# UNIVERSIDAD PRIVADA “FRANZ TAMAYO”

Facultad de Ingeniería Carrera de Ingeniería de Sistemas



# PROYECTO DE GRADO

## SISTEMA DE INFORMACIÓN WEB DE COMERCIO ELECTRÓNICO PARA LA ADQUISICIÓN VÍA ONLINE DE PRODUCTOS TECNOLÓGICOS

**CASO:** GRUPO ARTEC BOLIVIA

**Postulante:** Carlos Juan Baina Limpias

**Tutor(a):** Ing. Hilaria Adima Vásquez Duran

El Alto-Bolivia

2017

## INDICE DE CONTENIDOS

1. [INTRODUCCIÓN 1](#_bookmark0)
   1. [PRESENTACIÓN DEL PROBLEMA DE INVESTIGACIÓN 2](#_bookmark1)
   2. [ANTECEDENTES 2](#_bookmark2)
      1. [ANTECEDENTES DE LA EMPRESA 2](#_bookmark3)

[5.2.1.1. MISION, VISION Y ORGANIGRAMA INSTITUCIONAL 3](#_bookmark4)

* + 1. [ANTECEDENTES DEL PROYECTO 4](#_bookmark6)

1. [DESARROLLO 7](#_bookmark7)

[CAPITULO 1. MARCO TEORICO 7](#_bookmark8)

* 1. [SISTEMA DE INFORMACION 7](#_bookmark9)
     1. [ELEMENTOS DE UN SISTEMA DE INFORMACIÓN 7](#_bookmark10)
     2. [TIPOS DE SISTEMAS DE INFORMACION 9](#_bookmark12)
        1. [SISTEMA DE PROCESAMIENTO DE TRANSACCIONES 10](#_bookmark14)
        2. [SISTEMA DE APOYO A LA TOMA DE DECISIONES 10](#_bookmark15)
  2. [METODOLOGIA SCRUM 11](#_bookmark16)
     1. [VISION GENERAL DE SCRUM 11](#_bookmark17)
     2. [EL EQUIPO SCRUM 12](#_bookmark19)
        1. [DUEÑO DE PRODUCTO (PRODUCT OWNER) 13](#_bookmark20)
        2. [MAESTRO SCRUM (SCRUM MASTER) 13](#_bookmark21)
        3. [EL EQUIPO DE DESARROLLO (DEVELOPMENT TEAM) 13](#_bookmark22)
     3. [PROCESOS DE SCRUM 14](#_bookmark23)
  3. [COMERCIO ELECTRONICO (E-COMMERCE) 15](#_bookmark24)
     1. [NEGOCIO A CONSUMIDOR B2C (BUSINESS TO CONSUMER) 17](#_bookmark25)
        1. [VENTAJAS DE B2C 17](#_bookmark26)
     2. [PASARELA DE PAGO 18](#_bookmark27)
        1. [PAYPAL 18](#_bookmark28)
  4. [LENGUAJE UNIFICADO DE MODELADO (UML) 20](#_bookmark30)
     1. [MODELOS 21](#_bookmark31)
     2. [DIAGRAMAS 22](#_bookmark32)
        1. [DIAGRAMA DE CASOS DE USOS 22](#_bookmark33)
        2. [DIAGRAMAS DE CLASES 25](#_bookmark36)
        3. [DIAGRAMAS DE SECUENCIA 28](#_bookmark39)
        4. [DIAGRAMA DE ESTADOS 29](#_bookmark42)
        5. [DIAGRAMA DE ACTIVIDADES 30](#_bookmark45)
  5. [TECNICA DE MODELADO DE OBJETOS (OMT) 32](#_bookmark48)
     1. [ANALISIS DE OBJETOS 33](#_bookmark50)
        1. [MODELO DE OBJETOS 34](#_bookmark51)
        2. [MODELO DINAMICO 34](#_bookmark52)
        3. [MODELO FUNCIONAL 35](#_bookmark53)
  6. [ARQUITECTURA DEL SOFTWARE 35](#_bookmark55)
     1. [PATRON MODELO VISTA CONTROLADOR 36](#_bookmark56)
        1. [EL MODELO 37](#_bookmark58)
        2. [LA VISTA 37](#_bookmark59)
        3. [EL CONTROLADOR 38](#_bookmark61)
     2. [ARQUITECTURA DE 3 CAPAS 38](#_bookmark62)
        1. [CAPA DE PRESENTACIÓN 40](#_bookmark64)
        2. [CAPA DE REGLAS DE NEGOCIO (EMPRESARIAL) 40](#_bookmark65)
        3. [CAPA DE DATOS 40](#_bookmark66)
  7. [SEGURIDAD 41](#_bookmark67)
     1. [PROPIEDADES DE UN SISTEMA DE INFORMACION SEGURO 41](#_bookmark68)
        1. [INTEGRIDAD 41](#_bookmark69)
        2. [CONFIDENCIALIDAD 41](#_bookmark70)
        3. [DISPONIBILIDAD 42](#_bookmark71)
     2. [MÉTODOS 42](#_bookmark72)
        1. [CORTAFUEGOS (FIREWALL) 42](#_bookmark73)
        2. [AUTENTICACIÓN Y AUTORIZACIÓN 42](#_bookmark74)
        3. [CIFRADO DE DATOS 42](#_bookmark75)
        4. [SESIÓN 43](#_bookmark76)
  8. [MODELO DE ESTIMACION DE COSTOS COCOMO 43](#_bookmark77)
     1. [MODELO BASICO 45](#_bookmark78)
     2. [MODELO INTERMEDIO 45](#_bookmark80)
     3. [MODELO DETALLADO 46](#_bookmark82)
  9. [METRICAS DE CALIDAD DEL SOFTWARE 47](#_bookmark83)
     1. [EL ESTANDAR ISO/IEC 25010 47](#_bookmark84)
        1. [ADECUACIÓN FUNCIONAL 48](#_bookmark86)
        2. [EFICIENCIA DE DESEMPEÑO 48](#_bookmark87)
        3. [COMPATIBILIDAD 49](#_bookmark88)
        4. [USABILIDAD 49](#_bookmark89)
        5. [FIABILIDAD 50](#_bookmark90)
        6. [SEGURIDAD 51](#_bookmark91)
        7. [MANTENIBILIDAD 52](#_bookmark92)
        8. [PORTABILIDAD 53](#_bookmark93)
  10. [HERRAMIENTAS DE DESARROLLO WEB 53](#_bookmark94)
      1. [SISTEMA GESTOR DE BASE DE DATOS POSTGRESQL 53](#_bookmark95)
         1. [FUNCIONAMIENTO 54](#_bookmark96)
         2. [BASE DE DATOS RELACIONAL 54](#_bookmark97)
      2. [LENGUAJE DE PROGRAMACION PHP (“PREPROCESADOR DE HIPERTEXTO”) 54](#_bookmark98)
      3. [FRAMEWORK DE DESARROLLO CODEIGNITER 55](#_bookmark99)
         1. [CARACTERISTICAS GENERALES DE CODEIGNITER 56](#_bookmark100)
      4. [HTML (“LENGUAJE DE MARCADO DE HIPERTEXTO”) 57](#_bookmark101)
      5. [JAVASCRIPT 58](#_bookmark102)
         1. [JQUERY 58](#_bookmark103)
         2. [AJAX (JAVASCRIPT ASÍNCRONO Y XML) 58](#_bookmark104)

