte de los permisos a los usuarios y de igual manera mostrar usuarios que ya estén registrados en el sistema, en donde podrá editarlos, habilitar y deshabilitar.

**Pila del Sprint del incremento.**

**Escenarios de Casos de Uso.**

**Diagrama de casos de uso para gestionar usuarios.**

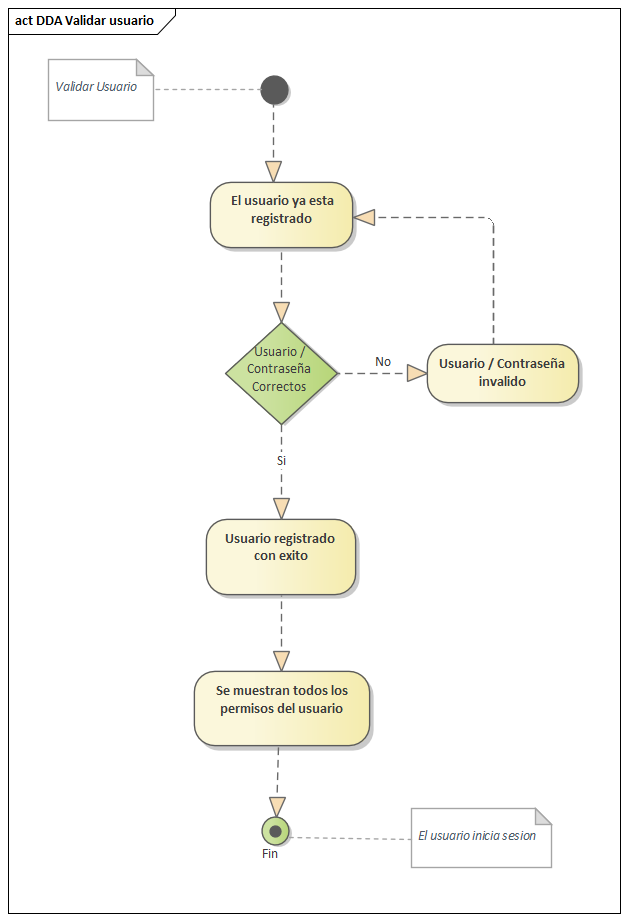
****

Diagramas Caso de uso gestión de usuario

Este diagrama general de caso de uso orientado a la gestión de usuarios, data más que todo en la debida validación de usuarios para respectivamente llevar a cabo las funciones correspondientes a lo que respecta al usuario como lo es: agregar usuario nuevo, mostrar usuarios, cambiar estado del usuario y editar sus datos.

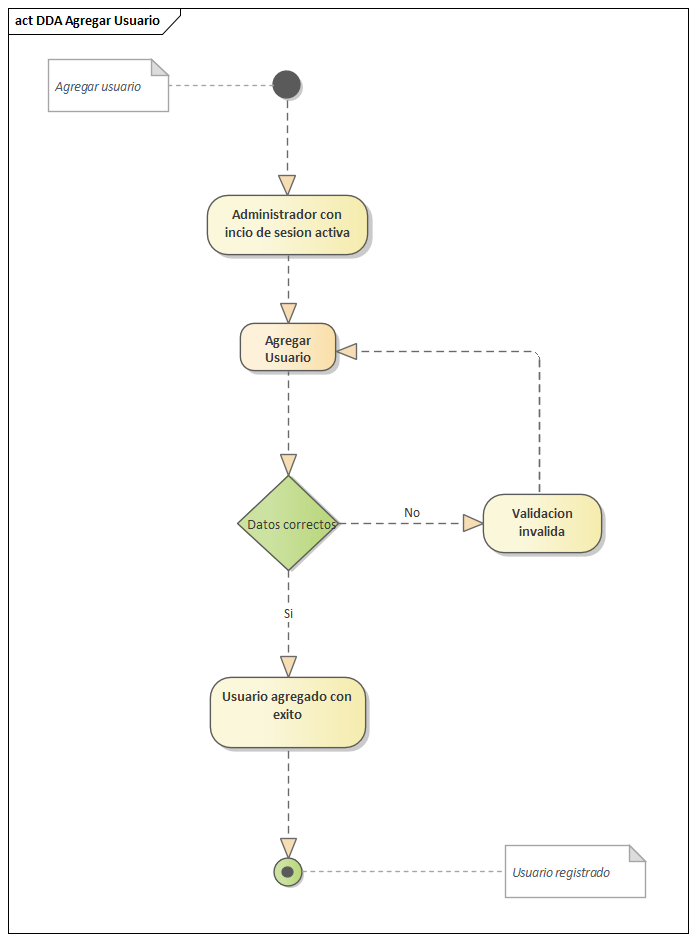
**Diagramas de Actividades.**

**Diagrama de actividad para validar usuario.**

****

Diagramas Diagrama de actividad validar usuario

**Diagrama de Actividad para agregar usuarios.**

****

Diagramas Diagrama de actividad agregar usuar

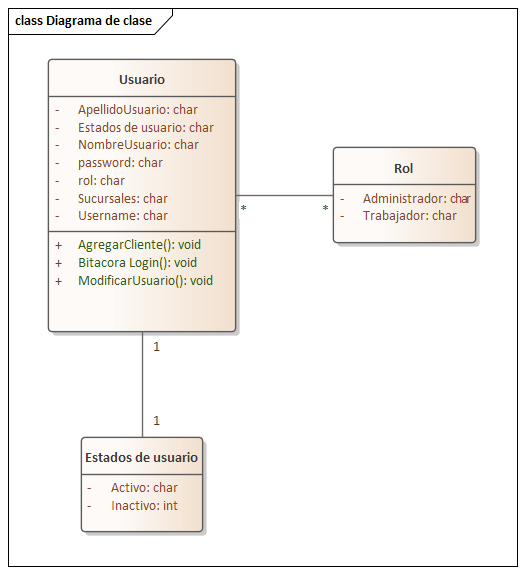
**Plantillas de Caso de uso.**

|  |  |
| --- | --- |
| **Caso de Uso.** | Escenario Validación de Usuarios. |
| **Actores.** | Administrador y vendedor. |
| **Descripción.** | Al instante de que los datos correspondientes del usuario sean ingresados: nombre de usuario y contraseña, el sistema ha de verificar si dichos datos son válidos o no, en caso de que estos sean válidos el usuario obtendrá los privilegios adaptados según su rol. |
| **Pre Condición.** | El usuario procederá a dar inicio al sistema y de tal manera ejecutará los pasos para así poder tener acceso al sistema. |
| **Pasos.** | 1. El usuario da inicio a la ejecución del sistema. 2. El sistema muestra un formulario para el correspondiente inicio de sesión. 3. El usuario deberá introducir sus datos (Nombre de usuario y contraseña). 4. Solicitud de parte del usuario al sistema para verificar si los datos son correctos. 5. El sistema verificará si los datos ingresados por el usuario son reconocibles en la base de datos. 6. Si los datos son correctos tendrá acceso al sistema. 7. Si los datos son incorrectos no tendrá acceso al sistema y permanecerá en el Login. 8. El sistema mostrará un mensaje que diga: “No existen coincidencias”. |
| **Post condición.** | Por lógica, si los datos que el usuario ingresará son válidos este tendrá acceso al sistema (según su rol podrá tener acceso a determinados ítems del sistema), de lo contrario permanecerá en el Login para que verifique sus datos. |

|  |  |
| --- | --- |
| **Caso de Uso.** | Escenario Agregar usuario nuevo. |
| **Actores.** | Administrador. |
| **Descripción.** | La función de este es agregar un nuevo usuario al sistema. |
| **Pre Condición.** | La persona encargada en administrar todo el sistema, decide agregar un nuevo usuario al sistema. Para que esto se pueda llevar a cabo el usuario no debe de existir dentro de los datos del sistema. |
| **Pasos.** | 1. Ejecutar el sistema y realizar la validación de usuario correspondiente. 2. Ya dentro del sistema el administrador deberá dirigirse al ítem de usuarios y ejecutar Agregar Usuarios. 3. Ya habiendo ejecutado esta opción, el usuario ingresará los datos que el sistema solicita para así poder agregar un usuario nuevo. 4. Con todos los datos solicitados ya ingresados, el administrador procederá a ejecutar la opción de guardar. 5. Si todos los campos dentro del formulario están debidamente completos con la información necesaria y correcta, el sistema agrega al nuevo usuario. 6. El sistema envía los datos ingresados a la base de datos. 7. El sistema valida correctamente los datos del nuevo usuario. 8. Mensaje de aviso: “Se ha insertado el registro de manera exitosa”. 9. Si hay falta de datos en los campos del formulario y el administrador ejecuta la opción de Guardar, aparecerá un mensaje: “Falta de datos para completar la validación”. 10. Si los datos son repetidos y el usuario ejecuta guardar, aparecerá mensaje: “Ya existe un registro con el nombre de usuario que escribió”. |
| **Post condición.** | Se ha agregado el nuevo usuario al sistema y ya podrá hacer uso de este. |

|  |  |
| --- | --- |
| **Caso de Uso.** | Escenario Mostrar usuario. |
| **Actores.** | Administrador. |
| **Descripción.** | Por medio de esta opción se podrá visualizar a todos los usuarios que han sido ingresados al sistema, estos divididos en dos listados: usuarios activos y usuarios inactivos, donde a la vez podrán realizarse ediciones de información a usuarios seleccionados y también cambiarles el estado a estos, es decir de activos a inactivos y viceversa. |
| **Pre Condición.** | El administrador podrá llevar un control de los usuarios ingresados al sistema y a la vez ver el estado en que se encuentran, los debidos cambios a estos se harán únicamente si se presentan casos en donde amerite aplicar estos cambios. |
| **Pasos.** | 1. Ejecutar el sistema y realizar la validación de usuario correspondiente. 2. Ya dentro del sistema el administrador deberá dirigirse al ítem de usuarios y ejecutar Mostrar Usuarios. 3. Ejecutado la opción de Mostrar Usuarios aparecerá la lista de los usuarios que tienen acceso al sistema. 4. La lista de usuarios se divide en dos partes: Usuarios activos y Usuarios inactivos. 5. Los usuarios activos son los únicos que tienen acceso al sistema. 6. Los usuarios inactivos no tienen acceso ya al sistema. 7. El usuario seleccionará al usuario que desee realizarle alguna modificación. 8. Si al usuario seleccionado le desea actualizar los datos que este tiene, solo ejecuta el botón editar donde el sistema lo llevará a la interfaz de Registro de usuarios. 9. Realizado el cambio de información (ya sea nombre, apellido, rol, nombre de usuario o clave) ejecuta el botón guardar. 10. El sistema manda mensaje de “Se ha modificado el registro de manera exitosa” y los cambios en el usuario aparecen en el listado de usuarios activos. 11. Si lo que el administrador desea es cambiar el estado de un usuario, solo selecciona el nombre y ejecuta la acción que desea (activar o desactivar). |
| **Post condición.** | Se realizan cambios en el sistema en lo que son los usuarios para su respectiva funcionalidad o así mismo dejar de funcionar dentro del sistema. |