[CAPITULO 2. JUSTIFICACIÓN DEL PROYECTO 60](#_bookmark105)

* 1. [JUSTIFICACION CIENTIFICO-TECNICA 60](#_bookmark106)
  2. [JUSTIFICACION ECONOMICA 60](#_bookmark107)
  3. [JUSTIFICACION SOCIAL 60](#_bookmark108)

[CAPITULO 3. DISEÑO TEORICO DE LA INVESTIGACIÓN 61](#_bookmark109)

* 1. [PLANTEAMIENTO DEL PROBLEMA 61](#_bookmark110)
  2. [FORMULACION DEL PROBLEMA 61](#_bookmark111)
     1. [DELIMITACION DEL CONTENIDO 61](#_bookmark112)
     2. [DELIMITACION TEMPORAL 62](#_bookmark113)
     3. [DELIMITACION ESPACIAL 62](#_bookmark114)
  3. [OBJETIVOS DE LA INVESTIGACION 62](#_bookmark115)
     1. [OBJETIVO GENERAL 62](#_bookmark116)
     2. [OBJETIVOS ESPECIFICOS 62](#_bookmark117)

[CAPITULO IV. DISEÑO O DISPOSITIVO DE PRUEBA 64](#_bookmark118)

* 1. [ARGUMENTACION DEL TIPO DE INVESTIGACION Y TIPO DE DISEÑO DE](#_bookmark119) [INVESTIGACION 64](#_bookmark119)
     1. [TIPO DE INVESTIGACION 64](#_bookmark120)
     2. [DISEÑO DE LA INVESTIGACION 64](#_bookmark121)
     3. [MÉTODOS TÉCNICAS E INSTRUMENTOS DE INVESTIGACIÓN A UTILIZAR 64](#_bookmark122)
        1. [MÉTODO 64](#_bookmark123)
        2. [TÉCNICAS 64](#_bookmark124)
        3. [HERRAMIENTAS 65](#_bookmark125)

[CAPITULO V. DESCRIPCIÓN DEL PROYECTO 65](#_bookmark126)

* 1. [METODOLOGIA SCRUM 65](#_bookmark127)
     1. [ROLES DEL PROYECTO SCRUM 65](#_bookmark128)
     2. [ARTEFACTOS SCRUM 66](#_bookmark129)
        1. [HISTORIAS DE USUARIO 66](#_bookmark130)
        2. [PRODUCT BACKLOG 68](#_bookmark131)
        3. [SPRINT BACKLOG 74](#_bookmark132)
        4. [BURNDOWN CHART (QUEMADO DE TAREAS) 75](#_bookmark133)
        5. [DIAGRAMA DE PAQUETES 77](#_bookmark135)
        6. [ACTORES DEL SISTEMA 78](#_bookmark137)
        7. [CASOS DE USO DEL SISTEMA 79](#_bookmark139)
        8. [DIAGRAMAS DE SECUENCIA DEL SISTEMA 93](#_bookmark149)
        9. [DIAGRAMA DE ACTIVIDADES DEL SISTEMA 101](#_bookmark166)
        10. [DIAGRAMAS DE COLABORACION DEL SISTEMA 106](#_bookmark177)
        11. [DIAGRAMA DE ESTADOS DEL SISTEMA 111](#_bookmark187)
        12. [DIAGRAMA DE CLASES DEL SISTEMA 116](#_bookmark198)
        13. [MODELO FUNCIONAL 117](#_bookmark200)
        14. [MODELO NAVEGACIONAL 119](#_bookmark203)
  2. [ARQUITECTURA DEL SISTEMA DE 3 CAPAS 121](#_bookmark204)
     1. [CAPA DE PRESENTACIÓN 121](#_bookmark205)
     2. [CAPA DE NEGOCIO 121](#_bookmark206)

[4.1.4. CAPA DE DATOS 121](#_bookmark207)

* 1. [PRUEBAS DE SOFTWARE 122](#_bookmark209)
     1. [PRUEBA DE CAJA NEGRA 122](#_bookmark210)
  2. [METRICAS DE CALIDAD DEL SOFTWARE 128](#_bookmark211)
     1. [FUNCIONALIDAD 128](#_bookmark212)
     2. [CONFIABILIDAD 133](#_bookmark213)
     3. [FIABILIDAD 133](#_bookmark214)
     4. [FACILIDAD DE USO 134](#_bookmark215)
     5. [CAPACIDAD DE MANTENIMIENTO 135](#_bookmark216)
        1. [MATENIMIENTO CORRECTIVO 135](#_bookmark217)
        2. [MANTENIMIENTO ADAPTATIVO 136](#_bookmark218)
        3. [MANTENIMIENTO PREVENTIVO 136](#_bookmark219)
     6. [PORTABILIDAD 137](#_bookmark220)
        1. [FACILIDAD DE INSTALACION 137](#_bookmark221)
        2. [FACILIDAD DE AJUSTE 137](#_bookmark222)
        3. [FACILIDAD DE ADAPTACION 137](#_bookmark223)
  3. [RESULTADOS ESPERADOS, APORTES, IMPACTOS Y OPORTUNIDADES](#_bookmark224)

[138](#_bookmark224)

* + 1. [APORTES 138](#_bookmark225)
    2. [IMPACTO 138](#_bookmark226)
    3. [OPORTUNIDADES 139](#_bookmark227)
  1. [PRESUPUESTO DEL PROYECTO 139](#_bookmark228)
     1. [MODELO COCOMO 139](#_bookmark229)
     2. [ANALISIS DE COSTOS 142](#_bookmark230)
        1. [COSTO DEL SOFTWARE 142](#_bookmark231)
        2. [COSTO DEL HARDWARE 143](#_bookmark232)
        3. [COSTO DE LA INVESTIGACION 144](#_bookmark233)
        4. [COSTO TOTAL DEL PROYECTO 145](#_bookmark234)
        5. [RESUMEN DEL PRESUPUESTO 145](#_bookmark235)
     3. [CALCULO DEL VAN Y EL TIR 146](#_bookmark236)
        1. [VALOR ACTUAL NETO (VAN) 146](#_bookmark237)
        2. [TASA INTERNA DE RETORNO (TIR) 148](#_bookmark238)
  2. [SEGURIDAD DEL SOFTWARE 149](#_bookmark239)

1. [CONCLUSIONES 150](#_bookmark240)
2. [RECOMENDACIONES 151](#_bookmark241)
3. [BIBLIOGRAFÍA 152](#_bookmark242)

**INDICE DE FIGURAS**

[**FIGURA 1.** ORGANIGRAMA – GROUPWARE SYSTEMS 4](#_bookmark5)

[**FIGURA 2.** ACTIVIDADES DE UN SISTEMA DE INFORMACIÓN 8](#_bookmark11)

[**FIGURA 3.** TIPOS DE SISTEMAS DE INFORMACION 10](#_bookmark13)

[**FIGURA 4.** VISION GENERAL DE SCRUM 12](#_bookmark18)

[**FIGURA 5.** PROCESO DE ENVIO Y RECEPCIÓN DE DINERO A TRAVÉS DE PAYPAL](#_bookmark29)

[........................................................................................................................................................... 19](#_bookmark29)

[**FIGURA 6.** COMPONENTES DEL DIAGRAMA DE CASOS DE USO 24](#_bookmark34)

[**FIGURA 7.** DIAGRAMA DE CASOS DE USO 25](#_bookmark35)

[**FIGURA 8.** SIMBOLOGÍA DEL DIAGRAMA DE CLASES 26](#_bookmark37)

[**FIGURA 9.** DIAGRAMA DE CLASES 26](#_bookmark38)

[**FIGURA 10.** COMPONENTES DEL DIAGRAMA DE SECUENCIA 28](#_bookmark40)

[**FIGURA 11.** DIAGRAMA DE SECUENCIA 29](#_bookmark41)

[**FIGURA 12.** COMPONENTES DEL DIAGRAMA DE ESTADOS 30](#_bookmark43)

[**FIGURA 13.** DIAGRAMA DE ESTADOS 30](#_bookmark44)

[**FIGURA 14.** COMPONENTES DEL DIAGRAMA DE ACTIVIDADES 31](#_bookmark46)

[**FIGURA 15.** DIAGRAMA DE ACTIVIDADES 31](#_bookmark47)

[**FIGURA 16**. CICLO DE VIDA OMT 33](#_bookmark49)

[**FIGURA 17**. DIAGRAMA DE FLUJO DE DATOS – OMT 35](#_bookmark54)

[**FIGURA 18.** MODELO VISTA CONTROLADOR 36](#_bookmark57)

[**FIGURA 19.** MODELO VISTA CONTROLADOR- DIAGRAMA DE SECUENCIA 38](#_bookmark60)

[**FIGURA 20.** ARQUITECTURA 3 CAPAS 39](#_bookmark63)

[**FIGURA 21.** TABLA DE CONSTANTES PARA CALCULAR DIFERENTES ASPECTOS](#_bookmark79) [DE COSTES 45](#_bookmark79)

[**FIGURA 22.** TABLA DE VALORES MODO INTERMEDIO – COCOMO 46](#_bookmark81)

[**FIGURA 23.** CALIDAD DEL PRODUCTO SOFTWARE – ISO/IEC 25010 47](#_bookmark85)

[**FIGURA 24.** BURNDOWN CHART DEL SISTEMA 76](#_bookmark134)

[**FIGURA 25.** DIAGRAMA DE PAQUETES-SISTEMA DE VENTAS 77](#_bookmark136)

[**FIGURA 26.** ACTORES-SISTEMA DE VENTAS 79](#_bookmark138)

[**FIGURA 27.** DIAGRAMA DE CASOS DE USO-SESIONES 79](#_bookmark140)

[**FIGURA 28**. DIAGRAMA DE CASOS DE USO- GESTION DE PRODUCTOS 81](#_bookmark141)

[**FIGURA 29.** DIAGRAMA DE CASOS DE USO – VENTAS FISICAS 83](#_bookmark142)

[**FIGURA 30.** DIAGRAMA DE CASOS DE USO – PRODUCTOS 85](#_bookmark143)

[**FIGURA 31.** CASOS DE USO – VENTAS ONLINE 87](#_bookmark144)

[**FIGURA 32.** CASO DE USO – VENTAS ONLINE 88](#_bookmark145)

[**FIGURA 33.** DIAGRAMA DE CASOS DE USO – COMPRAS Y PROVEEDORES 89](#_bookmark146)

[**FIGURA 34.** DIAGRAMA DE CASOS DE USO – CHARTS Y REPORTES 90](#_bookmark147)

[**FIGURA 35.** DIAGRAMA DE CASOS DE USO – CONTACTOS 92](#_bookmark148)

[**FIGURA 36.** DIAGRAMA DE SECUENCIA – REGISTRO DE EMPLEADOS 93](#_bookmark150)

[**FIGURA 37.** DIAGRAMA DE SECUENCIA - INIICIAR SESION 94](#_bookmark151)

[**FIGURA 38**. DIAGRAMA DE SECUENCIA – REGISTRARSE 94](#_bookmark152)

[**FIGURA 39.** DIAGRAMA DE SECUENCIA – REGISTRO DE CATEGORIA 95](#_bookmark153)

[**FIGURA 40.** DIAGRAMA DE SECUENCIA – REGISTRAR MARCA 95](#_bookmark154)

[**FIGURA 41.** DIAGRAMA DE SECUENCIA – REGISTRO DE PRODUCTOS 96](#_bookmark155)

[**FIGURA 42.** DIAGRAMA DE SECUENCIA – VENTAS FISICAS 96](#_bookmark156)

[**FIGURA 43.** DIAGRAMA DE SECUENCIA – PRODUCTOS 97](#_bookmark157)

[**FIGURA 44.** DIAGRAMA DE SECUENCIA – VENTAS ONLINE 97](#_bookmark158)

[**FIGURA 45.** DIAGRAMA DE SECUENCIA – REGISTRO DE PROVEEDORES 98](#_bookmark159)

[**FIGURA 46.** DIAGRAMA DE SECUENCIA – REGISTRO DE EMPLEADO DE](#_bookmark160) [PROVEEDOR 98](#_bookmark160)

[**FIGURA 47.** DIAGRAMA DE SECUENCIA – REGISTRO DE COMPRAS 99](#_bookmark161)

[**FIGURA 48**. DIAGRAMA DE SECUENCIA – REPORTE DE PRODUCTOS 99](#_bookmark162)

[**FIGURA 49.** DIAGRAMA DE SECUENCIA – CHART/REPORTE VENTAS 100](#_bookmark163)

[**FIGURA 50.** DIAGRAMA DE SECUENCIA – CHART/REPORTE COMPRAS 100](#_bookmark164)

[**FIGURA 51**. DIAGRAMA DE SECUENCIA – CONTACTOS 101](#_bookmark165)

[**FIGURA 52.** DIAGRAMA DE ACTIVIDADES – REGISTRAR EMPLEADO 101](#_bookmark167)