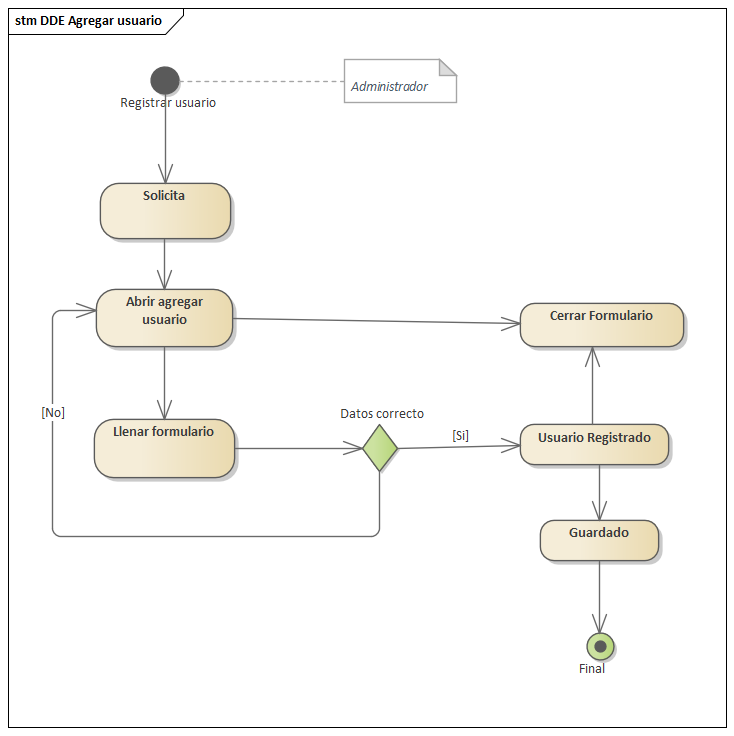
**Diagrama de Clase.**

****

Diagramas Diagrama de clase Gestión de usuario

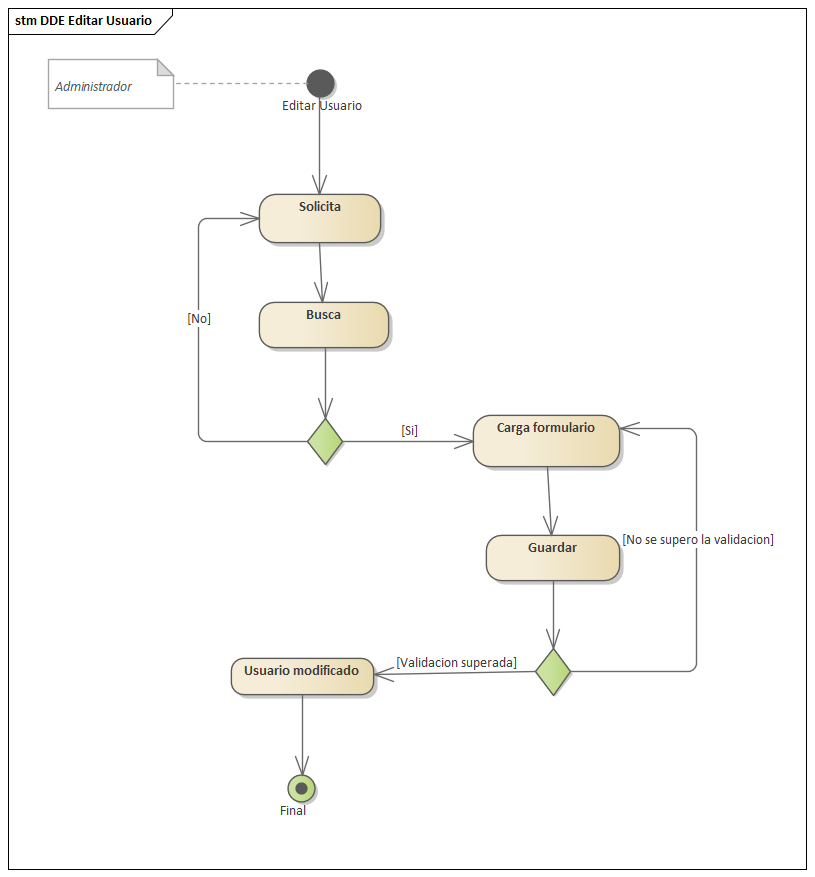
**Diagrama de Estados.**

**Agregar Usuario.**

****

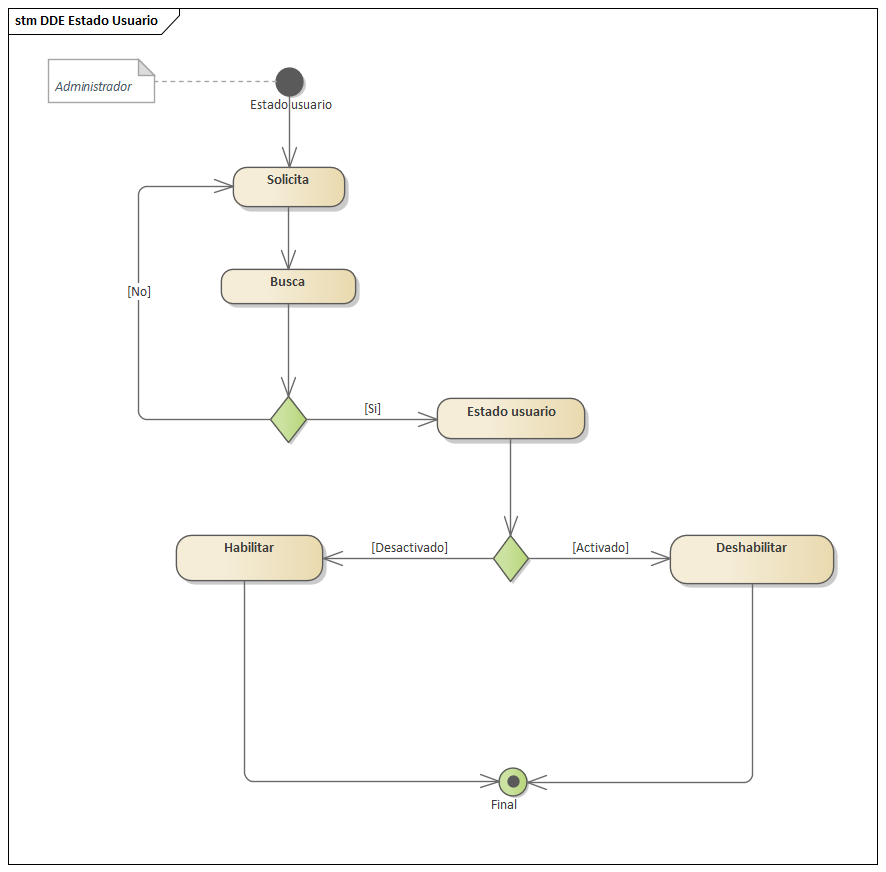
Diagramas Diagrama de estado agregar usuario

**Editar Usuario.**

****

Diagramas Diagrama de estado editar usuario

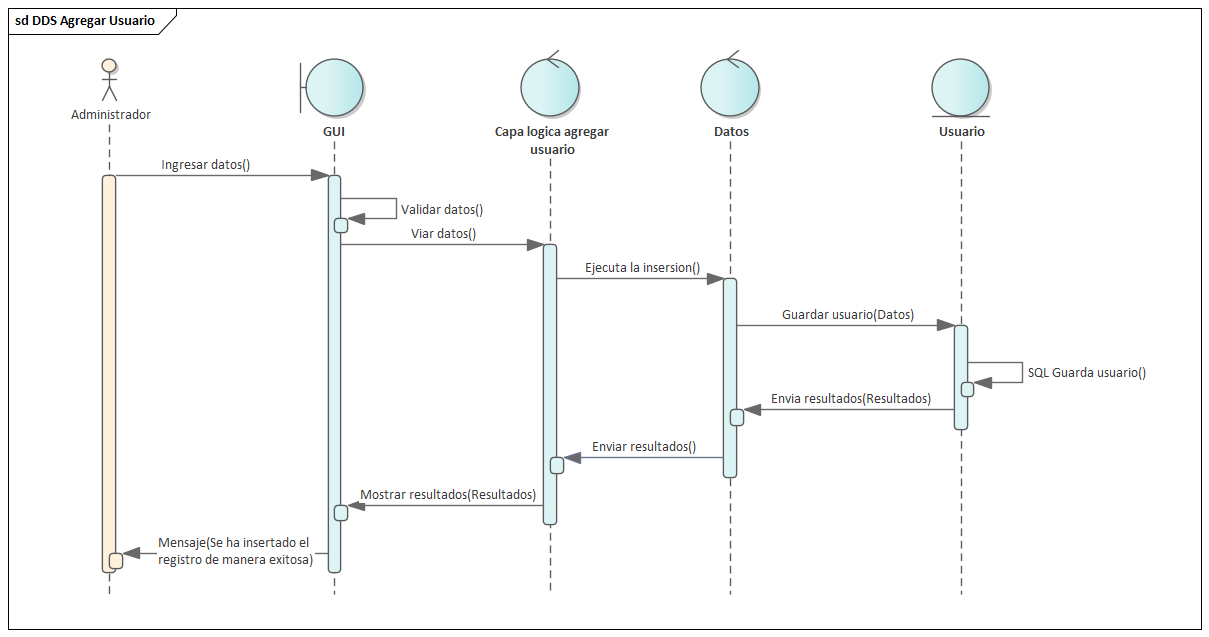
**Estado Usuario.**

****

Diagramas Diagrama Estado usuario

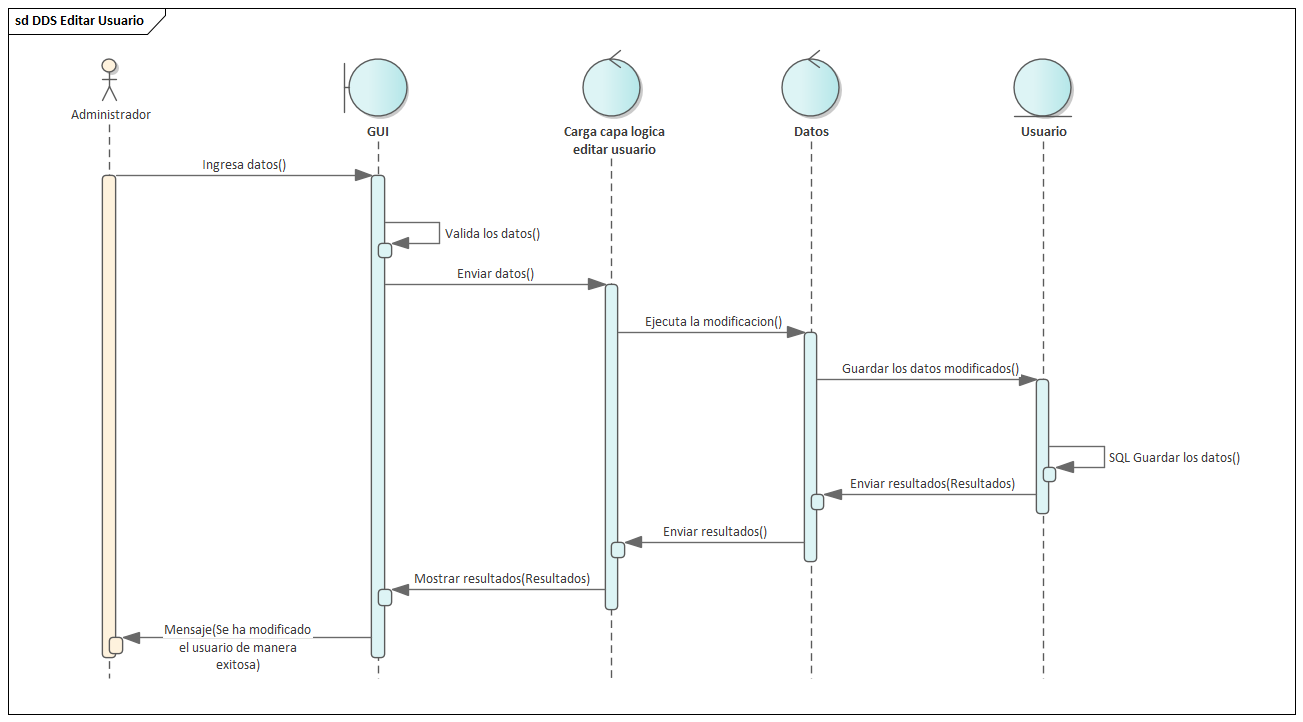
**Diagrama de Secuencias.**

**Agregar Usuario.**

****

Diagramas Diagrama de secuencia agregar usuario

**Editar Usuario.**

****

Diagramas Diagrama de secuencia editar usuario

**Imágenes del requerimiento mencionado.**

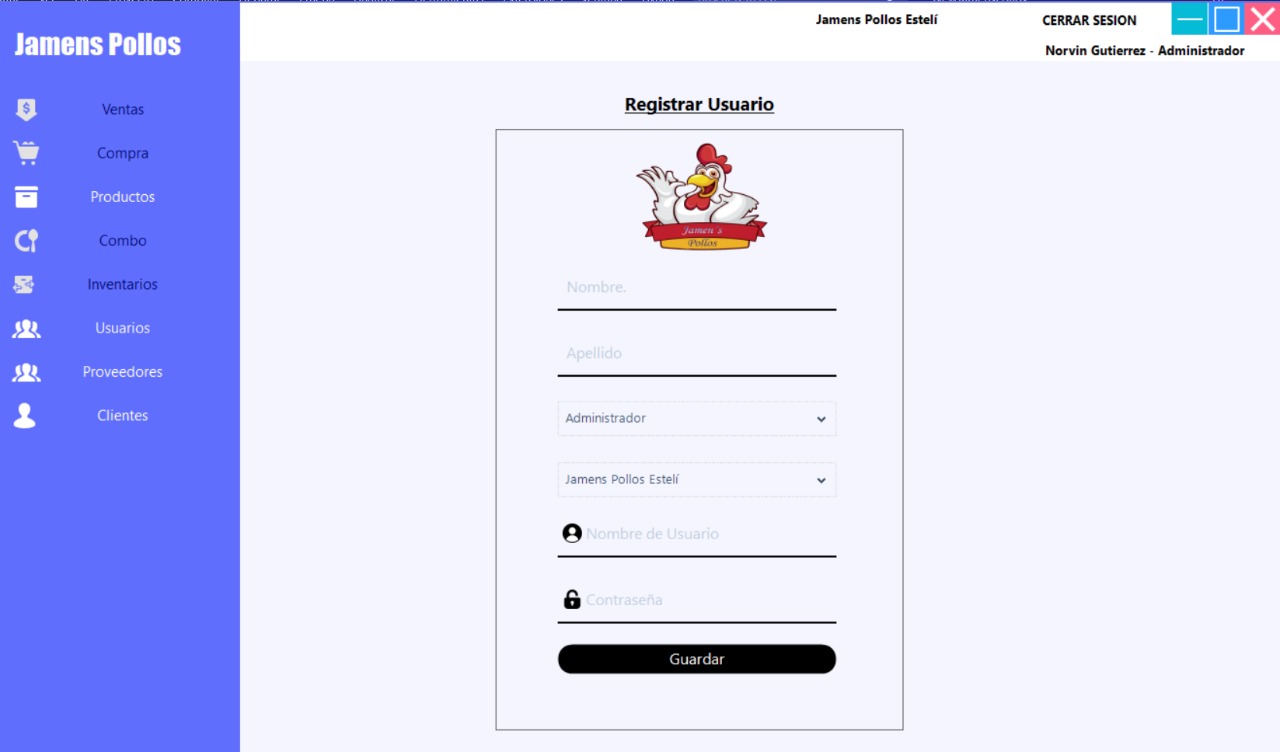
****

Ilustración Registro de usuario

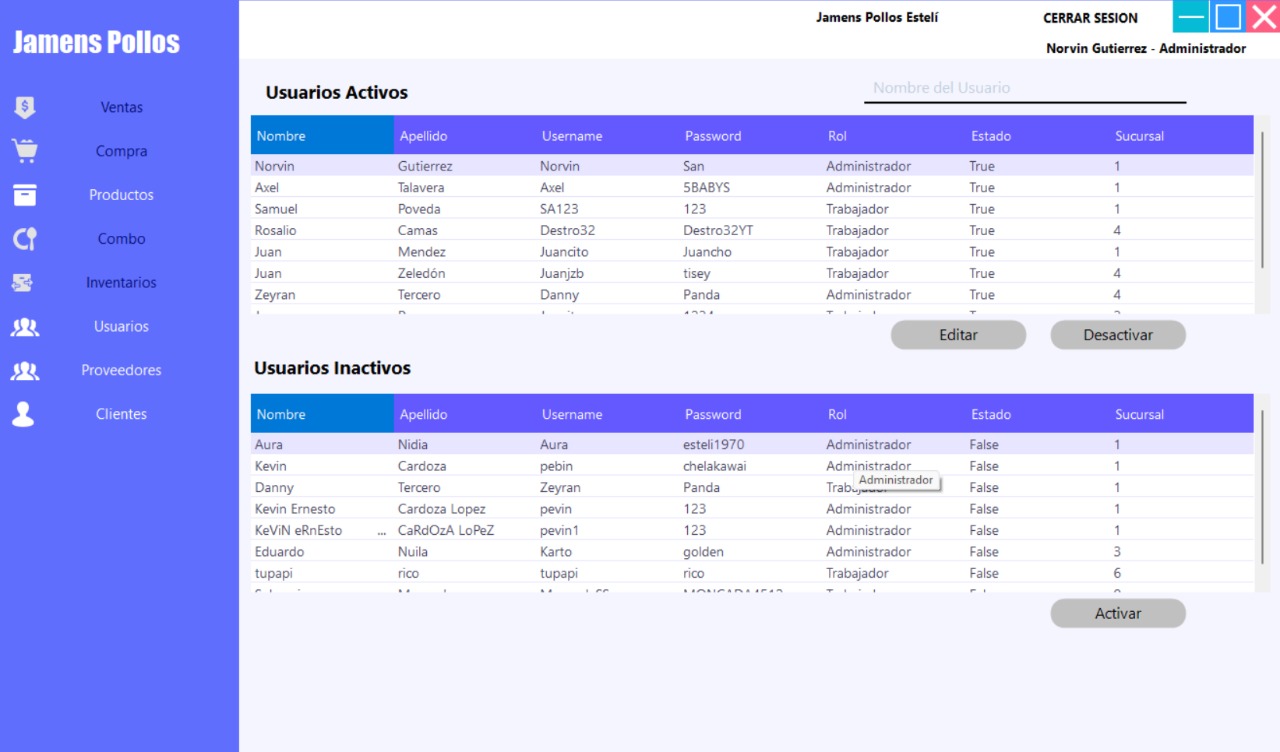
****

Ilustración usuarios registrados

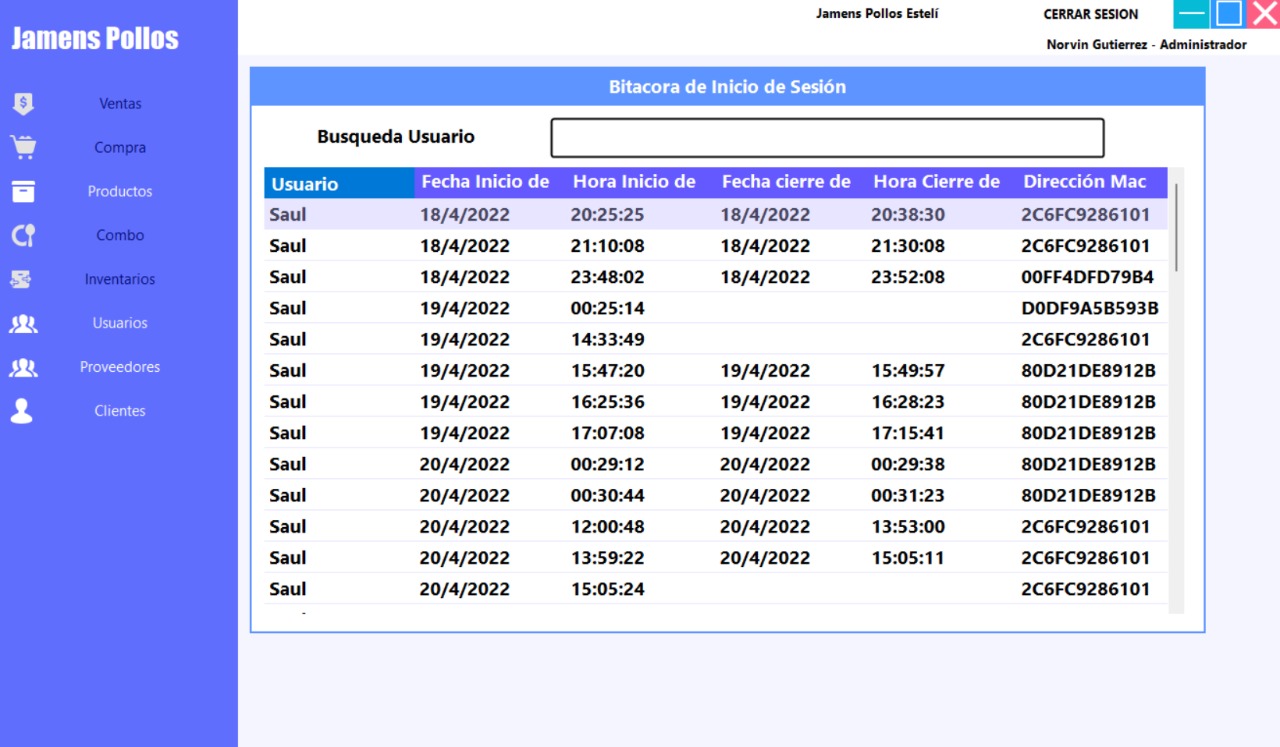
****

Ilustración Bitácora de inicio de sesión

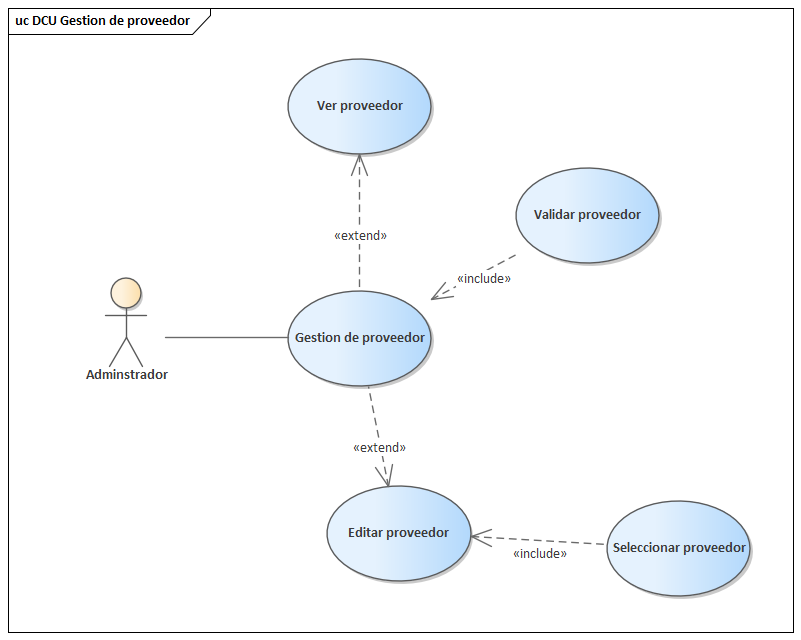
**Incremento #2: [Gestión de proveedor]**

El incremento de gestión de proveedores le da permiso al administrador para que pueda agregar proveedores, de igual manera puede inhabilitarlos, le permite editar los datos de los usuarios para una actualización de los datos de sus proveedores.