[**FIGURA 53.** DIAGRAMA DE ACTIVIDADES – INICIAR SESION 102](#_bookmark168)

[**FIGURA 54**. DIAGRAMA DE ACTIVIDADES – REGISTRARSE 102](#_bookmark169)

[**FIGURA 55.** DIAGRAMA DE ACTIVIDADES – GESTION DE PRODUCTOS 103](#_bookmark170)

[**FIGURA 56.** DIAGRAMA DE ACTIVIDADES – VENTAS FISICAS 103](#_bookmark171)

[**FIGURA 57.** DIAGRAMA DE ACTIVIDADES – PRODUCTOS 104](#_bookmark172)

[**FIGURA 58.** DIAGRAMA DE ACTIVIDADES – VENTAS ONLINE 104](#_bookmark173)

[**FIGURA 59.** DIAGRAMA DE ACTIVIDADES – COMPRAS Y PROVEEDORES 105](#_bookmark174)

[**FIGURA 60.** DIAGRAMA DE ACTIVIDADES – CHARTS Y REPORTES 105](#_bookmark175)

[**FIGURA 61.** DIAGRAMA DE ACTIVIDADES – CONTACTOS 106](#_bookmark176)

[**FIGURA 62**. DIAGRAMA DE COLABORACION – INICIAR SESION 106](#_bookmark178)

[**FIGURA 63.** DIAGRAMA DE COLABORACION – REGISTRARSE 107](#_bookmark179)

[**FIGURA 64.** DIAGRAMA DE COLABORACION – REGISTRARSE 107](#_bookmark180)

[**FIGURA 65.** DIAGRAMA DE COLABORACION – GESTION DE PRODUCTO 108](#_bookmark181)

[**FIGURA 66.** DIAGRAMA DE COLABORACION - VENTAS FISICA 108](#_bookmark182)

[**FIGURA 67.** DIAGRAMA DE COLABORACION – PRODUCTOS 109](#_bookmark183)

[**FIGURA 68.** DIAGRAMA DE COLABORACION – VENTAS ONLINE 109](#_bookmark184)

[**FIGURA 69.** DIAGRAMA DE COLABORACION – REGISTRO DE PROVEEDORES 110](#_bookmark185)

[**FIGURA 70.** DIAGRAMA DE COLABORACION – COMPRAS 110](#_bookmark186)

[**FIGURA 71.** DIAGRAMA DE ESTADOS – REGISTRAR EMPLEADO 111](#_bookmark188)

[**FIGURA 72.** DIAGRAMA DE ESTADOS – INICIAR SESION 111](#_bookmark189)

[**FIGURA 73.** DIAGRAMA DE ESTADOS – REGISTRARSE 112](#_bookmark190)

[**FIGURA 74.** DIAGRAMA DE ESTA DOS – GESTION DE PRODUCTOS 112](#_bookmark191)

[**FIGURA 75.** DIAGRAMA DE ESTADOS – VENTAS FISICAS 113](#_bookmark192)

[**FIGURA 76.** DIAGRAMA DE ESTADOS – PRODUCTOS 113](#_bookmark193)

[**FIGURA 77.** DIAGRAMA DE ESTADOS – VENTAS ONLINE 114](#_bookmark194)

[**FIGURA 78**. DIAGRAMA DE ESTADOS – REGISTRO DE PROVEEDOR 114](#_bookmark195)

[**FIGURA 79.** DIAGRAMA DE ESTADOS-COMPRAS 115](#_bookmark196)

[**FIGURA 80.** DIAGRAMA DE ESTADOS – CHARTS/REPORTES 115](#_bookmark197)

[**FIGURA 81.** DIAGRAMA DE CLASES- COMPRA Y VENTA DE PRODUCTOS ONLINE](#_bookmark199)

[......................................................................................................................................................... 116](#_bookmark199)

[**FIGURA 82.** MODELO FUNCIONAL – CIENTE 117](#_bookmark201)

[**FIGURA 83.** MODELO FUNCIONAL – COMPRAS Y VENTAS FISICAS 118](#_bookmark202)

[**FIGURA 84.** ARQUITECTURA DEL SISTEMA - 3 CAPAS 122](#_bookmark208)

## INDICE DE TABLAS

**TABLA 1.** SISTEMA DE E-COMMERCE, Luis Alberto Linares Gonzales 5

**TABLA 2.** SISTEMA WEB PARA EL CONTROL DE VENTAS Y FACTURACION USANDO AGENTES INTELIGENTES, Patricia Evelyn Mattacatacora. 5

**TABLA 3.** SISTEMA DE COMERCIO ELECTRONICO BASADO EN LA PLATAFORMA PRESTASHOP Y E-COMMERCE, Diego Umaña Zenteno 6

**TABLA 4.** ROLES DEL PROYECTO SCRUM 66

**TABLA 5.** HISTORIAS DE USUARIO DEL SISTEMA 68

**TABLA 6.** PRODUCT BACKLOG-SISTEMA DE VENTAS 73

**TABLA 7.** SPRINT BACKLOG-SISTEMA DE VENTAS 75

**TABLA 8**. CASO DE USO-SESIONES 81

**TABLA 9.** CASO DE USO- GESTION DE PRODUCTOS 83

**TABLA 10.** CASO DE USO –VENTAS FISICAS 84

**TABLA 11.** CASO DE USO – PRODUCTOS 86

**TABLA 12.** CASO DE USO – COMPRAS Y VENDEDORES 90

**TABLA 13.** CASO DE USO – CHARTS Y REPORTES 92

**TABLA 14.** CASO DE USO – CONTACTOS 93

**TABLA 34.** PRUEBA DE CAJA DE NEGRA DEL SISTEMA 127

**TABLA 35.** DOMINIOS DE INFORMACION DE PUNTO FUNCION 129

**TABLA 36.** ENTRADAS PARA EL CALCULO DE FUNCIONALIDAD 129

**TABLA 37.** CALCULO DE PUNTOS DE CUENTA TOTAL 130

**TABLA 38.** CALCULO DE AJUSTES DE COMPLEJIDAD DEL PUNTO DE FUNCION . 131

**TABLA 39.** ESCALA DE PUNTO DE FUNCION 132

**TABLA 40.** RESULTADOS DE LA FACILIDAD DE USO DEL SISTEMA 134

**TABLA 41.** METRICAS DE CALIDAD DEL SISTEMA 138

**TABLA 42.** FACTOR DE AJUSTE – COCOMO 141

**TABLA 43**. COSTO TOTAL DEL SOFTWARE 143

**TABLA 44.** COSTO TOTAL DEL HARDWARE 144

**TABLA 45.** COSTO TOTAL DE LA INVESTIGACION 144

**TABLA 46.** RESUMEN DEL PRESUPUESTO TOTAL DEL PROYECTO 146

**TABLA 47.** SEGURIDAD DEL SOFTWARE 149

## INTRODUCCIÓN

Hoy en día en el siglo XXI, el comercio electrónico web se ha vuelto imprescindible para la mayoría de las empresas en diferentes rubros, que han decidido por automatizar y facilitar el uso de las nuevas tendencias tecnológicas dentro de las mismas lo que ha provocado que el ser humano se vuelva prescindible, simplemente un supervisor, o un mediador dentro de las empresas.