**Pila del Sprint del Incremento.**

**Escenarios de Caso de Uso.**

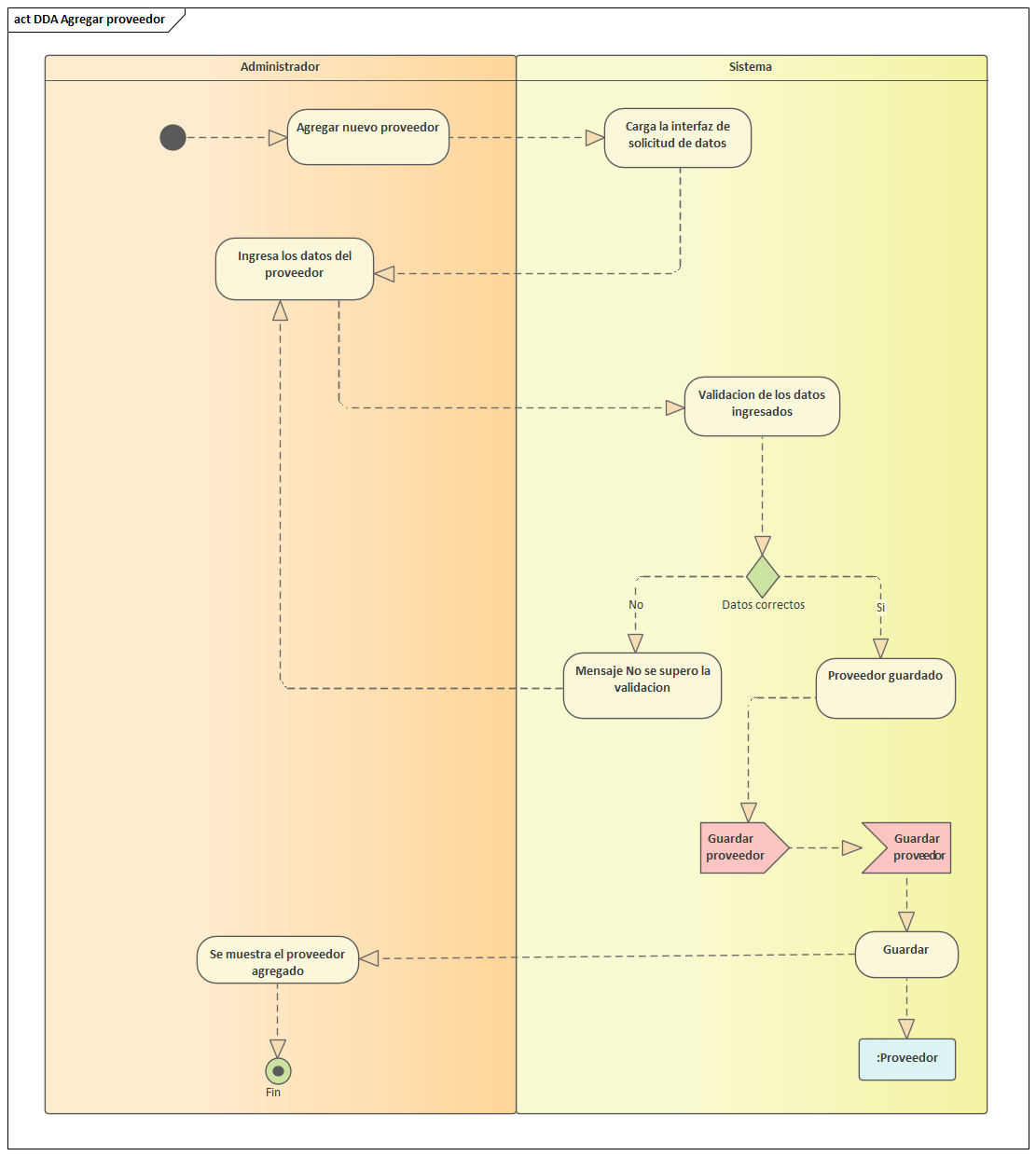
**Diagrama de Caso de Uso para gestión de Proveedores.**

****

Diagramas Diagrama caso de uso gestión de proveedor

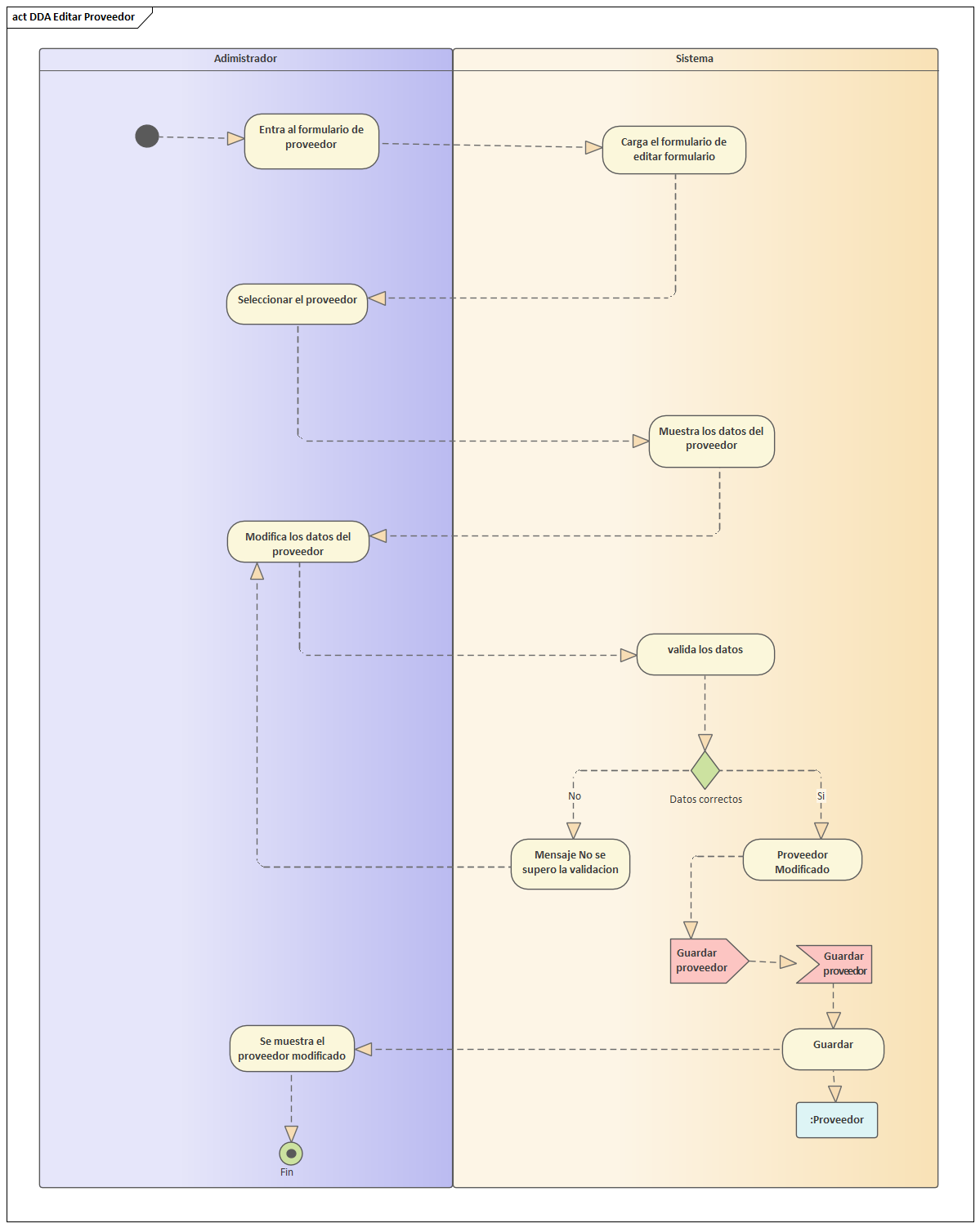
**Diagrama de Actividades.**

**Diagrama de Actividades para agregar proveedor.**

****

Diagramas Diagrama de actividad agregar proveedor

**Diagrama de Actividades editar proveedor.**

****

Diagramas Diagrama de actividad editar proveedor

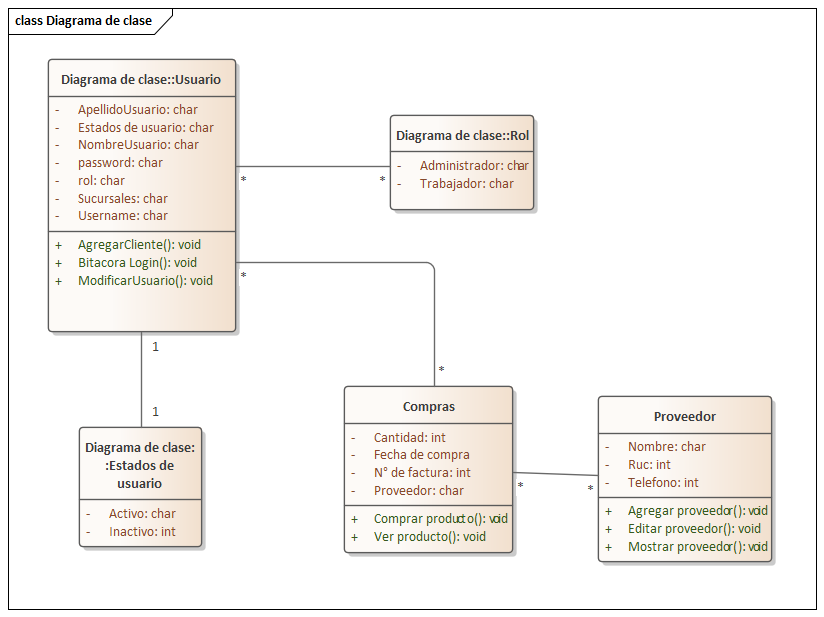
**Plantillas de casos de uso.**

|  |  |
| --- | --- |
| Caso de uso | **Escenario ver proveedores** |
| Actores | **Administrador del sistema** |
| Descripción | **Sirve para ver la lista de los proveedores** |
| Pre condición | **Se desea los proveedores** |
|  | 1. **El sistema ejecuta el caso de uso valida usuario: Administrador** 2. **El administrador ejecuta ver proveedor.** 3. **El sistema genera una ventana donde muestra el listado de proveedores** |
| Post condición | **Se mostró el listado de proveedores** |

|  |  |
| --- | --- |
| Caso de uso | Crear proveedor |
| Actores | Administrador del sistema |
| Descripción | Sirve para agregar un nuevo proveedor |
| Pre condición | Se desea agregar un nuevo proveedor |
|  | 1. El sistema ejecuta el caso de uso valida usuario: Administrador 2. El administrador ejecuta el comando propuesto 3. El administrador del sistema ingresa los datos solicitados 4. El administrador ejecuta el comando guardar. 5. Si todos los campos son rellenados continuar, si no volver al paso 4 6. El sistema ejecuta la función agregar proveedor en la base de datos. 7. El sistema muestra un mensaje proveedor guardado. |
| Post condición | proveedor finalizado |

|  |  |
| --- | --- |
| Caso de uso | Escenario editar proveedor |
| Actores | Administrador del sistema |
|  |  |
| Descripción | Sirve para editar los proveedores |
| Pre condición | Se desea editar los datos del proveedor |
|  | 1. Se ejecuta el caso de uso validar usuario: Administrador 2. El administrador ejecuta editar proveedor 3. El administrador ingresa los datos a modificar. 4. Si todos los campos están llenos continuar, si no volver a paso 3. 5. El sistema ejecuta la función editar proveedor en la base de datos. 6. El sistema muestra un mensaje proveedor guardado. |
| Post condición | Se mostró el listado de proveedores |

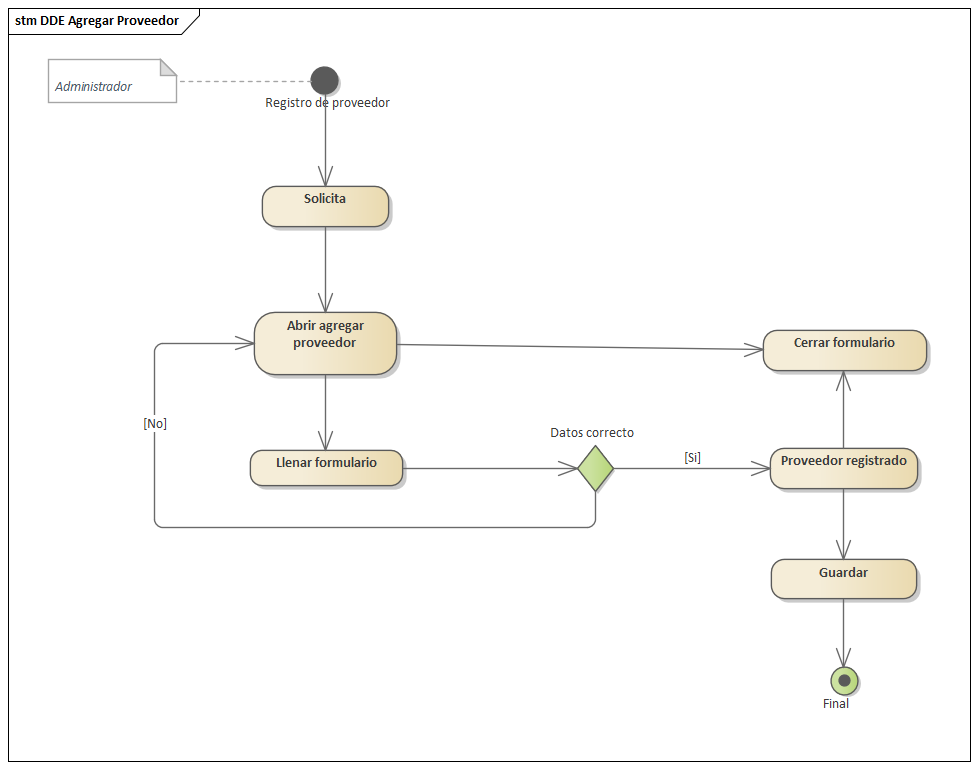
**Diagrama de Clases.**

****

Diagramas Diagrama de clase gestión de proveedor

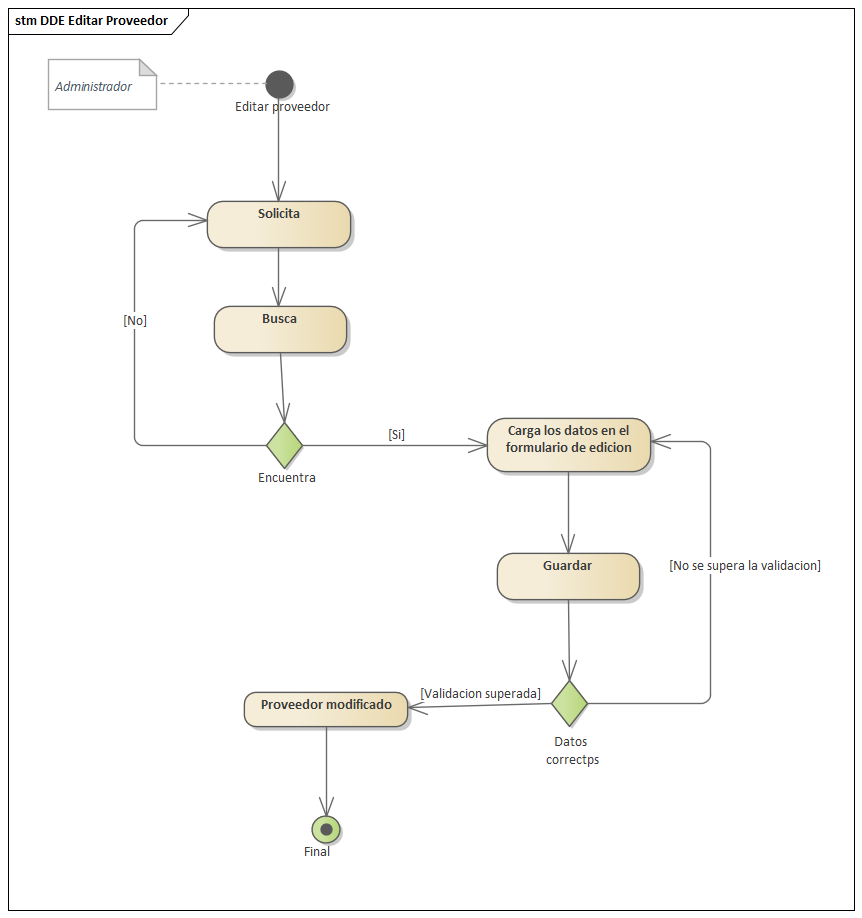
**Diagramas de Estado.**

**Diagrama de estado Agregar proveedor.**

****

Diagramas Diagrama de estado agregar proveedor

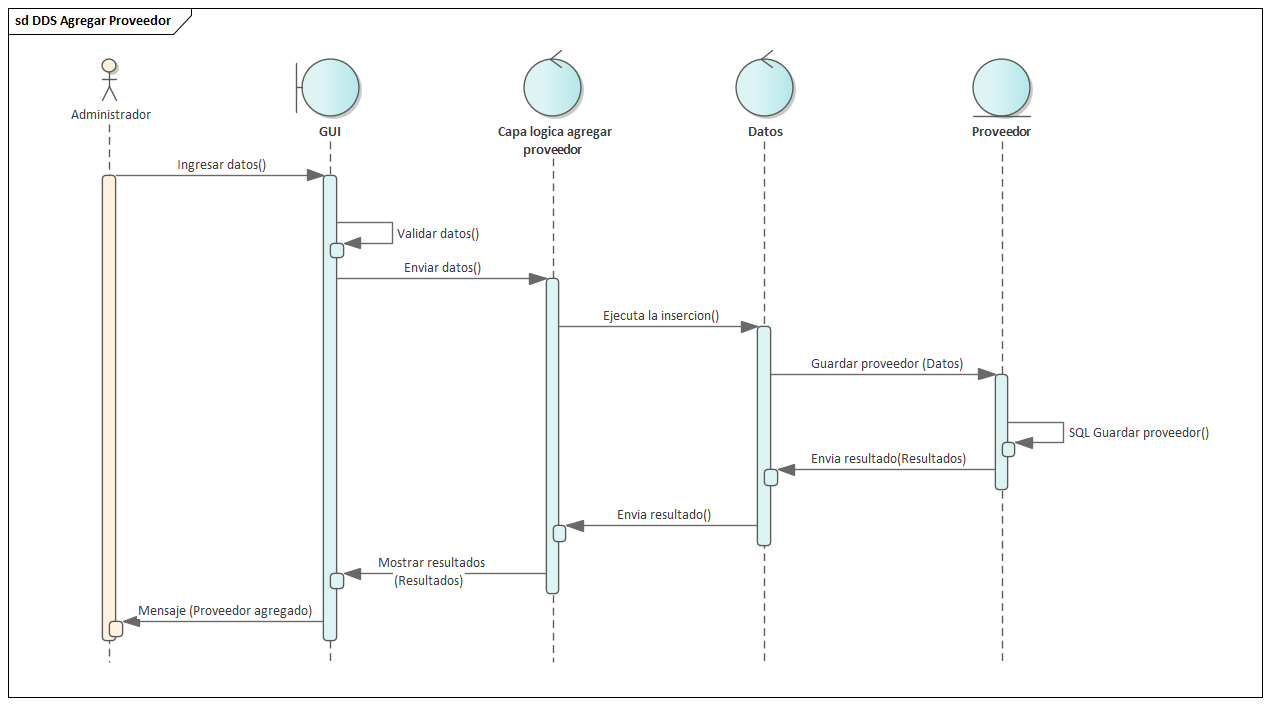
**Diagrama de estado Editar proveedor.**