El presente proyecto de grado, surge a través de la necesidad de la empresa GROUPWARE SYSTEMS de contar con un sistema de comercio eléctrico web, para expandir el negocio, y facilitar a sus clientes la adquisición de diferentes productos de manera remota, registrando las compras y ventas para tener un mejor control.

Es por eso que con el uso de este sistema de información web el cliente sentirá satisfacción y la comodidad de poder realizar distintas tareas de igual o mejor manera como si estuviera presente en la empresa. Por tal motivo surge la necesidad de poder realizar pagos de dinero a distancia para que el cliente de la empresa se sienta a gusto y el producto llegue hasta la puerta de su casa. Paypal ofrece una pasarela de pagos que permite que un cliente a través de una cuenta Paypal pueda realizar pagos a una determinada empresa por un servicio solicitado.

###### PRESENTACIÓN DEL PROBLEMA DE INVESTIGACIÓN

La mayoría de las empresas pequeñas/medianas dedicadas al rubro de la venta de artefactos electrónicos manejan diferentes tipos de productos, y tienen diferentes proveedores, debido a que su objetivo principal es tener una gran cantidad de ventas realizadas dentro de un lapso de tiempo mínimo, en muchos casos olvidan registrar o llevar un control de las ventas que se realizan asimismo el control interno de las compras y ventas se las realiza de manera manual y esto ocasiona perdida de registros y lleva tiempo obtener información de los mismos. Sin embargo La importancia de este proyecto radica en tener una herramienta que sea capaz de incrementar las ventas de los diferentes productos que ofrece la empresa y poder manejar de manera sistematizada el control de las compras y ventas.

###### ANTECEDENTES

###### ANTECEDENTES DE LA EMPRESA

La empresa GROUPWARE SYSTEMS nació bajo la idea de un grupo de profesionales emprendedores de formar un equipo de trabajo dedicado en el área de venta de productos y soluciones tecnológicas, el 29 de abril del 2016 se decide abrir en la Av. Juan Pablo II zona Ferropetrol una pequeña oficina informática con el fin de poder realizar mantenimiento, configuración, reparación, actualización de equipos informáticos, debido a la alta demanda de venta de aparatos electrónicos GROUPWARE SYSTEMS decide ampliar los servicios prestados e incursiona en la venta de computadoras, equipos de redes y artefactos electrónicos es así que GROUPWARE SYSTEMS se consolida como una empresa joven y de gran capacidad de hacer negocios dentro del mercado boliviano. Sin embargo GROUPWARE SYSTEMS al consolidarse como empresa decide proveer sus

servicios a entidades públicas y municipales teniendo como objetivo crecer empresarialmente. Es por eso que la empresa decide cambiar de domicilio y mudarse a una zona más céntrica como ser la Ceja de El Alto específicamente hoy en día funciona dentro de la Galería Inti, en el Piso 2, Oficina 3-0, desde ahí provee los servicios mencionados.

###### MISION, VISION Y ORGANIGRAMA INSTITUCIONAL

* + - * + **MISION**

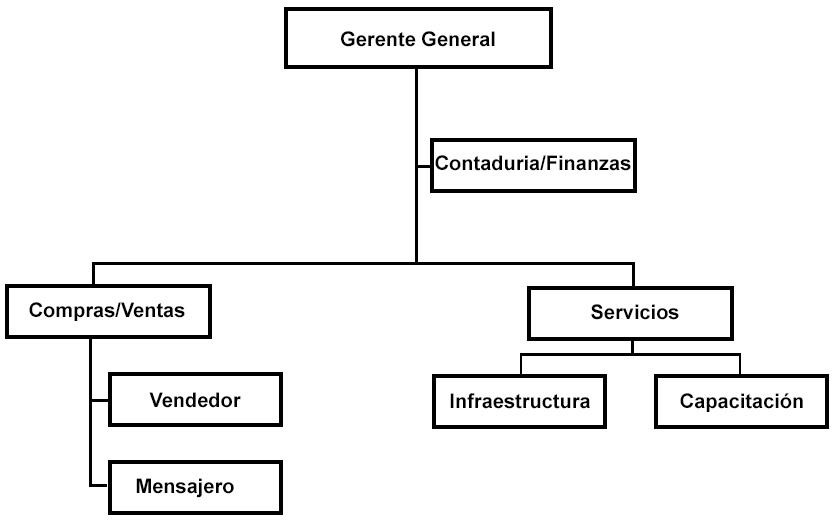
Ser la empresa líder en la comercialización de productos informáticos y electrónicos dentro del mercado boliviano, ya que el cliente es lo más importante de la empresa por lo tanto es importante tener consecuencia y buen trato hacia el mismo, ser en un futuro una de las empresas más reconocidas e influyentes del mercado.

###### VISION

Ofrecer un servicio de calidad a nuestra clientela de acuerdo a las necesidades planteadas, trabajando bajo estándares de calidad para que los clientes se sientan satisfechos y seguros de poder contar con nuestra empresa.

###### ORGANIGRAMA DE LA EMPRESA

La empresa GROUPWARE SYSTEMS cuenta con 2 departamentos elementales que son: Compras/Ventas y Servicios los cuales albergan varios empleados con el fin de realizar un trabajo calificado.



***FIGURA 1.*** *ORGANIGRAMA – GROUPWARE SYSTEMS*

***FUENTE.*** *GROUPWARE SYSTEMS*

###### ANTECEDENTES DEL PROYECTO

Dentro de la búsqueda realizada en los diferentes proyectos de grado enmarcados en el contexto de la problemática planteada tenemos los siguientes:

|  |  |  |  |
| --- | --- | --- | --- |
| **AUTOR** | **TITULO** | **INSTITUCION** | **UNIVERSIDAD** |
| **LUIS ALBERTO LINARES GONZALES** | SISTEMA DE E- COMMERCE | EMPRESA AHAYO-BOL | UNIVERSIDAD MAYOR DE SAN ANDRES |
| **DESCRIPCION** | Dentro de este proyecto se describe que la empresa “Ahayo  Bol” no cuenta con proyecciones locales, departamentales, nacionales e internacionales, por tal motivo el autor pretende | | |

incrementar las proyecciones de la empresa con el fin de tener mayores ingresos económicos, y de tal manera incrementar los clientes y la cantidad de stock con la cuenta. Dentro del trabajo mencionado las ventas se lo realiza mediante la modalidad B2C “Business to Consumer” venta de productos finales a un consumidor (Tiendas virtuales como Amazon.com)