****

Diagramas Diagrama de estado editar proveedor

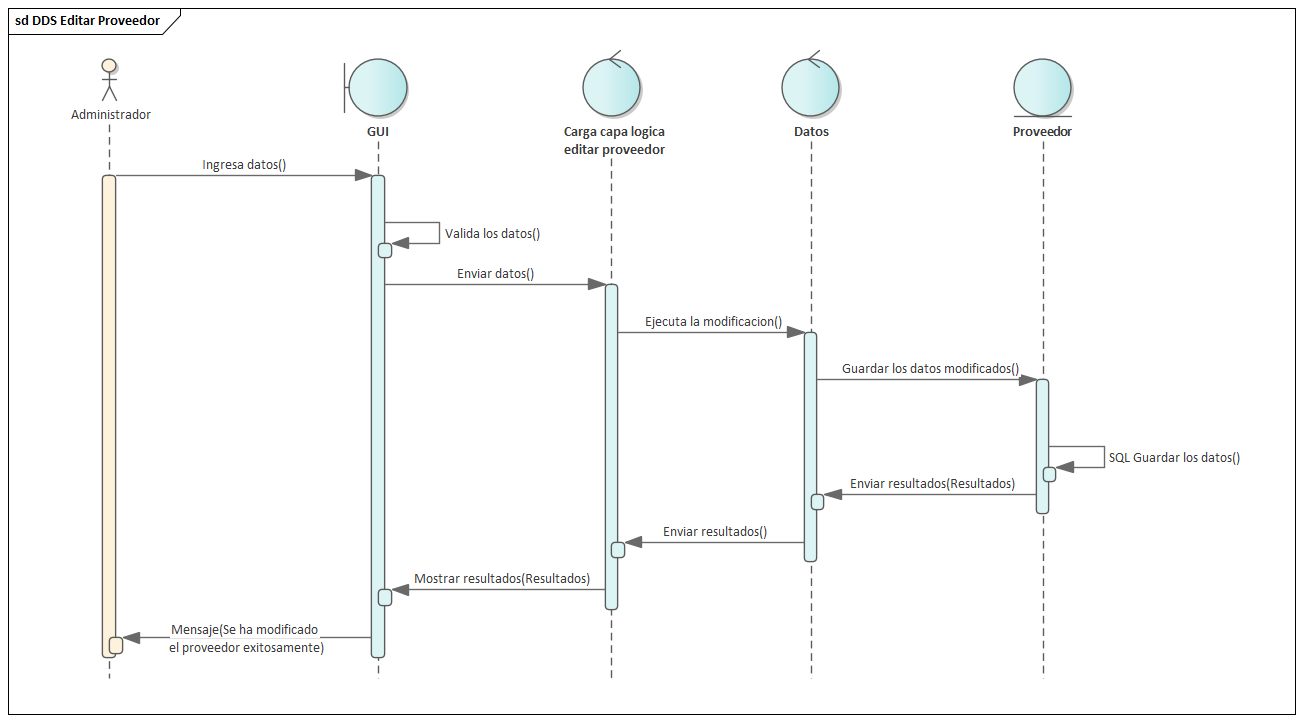
**Diagrama de Secuencia.**

**Agregar Proveedor.**

****

Diagramas Diagrama de sescuncia agregar proveedor

**Editar proveedor.**

****

Diagramas Diagrama de secuencia editar proveedor

**Diseño de interfaz.**

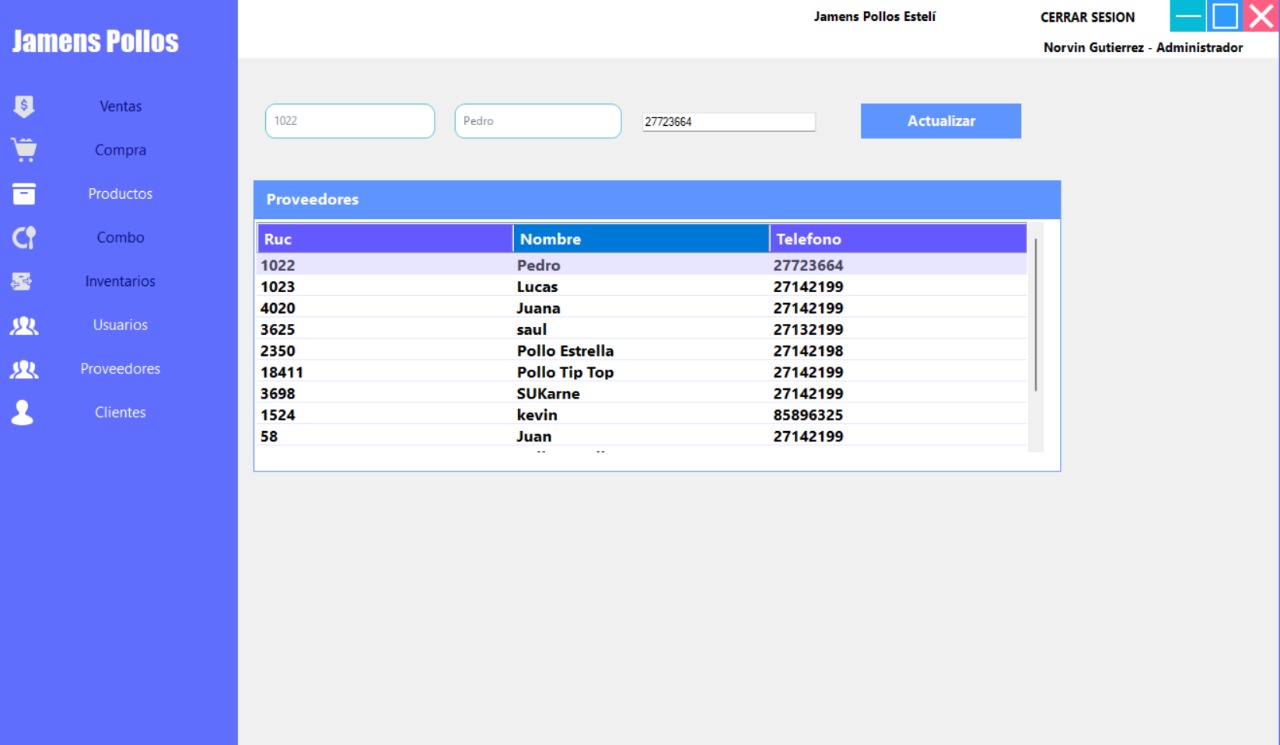
****

Ilustración Ver proveedores, Editar proveedores

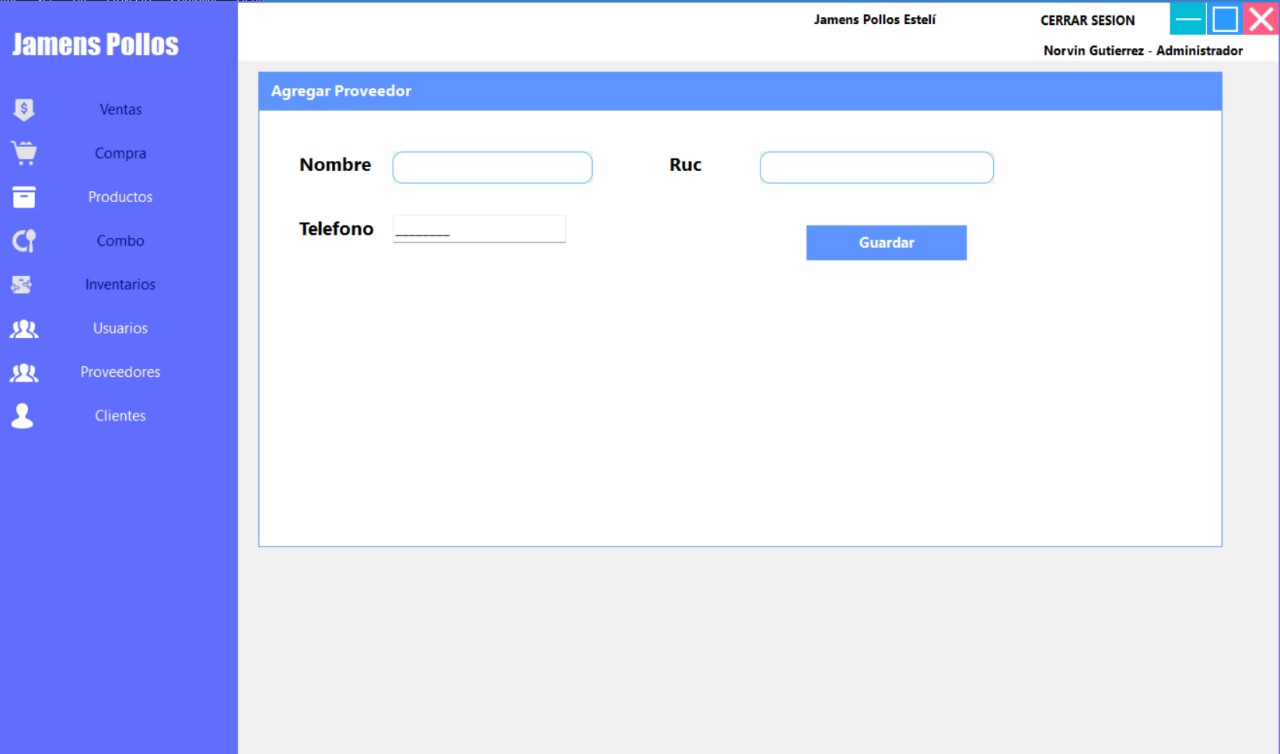
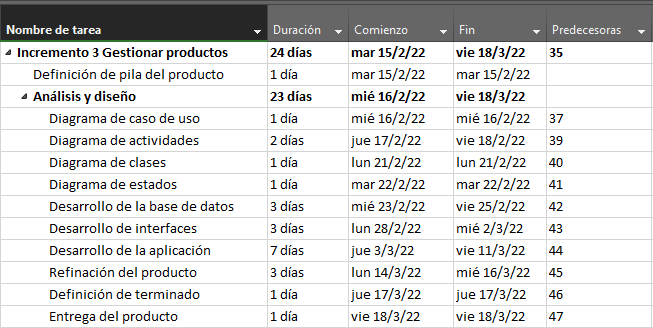
****

Ilustración Agregar proveedor

**Incremento #3: [Gestión de producto]**

El incremento de gestión de producto consiste en la manipulación del mismo es decir que al administrador el sistema debe permitirle crear e ingresar nuevos productos.

**Pila del Sprint del Incremento.**

**Escenarios de Caso de uso.**

**Diagrama de Caso de Usos de Gestionar Productos.**

**Diagrama de Actividades.**

**Editar Productos.**

**Crear Producto.**

**Plantillas de Caso de Uso.**

|  |  |
| --- | --- |
| Caso de uso | Escenario ver producto |
| Actores | Administrador del sistema |
| Descripción | Sirve para ver los productos registrado en el sistema |
| Pre condición | Se desea ver los productos registrados |
|  | 1. El sistema ejecuta el caso de uso valida usuario: Administrador 2. El administrador ejecuta ver producto. 3. El sistema genera una ventana donde muestra el listado de producto |
| Post condición | Se mostró el listado de producto |

|  |  |
| --- | --- |
| Caso de uso | Crear producto |
| Actores | Administrador del sistema |
| Descripción | Sirve para agregar un nuevo producto |
| Pre condición | Se desea agregar un nuevo producto |
|  | 1. El sistema ejecuta el caso de uso valida usuario: Administrador 2. El administrador ejecuta el comando propuesto 3. El administrador del sistema ingresa los datos solicitados 4. El administrador ejecuta el comando guardar. 5. Si todos los campos son rellenados continuar, si no volver al paso 4 6. El sistema ejecuta la función agregar producto en la base de datos. 7. El sistema muestra un mensaje producto agregado. |
| Post condición | Producto finalizado |

|  |  |
| --- | --- |
| Caso de uso | Escenario editar producto |
| Actores | Administrador del sistema |
| Descripción | Sirve para editar los datos del producto |
| Pre condición | Se desea editar los datos de un producto |
|  | 1. Se ejecuta el caso de uso validar usuario: Administrador 2. El administrador ejecuta editar producto 3. El administrador ingresa los datos a modificar. 4. Si todos los campos están llenos continuar, si no volver a paso 3. 5. El sistema ejecuta la función editar producto en la base de datos. 6. El sistema muestra un mensaje producto guardado. |
| Post condición | Se mostró el listado de producto |

Diagrama de Clases.

Diagrama de Secuencias.

Agregar Producto.

Modificar Producto.

Diagrama de Estados.

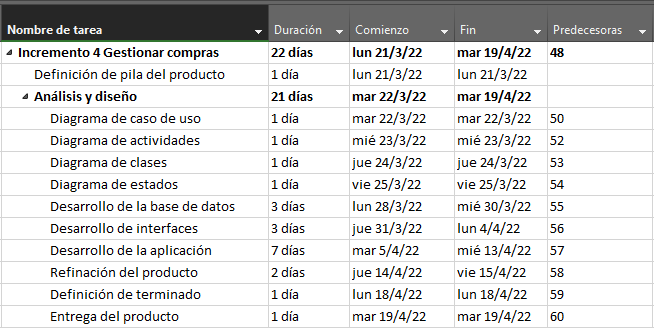
Crear Productos.

Editar Productos.

Diseño de Interfaz.

**Incremento #4: [Gestión de compra]**

El incremento de gestionar compra consiste en que el sistema debe permitirle al administrador hacer las diferentes tareas como lo es agregar compra, modificar las compras y ver las compras realizadas.

**Pila del Sprint del Incremento.**

Escenarios de Caso de Uso.

Diagrama de Actividades.

Agregar Compra.

**Plantilla de Caso de Usos.**

|  |  |
| --- | --- |
| Caso de uso | Escenario ver compra |
| Actores | Administrador del sistema |
| Descripción | Sirve para ver las compras realizadas |
| Pre condición | Se desean ver las compras |
|  | 1. El sistema ejecuta el caso de uso valida usuario: Administrador 2. El administrador ejecuta ver compra. 3. El sistema genera una ventana donde muestra el listado de compras |
| Post condición | Se mostró el listado de compra |

|  |  |
| --- | --- |
| Caso de uso | Escenario agregar compra |
| Actores | Administrador del sistema |
| Descripción | Sirve para agregar nueva compra |
| Pre condición | Se desea agregar una compra |
|  | 1. El sistema ejecuta el caso de uso valida usuario: Administrador 2. El administrador ejecuta el comando propuesto 3. El sistema genera una ventana donde solicita los datos del producto 4. El administrador ejecuta el comando guardar 5. Si todos los campos son correctos guardar, si no volver a paso 4 6. El sistema ejecuta la función agregar compra en la base de datos. 7. El sistema muestra un mensaje compra agregada |
| Post condición | Se mostró el listado de las compras |

|  |  |
| --- | --- |
| Caso de uso | Escenario editar compra |
| Actores | Administrador del sistema |
| Descripción | Sirve para editar las compras |
| Pre condición | Se desea editar los datos de las compras |
|  | 1. Se ejecuta el caso de uso validar usuario: Administrador 2. El administrador ejecuta editar compra. 3. El administrador ingresa los datos a modificar. 4. Si todos los campos están llenos continuar, si no volver a paso 3. 5. El sistema ejecuta la función editar compra en la base de datos. 6. El sistema muestra un mensaje compra guardado. |
| Post condición | Se mostró el listado de compra |

Diagrama de Clase.

Diagrama de Secuencia.

Agregar Compra.

Diagrama de Estado.

Agregar Compra.

Diseño de Interfaz.

**Incremento #5: [Manejo de Combos]**

El incremento manejo de combos consiste en la manipulación de los mismos, esto quiere decir que el sistema debe de permitirle al administrador ingresar, actualizar y desechar combos, mostrar la cantidad en existencia de cada combo ingresado en el sistema y poder permitir al vendedor o admón. ingresar la preparación de combos en el día actual.

**Pila del Sprint del Incremento.**

Escenario de Caso de Uso.

Diagrama de Actividades.

Crear Combo.

Editar Combo.

Preparar Orden.

**Plantilla de Caso de Usos.**

|  |  |
| --- | --- |
| **Caso de uso** | Escenario Agregar Combo |
| **Actores** | Vendedor |
| **Descripción** | El sistema agregará el combo, así como los ingredientes necesarios para elaborarlo |
| **Precondición** | Existir Ingredientes |
| **Pasos** | 1. El usuario ingresa al apartado agregar combo. 2. El sistema carga los campos a rellenar. 3. Una vez ingresados y mandados a guardar el sistema verifica que no exista ese nombre de combo. 4. El sistema mostrará un mensaje de éxito y vaciará los campos |
| **Post-Condición** | Si los datos pasaron los filtros estará disponible el combo para preparar, sino se manda a rellenar de nuevo los campos. |

|  |  |
| --- | --- |
| **Caso de uso** | Escenario Editar Combo |
| **Actores** | Vendedor |
| **Descripción** | El sistema editará combos existentes, así como los ingredientes necesarios para elaborarlo |
| **Precondición** | Existir Ingredientes y combos |
| **Pasos** | 1. El usuario ingresa al apartado editar combo. 2. El sistema carga los combos disponibles y el usuario escoge cual editar. 3. Una vez editados y mandados a guardar los cambios, el sistema verifica que no exista ese nombre de combo. 4. El sistema mostrará un mensaje de éxito y vaciará los campos |
| **Post-Condición** | Si los datos pasaron los filtros estarán los datos del combo actualizado, caso contario se manda a rellenar de nuevo |

|  |  |
| --- | --- |
| **Caso de uso** | Escenario Preparar Combo |
| **Actores** | Vendedor |
| **Descripción** | El sistema preparará los combos y sustraerá de los inventarios los ingredientes necesarios |
| **Precondición** | Existir Ingredientes en existencia |
| **Pasos** | 1. El usuario ingresa al apartado preparar combos. 2. El sistema carga los combos disponibles y el usuario escoge cual meter a la orden. 3. Una vez elegidos el sistema guarda la orden y resta los ingredientes que se necesitan para vender la orden. 4. El sistema mostrará un mensaje de éxito y vaciará los campos |
| **Post-Condición** | No se pueden preparar combos con productos con existencia 0 o preparar combos que no tengan ingredientes suficientes. |

Diagrama de clase.

Diagrama de Estados.

Crear Combos.

Editar Combos.

Preparar Combos.

Diagrama de Secuencias.

Agregar Combos.

Editar Combos.

Preparar Orden.

Diseño de Interfaces.

**Incremento #6: [Gestión de venta]**

El incremento Gestión de venta consiste en añadir ventas de combos y productos extras, además de poder editarse estás ventas. Indirectamente está función afecta al inventario de combos y productos extras, restando la cantidad comprada por la existente.

**Pila del Sprint del Incremento.**



Escenarios de Caso de Uso.

Diagrama de Actividades.

Agregar Ventas.

**Plantilla de Casos de Uso.**

|  |  |
| --- | --- |
| **Caso de uso** | Escenario Agregar Venta |
| **Actores** | Empleado |
| **Descripción** | El sistema registrará la venta de los productos extras y de los combos |
| **Precondición** | Tener la existencia de los productos en inventario |
| **Pasos** | 1. El usuario ingresa al apartado agregar venta. 2. El sistema carga los combos y productos posteriormente el usuario escoge cual meter a la factura. 3. Una vez elegidos el sistema guarda la venta y resta del inventario correspondiente la cantidad comprada. 4. El sistema mostrará un mensaje de éxito y vaciará los campos |
| **Post-Condición** | No se pueden vender productos con existencia 0 o vender una cantidad superior a la existente. |

|  |  |
| --- | --- |
| **Caso de uso** | Escenario Editar Venta |
| **Actores** | Empleado |
| **Descripción** | El sistema editará las ventas realizadas |
| **Precondición** | La venta no debe tener un plazo de 10 días de haber sido efectuada |
| **Pasos** | 1. El usuario ingresa al apartado editar venta. 2. El sistema carga las ventas posteriormente el usuario escoge cual editar y efectúa las modificaciones. 3. Una vez modificado el sistema guarda la edición y resta del inventario o suma correspondiente la cantidad editada. 4. El sistema mostrará un mensaje de éxito y vaciará los campos |
| **Post-Condición** | No se puede meter cantidades que sobrepasen el inventario actual |

Diagrama de Clases.

Diagrama de Estados.

Agregar Ventas.

Ver ventas.

Diagrama de Secuencias.

Agregar Ventas.

Ver ventas.

Diseño de Interfaces.

Diagrama de Componentes.

Diagrama Modelo Relacional.

Interfaces Finales del Sistema.

**Conclusiones.**

Para concluir con esta investigación, la cual va orientada al negocio de ventas de alimentos llamado Jamen’s Pollos se llega al siguiente resultado:

Se realizó un estudio de factibilidad para la correspondiente implementación de sistema hasta obtener que este proyecto es factible por las siguientes razones:

* Desde lo técnico, únicamente se ha de adquirir un equipo en caso de que la administración desee expandir el negocio, lo que a su vez conlleva a contratación de personal, pero de momento esto solo se mantiene como una posible vista al futuro.
* En lo que respecta a lo operativo, este sistema no tendrá de ninguna manera un impacto negativo dentro del negocio ni retrasará algún proceso de función en este, más bien vendrá a mejorar toda la funcionalidad de Jamen’s pollos.
* Por otro lado, tomando en cuento los aspectos económicos y financieros, pretendemos que los resultados se noten de manera directa y este agilice todo proceso en el negocio, lo cual vendrá a disminuir horas y esfuerzo de trabajo y ayudará a que crezca el prestigio entre los demás negocios.
* Legalmente, la implementación de este sistema no incumple con ninguna ley en nuestro país referentes a ventas y manejo de información de datos.

Se logró la identificación de manera exitosa de los requerimientos funcionales y no funcionales, por lo que definimos los requerimientos a través de medios de recopilación de información a como lo es una entrevista.

De igual modo se diseñó el sistema de ventas local y así mismo se desarrolló para su respectivo uso ya en el negocio, aplicamos el diseño de Lenguaje Unificado de Modelado UML donde obtuvimos unos buenos resultados y a su vez implementamos la metodología Scrum.

# **Recomendaciones.**

* Es necesario que la empresa inicie un periodo de prueba del sistema, para realizar una validación del correcto funcionamiento del mismo.
* Se debe realizar capacitación para los usuarios que utilizaran el sistema, con el objetivo de que estos (los usuarios) adquieran las habilidades para el manejo del software.
* Asignar a un administrador principal que permita la gestión de los usuarios y configuraciones que se requieran en el sistema.
* Realizar reuniones de manera periódica con el fin de realizar mantenimiento al sistema, para reparar fallos y/o agregar nuevas funcionalidades.
* En el aspecto técnico se debe proceder a la adquisición de los equipos informáticos que se propusieron para poder garantizar el funcionamiento adecuado del entorno de producción.
* De igual manera, al momento de hacer una expansión, es necesario realizar un estudio técnico para la adquisición de nuevas tecnologías que permitan cubrir las especificaciones técnicas para la implementación del sistema en un entorno de red más amplio.

# Bibliografía

*A.L.EPH*. (28 de Marzo de 2021). Obtenido de ¿Qué es la metodología RUP?: https://aleph.org.mx/que-es-la-metodologia-rup

Acevedo, W. (18 de Julio de 2008). *wordpress*. Obtenido de modelo cocomo: https://acevedodelacru.wordpress.com/2008/07/18/%C2%BFque-es-el-cocomo/

*besoftware*. (s.f.). Obtenido de ¿Qué es C# y para qué sirve?: https://bsw.es/que-es-c/

*ConceptoABC*. (s.f.). Obtenido de Software de sistema: https://conceptoabc.com/software-de-sistema/

*es.what-this.com*. (16 de Mayo de 2020). Obtenido de Especificación de requisitos de software: https://amp.es.what-this.com/1155021/1/especificacion-de-requisitos-de-software.html

*Interpolados*. (28 de Julio de 2018). Obtenido de ANÁLISIS DE REQUERIMIENTOS: https://interpolados.wordpress.com/2018/07/28/analisis-de-requerimientos/

*Lucidchart*. (s.f.). Obtenido de Qué es el lenguaje unificado de modelado (UML): https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml

Marín, R. (16 de Abril de 2019). *Revista Digital*. Obtenido de Los gestores de base de datos más usados en la actualidad: https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/

Mesa, A. R. (19 de Diciembre de 2018). *Blog de programación y sistemas*. Obtenido de Qué es un Sprint de Scrum: https://openwebinars.net/blog/que-es-un-sprint-scrum/

*Monografias*. (s.f.). Obtenido de Viabilidad operacional: https://www.monografias.com/docs/Viabilidad-operacional-F3ZABJGPCDGNZ

*Negocios y estrategias*. (s.f.). Obtenido de C#. Qué es y para qué se utiliza: https://negociosyestrategia.com/blog/que-es-csharp/

Nicuesa, M. (Mayo de 2015). *Definición ABC*. Obtenido de Viabilidad Legal: https://www.definicionabc.com/derecho/viabilidad-legal.php

Peiró, R. (s.f.). *Sistema de información*. Obtenido de Economipedia: https://economipedia.com/definiciones/sistema-de-informacion.html

Polo, C. A. (s.f.). *Sistema de ventas*. Obtenido de monografias.com: https://www.monografias.com/trabajos109/monografia-sistema-ventas/monografia-sistema-ventas.shtml#sistemadea

*Rambla Informática*. (s.f.). Obtenido de Enterprise Architect: https://www.ramblainf.com/enterprise-architect

Robledano, Á. (24 de Septiembre de 2019). *Open Webinars*. Obtenido de Qué es MySQL: Características y ventajas: https://openwebinars.net/blog/que-es-mysql/

Romero, G. (s.f.). *Espacios media*. Obtenido de ¿Qué es un Análisis de Requerimientos?: https://www.espacios.media/que-es-un-analisis-de-requerimientos/

S., J. (12 de Septiembre de 2021). *Economía 3*. Obtenido de ¿Qué es la viabilidad de un proyecto y cómo aplicarlo?: https://economia3.com/que-es-la-viabilidad-de-un-proyecto-como-aplicarlo/

Schiaffarino, A. (12 de Marzo de 2019). *infranetworking*. Obtenido de Modelo cliente servidor: https://blog.infranetworking.com/modelo-cliente-servidor/

Silverio, Y. A. (s.f.). *Monografias.com*. Obtenido de Modelos del proceso del software: https://www.monografias.com/trabajos108/modelos-del-proceso-del-software/modelos-del-proceso-del-software.shtml#elprocesoa

*Sistema Venta*. (10 de Julio de 2020). Obtenido de Qué es un Sistema de Ventas: https://sistema-ventas.com.mx/sistemas-de-ventas/que-es-un-sistema-de-ventas/

*Sistemas Ventas*. (10 de Julio de 2020). Obtenido de Qué es un Sistema de Ventas: https://sistema-ventas.com.mx/sistemas-de-ventas/que-es-un-sistema-de-ventas/#3-%C2%BFqu%C3%A9-es-un-sistema-de-ventas-local

*Sites google*. (s.f.). Obtenido de Estimaciones de costos: https://sites.google.com/site/gestiondeproyectossoftware/unidad-3-planificacion-de-proyecto/3-3-estimaciones-de-costos

*SOFTENG*. (s.f.). Obtenido de Metodología Scrum para desarrollo de software - aplicaciones complejas: https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html

*UGR Emprendedora*. (1 de Junio de 2017). Obtenido de Viabilidad de proyectos: https://ugremprendedora.ugr.es/viabilidad-de-proyectos/

*Ungoti*. (s.f.). Obtenido de Ciclo de vida del desarrollo de software: https://ungoti.com/es/soluciones/desarrollo-de-software/sdlc/

Anexos.