***TABLA 1.*** *SISTEMA DE E-COMMERCE, Luis Alberto Linares Gonzales*

***FUENTE.*** *ELABORACION PROPIA*

|  |  |  |  |
| --- | --- | --- | --- |
| **AUTOR** | **TITULO** | **INSTITUCION** | **UNIVERSIDAD** |
| **PATRICIA EVELYN MATTACATACORA** | SISTEMA WEB PARA EL CONTROL DE VENTAS Y FACTURACIÓN USANDO AGENTES  INTELIGENTES | IMPORTADORA DE FÁRMACOS “IMESMAT” | UNIVERSIDAD MAYOR DE SAN ANDRES |
| **DESCRIPCION** | En este proyecto el autor plantea solucionar los problemas que tiene la empresa “IMESMAT” en el control de la información de las ventas y controlar la facturación de dicha empresa, automatizando y optimizando los procesos manuales para que conlleve menos tiempo durante la realización de los mismos. Al mismo tiempo se pretende realizar un módulo de facturación para tener un mejor  control. | | |

***TABLA 2.*** *SISTEMA WEB PARA EL CONTROL DE VENTAS Y FACTURACION USANDO AGENTES*

*INTELIGENTES, Patricia Evelyn Mattacatacora.*

***FUENTE.*** *ELABORACION PROPIA*

|  |  |  |  |
| --- | --- | --- | --- |
| **AUTOR** | **TITULO** | **INSTITUCION** | **UNIVERSIDAD** |
| **DIEGO UMAÑA ZENTENO** | SISTEMA DE COMERCIO ELECTRONICO BASADO EN LA PLATAFORMA PRESTASHOP Y F-  COMMERCE | JULIANA BOUTIQUE | UNIVERSIDAD CATOLICA BOLIVIANA “SAN PABLO” |
| **DESCRIPCION** | Con el uso de Prestashop el autor pretende realizar un software, dinámico, llamativo y funcional de manera rápida y poco complicada para que este sea capaz de revolucionar el enfoque de la empresa al uso de recursos tecnológicos online para poder optimizar tiempos y métodos de pago realizando las transacciones mediante internet, utilizando el lenguaje de programación PHP y el motor gestor de base de datos MySql,  que en la gestión desarrollada no tenía costo la licencia de uso. | | |

***TABLA 3.*** *SISTEMA DE COMERCIO ELECTRONICO BASADO EN LA PLATAFORMA PRESTASHOP Y E-*

*COMMERCE, Diego Umaña Zenteno.*

***FUENTE.*** *ELABORACION PROPIA*

## DESARROLLO

### CAPITULO 1. MARCO TEORICO

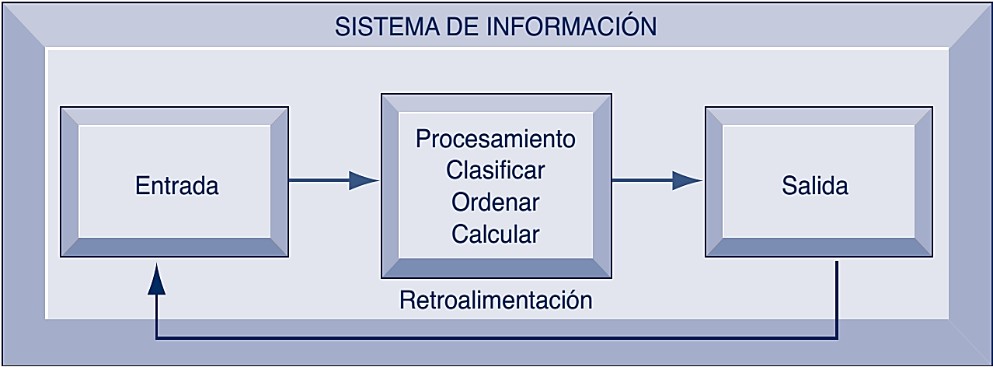
###### SISTEMA DE INFORMACION

Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. En un sentido amplio, un sistema de información no necesariamente incluye equipo electrónico (hardware). Sin embargo, en la práctica se utiliza como sinónimo de “sistema de información computarizado”. (Cohen & Asin, 2000)

Otros autores como (Rodriguez Rodriguez & Daureo Campillo, 2003) define que un Sistema de Información (S.I.) es un conjunto de procedimientos, manuales y automatizados, y de funciones dirigidas a la recogida, elaboración, evaluación, almacenamiento, recuperación, condensación y distribución de informaciones dentro de una organización, orientado a promover el flujo de las mismas desde el punto en el que se generan hasta el destinatario final de las mismas".

###### ELEMENTOS DE UN SISTEMA DE INFORMACIÓN

Hay tres actividades en un sistema de información que producen los datos necesarios para que las organizaciones tomen decisiones, controlen las operaciones, analicen problemas y creen nuevos productos o servicios. Estas actividades son: entrada, procesamiento y salida (vea la figura 1). La entrada captura o recolecta los datos en crudo desde el interior de la organización o a través de su entorno externo. El procesamiento convierte esta entrada en bruto en un formato significativo. La salida transfiere la información procesada a las personas que harán uso de ella, o a las actividades para las que se utilizará. Los sistemas de

información también requieren retroalimentación: la salida que se devuelve a los miembros apropiados de la organización para ayudarles a evaluar o corregir la etapa de entrada. (Laudon & Laudon, 2012)

**FIGURA 2.** ACTIVIDADES DE UN SISTEMA DE INFORMACIÓN

***FUENTE:*** (Laudon & Laudon, 2012)

* + - * **ENTRADA DE INFORMACION:** Es el proceso por el cual se recaba los datos que se requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que proporciona directamente el usuario, mientras que las automáticas son datos o información que provienen de otros sistemas o módulos, los datos pueden ser recolectados por medio de diferentes medios como ser: medios magnéticos, dispositivos USB, código de barras, scanner, papeles,

cartulinas, OTROS.

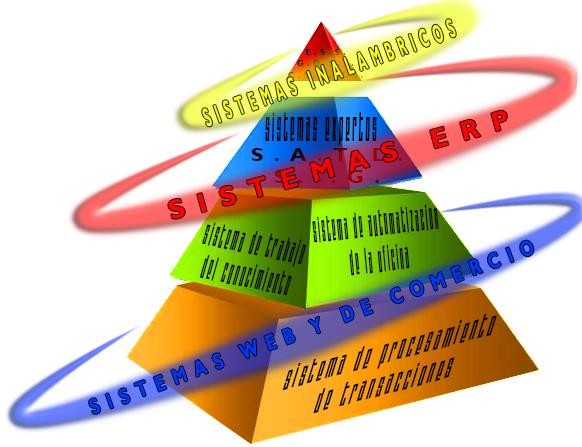
* + - * **PROCESAMIENTO DE INFORMACION:** Es la acción de efectuar cálculos, clasificación y/o operaciones mediante las cuales se pueden realizar con datos introducidos recientemente en el sistema o con datos que ya estén almacenados. Esta característica permite la transformación de datos en fuentes de información que pueden ser utilizadas para la toma de decisiones.
      * **SALIDA DE INFORMACION**: Es la etapa por el cual el sistema muestra la información ya procesada, interpretada y que tiene sentido para su uso

posterior como ser: servir para nuevas entradas de otro sistema de información o modulo. Las unidades típicas de salida son las impresoras, dispositivos de almacenamiento USB, entre otros.

###### TIPOS DE SISTEMAS DE INFORMACION

Los sistemas de información se desarrollan con diversos propósitos, según las necesidades de la empresa (*Véase la Figura 3)*. Los sistemas de procesamiento de transacciones (TPS, Transaction Processing Systems) funcionan al nivel operativo de una organización, los sistemas de automatización de la oficina (OAS, Office Automañon Systems) y los sistemas de trabajo del conocimiento (KWS, Knowledge Work Systems) apoyan el trabajo al nivel del conocimiento. Los sistemas de información gerencial (MIS, Management Information Systems) y los sistemas de apoyo a la toma de decisiones (DSS, Decisión Support Systems) se encuentran entre los sistemas de alto nivel. (Kendall & Kendall, 2005).

Los sistemas de apoyo a la toma de decisiones en grupo (GDSS, Group Decisión Support Systems) y los sistemas de trabajo corporativo apoyado por computadora (CSCWS, Computer-Supported Collaborative Work Systems), descrito de manera más general, auxilian la toma de decisiones semiestructuradas o no estructuradas a nivel de grupo. (Kendall & Kendall, 2005)



***FIGURA 3.*** *TIPOS DE SISTEMAS DE INFORMACION*

***FUENTE.*** (Kendall & Kendall, 2005)

A continuación se cita los tipos de Sistemas de Información más relevantes:

###### SISTEMA DE PROCESAMIENTO DE TRANSACCIONES

Son sistemas encargados del procesamiento de transacciones a nivel empresarial, es decir son todos aquellos que se realizan de forma cotidiana o rutinaria, entre estos tenemos, el pago de nómina, facturación, entrega de stock y depósito de cheques. Estos varían de acuerdo al tipo de empresa.

###### SISTEMA DE APOYO A LA TOMA DE DECISIONES

Un Sistema de apoyo a la toma decisiones, en términos muy generales según (Finlay, 1994), es: Un sistema basado en computador que ayuda en el proceso de toma de decisiones. En términos más específicos, (Turban, 1995), define como: Un sistema de información basado en un computador interactivo, flexible y adaptable, especialmente desarrollado para apoyar la solución de un problema de gestión no estructurado para mejorar la toma de decisiones. Utiliza datos, proporciona una

interfaz amigable y permite la toma de decisiones en el propio análisis de la situación.

###### METODOLOGIA SCRUM

###### VISION GENERAL DE SCRUM

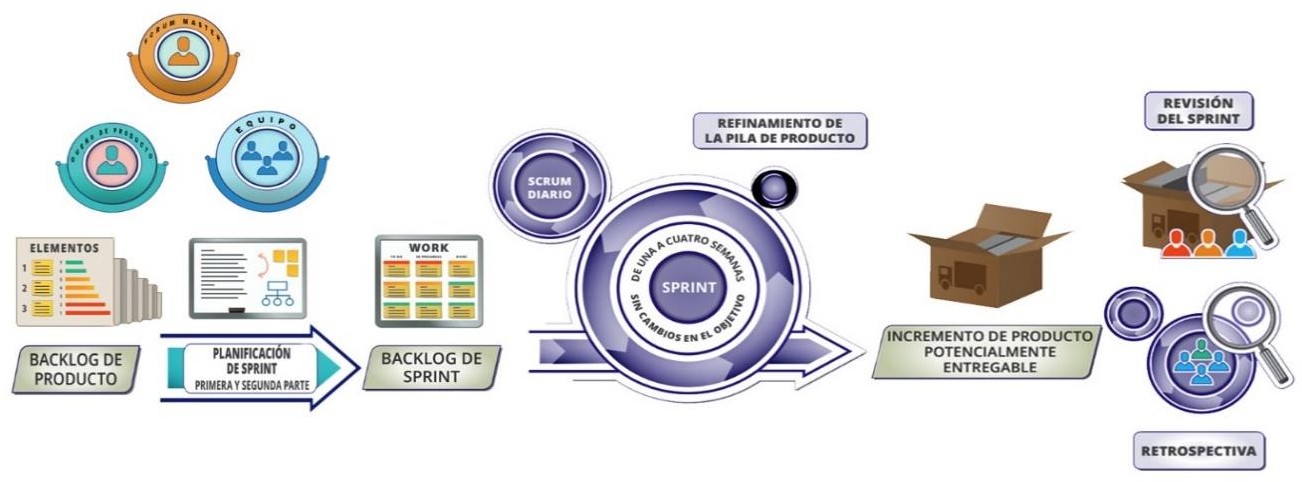
Un marco de trabajo por el cual las personas pueden acometer problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente. Scrum es:

* + - * Ligero
      * Fácil de entender
      * Extremadamente difícil de llegar a dominar

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. (Schwaber & Sutherland, 2013)

De manera más específica (Deemer, Benefield, Larman, & Vodde, 2012) en su libro SCRUM PRIMER definen que: Scrum es un marco de trabajo en el que equipos cross-funcionales pueden crear productos o desarrollar proyectos de una forma iterativa e incremental. El desarrollo se estructura en ciclos de trabajo llamados Sprints (también conocidos como iteraciones). Estas iteraciones no deben durar más de cuatro semanas cada una (siendo dos semanas la duración más habitual) y tienen lugar una tras otra sin pausa entre ellas.

El equipo acuerda un objetivo colectivo respecto a lo que creen que podrán entregar al final del Sprint, algo que sea tangible y que estará “terminado” por completo. Durante el Sprint no se podrán añadir nuevos elementos; Scrum se adapta a los cambios en el siguiente Sprint, pero el pequeño Sprint actual está pensado para concentrarnos en un objetivo pequeño, claro y relativamente estable.

Al final del Sprint, el Equipo revisa el Sprint con los diferentes Stakeholders (interesados e involucrados en el producto) y realiza una demostración de lo que han desarrollado. Se obtiene feedback que podrá ser incorporado en el siguiente Sprint. Scrum enfatiza un producto “funcionando” al final del Sprint que este realmente “terminado”.