Entrevistas.

**Entrevista para la gerencia de Jamen’s Pollos.**

Objetivo: La presente entrevista tiene como objetivo la recopilación de información para así conocer los requerimientos necesarios que deben tener en cuenta para la creación de un sistema de ventas que se pondrá en uso en la sucursal Jamen's Pollos, ubicado en el departamento de Estelí, Nicaragua.

1. **¿En la actualidad existe algún problema que afecte su negocio?**

|  |
| --- |
|  |
|  |

Si

No

1. **¿Cuál es la problemática que tiene?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Cuál es la razón por la que se desea dar solución a este problema?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Cómo resolvería usted dicha problemática?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Quiénes son las personas que se encargan de ejecutar las actividades relacionadas a la problemática?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿De acuerdo a la pregunta N°5 qué funciones desempeñan estas personas?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Actualmente el negocio cuenta con un sistema.**

|  |
| --- |
|  |
|  |

Si

No

1. **¿Cree usted necesario un sistema o la implementación de un software que resuelva las problemáticas actuales del negocio?**

|  |
| --- |
|  |
|  |

Si

No

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Estaría dispuesto a apoyar el desarrollo y la implementación de este sistema para así considerar la solución a la problemática que su negocio presenta?**

|  |
| --- |
|  |
|  |

Si

No

1. **¿Qué funciones considera usted que debería tener el sistema para así sea útil y ayude a solucionar considerablemente su problema?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Dispone usted con el equipo necesario para la instalación de este sistema?**

|  |
| --- |
|  |
|  |

Si

No

1. **De acuerdo a la pregunta 11 ¿Estaría dispuesto a comprar el equipo necesario para la implementación del software?**

|  |
| --- |
|  |
|  |

Si

No

1. **¿En qué ambiente desea que funcione el sistema para su negocio?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Qué personas serían las encargadas de utilizar el sistema?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿El personal que labora en su negocio, tiene experiencia en el uso de sistemas?**

|  |
| --- |
|  |
|  |

Si

No

1. **En caso de que la pregunta 14 sea No. ¿Estaría dispuesto a capacitar a su personal para el uso de este sistema?**

|  |
| --- |
|  |
|  |

Si

No

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Contribuiría el sistema a alcanzar los objetivos de la empresa**

|  |
| --- |
|  |
|  |

Si

No  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **. ¿Considera necesario que el sistema lleve un registro de los clientes que llegan a comprar al local?**

|  |
| --- |
|  |
|  |

Si

No

1. **¿Para usted el sistema debería de tener un respaldo de la información?**

|  |
| --- |
|  |
|  |

Si

No

1. **En caso de que la pregunta 19 su respuesta sea sí ¿Con qué frecuencia se debería hacer los respaldos?**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **¿Qué tan vitales serán los siguientes factores en el Sistema de Información a**

**desarrollar? Tome en cuenta los siguientes valores y marque con una X según**

**corresponda**:

1: Innecesario 3: Vital

2: Opcional 4: Obligatorio

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Característica | 1 | 2 | 3 | 4 |
| Atractivo diseño de interfaces amigables con el usuario |  |  |  |  |
| Funcionamiento eficiente del sistema |  |  |  |  |
| Seguridad de la información almacenada |  |  |  |  |
| Confiabilidad en los resultados |  |  |  |  |
| Disponibilidad inmediata del sistema |  |  |  |  |
| Abierto a mejoras en el futuro |  |  |  |  |

1. **¿Hay alguna mejora que quisiera implementar en el futuro?**

|  |
| --- |
|  |
|  |

Si

No  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fichas ocupaciones o Manual de Funciones.

|  |  |  |
| --- | --- | --- |
| **Manual de Funciones** | **Documento No. 04-17-20-00-00** | |
| **Emisión.**  **15/04/2022** | **Generado por:**  División de Recursos Humanos | **Aprobado por:**  Rectoría. |
| **Descripción del Cargo De Trabajo.** | | |
| **Dependencia Organizativa.** | | Administración. |
| **Nombre del Cargo.** | | Administrador Jamen´s Pollos. |
| **Cargo Superior Inmediato.** | | - |
| **Cargo Subordinado.** | | Atención al Cliente.1 |

**Propósitos del Cargo:** Elabora, resguarda, controla todos los documentos administrativos del negocio, de igual manera coordina las actividades del día a día de los demás trabajadores.

|  |  |
| --- | --- |
| **Funciones.** | |
| 1 | Es el responsable de llevar a cabo todas aquellas actividades necesarias para alcanzar los objetivos planteados. |
| 2 | Planificar, organizar y dirigir el control de una sociedad, asumiendo el cargo de una parte muy importante del negocio. |
| 3 | Conocer diversos aspectos del negocio, tanto de finanzas como de innovación. |
| 4 | Representar la empresa frente al resto de sociedades, ya que, hay que seleccionar a una persona física para que asuma el control de todas las relaciones de dicha organización. |
| 5 | Es responsable del cumplimiento de la Ley, tanto de los estatutos como de la actividad general. |
| 6 | Deberá cumplir las siguientes obligaciones legales, como son, las laborales, las penales, las civiles, las mercantiles y las tributarias. |
| 7 | Monitorear las actividades y tareas que se están realizando y corregir los errores que se puedan presentar. Permite obtener información precisa de lo que sucede dentro del proceso. |
| 8 | Verificar que los datos resultantes de todos los días coincidan con lo recibido por parte de los trabajadores. |
| 9 | Brindar un ambiente de confianza y armonía laboral para que así todos trabajen emotivamente y den lo mejor de sí. |

|  |  |  |
| --- | --- | --- |
| **Manual de Funciones** | **Documento No. 04-17-20-00-00** | |
| **Emisión.**  **15/04/2022** | **Generado por:**  División de Recursos Humanos | **Aprobado por:**  Rectoría. |
| **Descripción del Cargo De Trabajo.** | | |
| **Dependencia Organizativa.** | | Jamen´s Pollos. |
| **Nombre del Cargo.** | | Atención al Cliente  Jamen´s Pollos. |
| **Cargo Superior Inmediato.** | | Administración. |
| **Cargo Subordinado.** | | Operador. |

**Propósitos del Cargo:** Es el encargado de atender con amabilidad a cada cliente que llegue al negocio, de igual manera de tomar las órdenes de comida correspondientes emitidas de los clientes.

|  |  |
| --- | --- |
| **Funciones.** | |
| 1 | Recepción del cliente y gestión de la expectativa de la reserva. |
| 2 | Satisfacer las necesidades de los clientes es una responsabilidad básica del personal de servicio. |
| 3 | Ser siempre educado con los clientes, dar la bienvenida por cada cliente que llegue al negocio. |
| 4 | Tener conocimiento previo de todo lo que se oferta en el negocio, para que de tal manera puedan brindan mayor seguridad al momento de elegir el producto a consumir. |
| 5 | Traslado de los pedidos correspondientes siempre de manera segura, para evitar quejas por parte de los clientes. |
| 6 | Brindar la respectiva cuenta a los clientes, tomar el dinero, procesar el pago y registrarlo en el sistema del negocio. |

|  |  |  |
| --- | --- | --- |
| **Manual de Funciones** | **Documento No. 04-17-20-00-00** | |
| **Emisión.**  **15/04/2022** | **Generado por:**  División de Recursos Humanos | **Aprobado por:**  Rectoría. |
| **Descripción del Cargo De Trabajo.** | | |
| **Dependencia Organizativa.** | | Jamen´s Pollos. |
| **Nombre del Cargo.** | | Operadora  Jamen´s Pollos. |
| **Cargo Superior Inmediato.** | | Atención al cliente. |
| **Cargo Subordinado.** | | - |

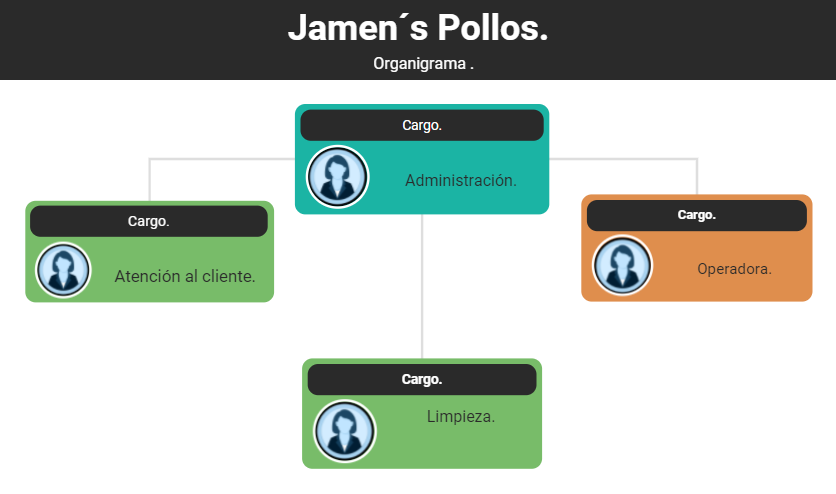
**Propósitos del Cargo:** Realizar todos los procesos de producción de los productos una vez son presentados por el encargado de atención al cliente.

|  |  |
| --- | --- |
| **Funciones.** | |
| 1 | Mantener aseadamente el área de producción. |
| 2 | Contar siempre con buena salud, en caso de que presente complicaciones deberá comunicarlo a administración. |
| 3 | Seguir correctamente las órdenes que se le presenten para su debida preparación. |
| 4 | Realizar las producciones correspondientes para la satisfacción de las necesidades de los clientes. |
| 5 | Dar el sabor al gusto según las especificaciones de los clientes según lo visto en el menú o bien expuesto por el encargado de atención al cliente. |
| 6 | Trabajar bajo presión y en tiempo delimitado por cada orden. |

|  |  |  |
| --- | --- | --- |
| **Manual de Funciones** | **Documento No. 04-17-20-00-00** | |
| **Emisión.**  **15/04/2022** | **Generado por:**  División de Recursos Humanos | **Aprobado por:**  Rectoría. |
| **Descripción del Cargo De Trabajo.** | | |
| **Dependencia Organizativa.** | | Jamen´s Pollos. |
| **Nombre del Cargo.** | | Limpieza  Jamen´s Pollos. |
| **Cargo Superior Inmediato.** | | Administración. |
| **Cargo Subordinado.** | | - |

**Propósitos del Cargo:** Mantener todas las áreas de la localidad de la empresa aseadas ante la llegada de los clientes, de igual manera utensilios a usar.

|  |  |
| --- | --- |
| **Funciones.** | |
| 1 | Mantener aseadamente el área de producción, atención al cliente, sala de estar y administración. |
| 2 | Ser una persona activa. |
| 3 | Tener constante asea en general. |
| 4 | Ser honesto y puntual. |
| 5 | Trabajar bajo presión. |

**Organigrama de la empresa.**

Como se puede apreciar en el organigrama de Jamen´s Pollos, su estructura se basa en:

* Un administrador que lleva el control de todo el negocio como tal.
* Una persona encargada en lo que respecta la atención al cliente.
* Una operadora que lleve a cabo todo el proceso de producción de los productos que se ofertan en el negocio.
* Una persona para la limpieza correspondiente de todo el negocio.