***FIGURA 4.*** *VISION GENERAL DE SCRUM*

***FUENTE.*** (Deemer, Benefield, Larman, & Vodde, 2012)

###### EL EQUIPO SCRUM

Según (Schwaber & Sutherland, 2013) El Equipo Scrum consiste en un Dueño de Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master. Los Equipos Scrum son auto organizados y multifuncionales. Los equipos auto organizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. Los equipos multifuncionales tienen

todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo. El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad.

###### DUEÑO DE PRODUCTO (PRODUCT OWNER)

El propietario del producto (product owner) es quien toma las decisiones del cliente. Su responsabilidad es el valor del producto. Para simplificar la comunicación y toma de decisiones es necesario que este rol recaiga en una única persona. Si el cliente es una organización grande, o con varios departamentos, puede adoptar la forma de comunicación interna que consideren oportuna, pero en el equipo de desarrollo sólo se integra una persona en representación del cliente, y ésta debe tener el conocimiento suficiente del producto y las atribuciones, necesarias para tomar las decisiones que le corresponden. (Palacio, 2015)

###### MAESTRO SCRUM (SCRUM MASTER)

Es la persona encargada de enseñar la metodología scrum a cada integrante del equipo de trabajo, con el fin de poner en práctica la metodología, asegurándose de que cada uno siga las reglas y prácticas de Scrum. Asimismo el Scrum Master es un líder que está al servicio del Equipo Scrum. El Scrum Master ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum puede ser de ayuda y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum.

###### EL EQUIPO DE DESARROLLO (DEVELOPMENT TEAM)

El Equipo de Desarrollo consiste en los profesionales que desempeñan el trabajo de entregar un Incremento de producto “Terminado”, que potencialmente se

pueda poner en producción, al final de cada Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento. Los Equipos de Desarrollo son estructurados y empoderados por la organización para organizar y gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y efectividad del Equipo de Desarrollo. (Schwaber & Sutherland, 2013)

###### PROCESOS DE SCRUM

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio.

* + - * **PILA DE PRODUCTO (PRODUCT BACKLOG):** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
      * **PLANIFICACIÓN DEL SPRINT (SPRINT PLANNING):** Reunión durante la cual el Product Owner presenta las historias del Backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
      * **SPRINT:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
      * **PILA DEL SPRINT (SPRINT BACKLOG):** Lista de las tareas necesarias para llevar a cabo las historias del sprint.
      * **REUNIÓN DE SPRINT DIARIO (DAILY SPRINT MEETING):** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
      * **DEMOSTRACIÓN Y RETROSPECTIVA (DEMO Y RETROSPECTIVA):** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.
  1. **COMERCIO ELECTRONICO (*E-COMMERCE)***

Comercio electrónico es hacer negocios en línea. Esto es usando el poder de la información digital para entender las necesidades y preferencias de cada consumidor, la compañía puede personalizar productos y servicios y enviarlos tan pronto sea posible. Personalizar y automatizar servicios ofrece a los negocios el potencial de incrementar ingresos, bajar costos, y establecer fuertes vínculos entre los consumidores y la empresa. (Turban, 1995)

Para lograr estos beneficios muchas compañías actualmente ingresan al comercio electrónico para: promocionar, vender, ofrecer servicios directos, realizar

transacciones y pagos, distribuir información en forma segura, tener cadenas de valor comercial y realizar compras corporativas.

Existen diferentes modalidades para el comercio electrónico:

* B2C "Business to Consumer": venta de productos finales a un consumidor (Tiendas virtuales como Amazon.com).
* B2B "Business to Business": comercio entre empresas, cliente-proveedor. (Mercados para empresas como opciona.com).
* C2C "Consumer to Consumer": subastas en las que usuarios particulares venden productos (Subastas como Ebay.com).
* C2B "Consumer to Business": consumidores particulares se agrupan para tener más fuerza y hacer pedidos a empresas (Cooperativas como letsbuyit.com).

Gobierno electrónico o e-Goverment A2B, A2C, A2A (Administration to Business, Consumer, Administration), G2B/C/G "Government to Business/Consumer o Government": relaciones con las administraciones públicas y los ciudadanos, empresas u otras administraciones. (Impuestos vía Internet como Aeat.es).

* P2P "Peer to Peer": de amigo a amigo, como el intercambio de música P2P o los préstamos personales Prosper (Prosper.com).
* B2E, "Business to Employee": comunicación entre empresa y trabajador, (teletrabajo).
  + 1. **NEGOCIO A CONSUMIDOR B2C (*BUSINESS TO CONSUMER*)**

Este tipo de comercio se produce entre una empresa y un consumidor final que consume el bien o servicio, esto es, no lo transforma en algo distinto para venderlo o traspasarlo a otro actor del ciclo de negocios.

El B2C trata la venta de los productos y/o servicios de la empresa a consumidores de a pie (no empresariales). Este tipo de e-Commerce mejora el servicio prestado a los clientes y genera oportunidades para elevar ventas e ingresos.

La mayoría de los usuarios que compran en Internet buscan artículos a precios competitivos. A medida que aumenta el número de empresas que ofrecen productos en línea, los clientes disponen de mayor libertad para investigar y comparar precios. Con una solución B2C, la empresa puede ofrecer a sus clientes la comodidad de efectuar compras en Internet, además de atraer a nueva clientela.

###### VENTAJAS DE B2C

Para el cliente:

* + - * + Información precisa de estado de compras.
        + Comparación de precios ágil y rápida.
        + Posibilidad de consulta de información muy detallada. (Catálogos electrónicos, vídeos, foros, etc. De los productos/servicios).
        + Comprar a distancia, sin necesidad de desplazamiento.
        + Compras más económicas.

Para la empresa:

* + - * + Menor coste de infraestructura.
        + Amortización a corto plazo.
        + Mejoras en la gestión de stocks y disminución de niveles de inventario.
        + Mayor aprovechamiento del recurso humano en área de compras.
        + Expansión geográfica de mercado.
        + Mejoras en la gestión de compras y proceso de ventas

###### PASARELA DE PAGO

(Mantilla Garcia, 2015) Una pasarela de pago es un servicio de un proveedor de servicios de aplicación de comercio electrónico con el que se autorizan pagos a negocios electrónicos/online, ventas online al detalle, negocios con presencia física y online simultánea, o a negocios tradicionales. Es el equivalente de una TPV (Terminal Punto de Venta) física ubicada en la mayoría de los almacenes al detalle. Las pasarelas de pago cifran información sensible, tal como números de tarjetas de crédito, para garantizar que la información pasa en forma segura entre el cliente y el vendedor.

Las pasarelas de pago son piezas de software vitales para el comercio electrónico. Su evolución está siendo vital para el desarrollo de la actividad del comercio online, dado que han pasado de ser unas cuantas líneas de código que permitían pagar un producto en una tienda en Internet, a ser unas auténticas plataformas que permiten una gestión integral de los pagos.

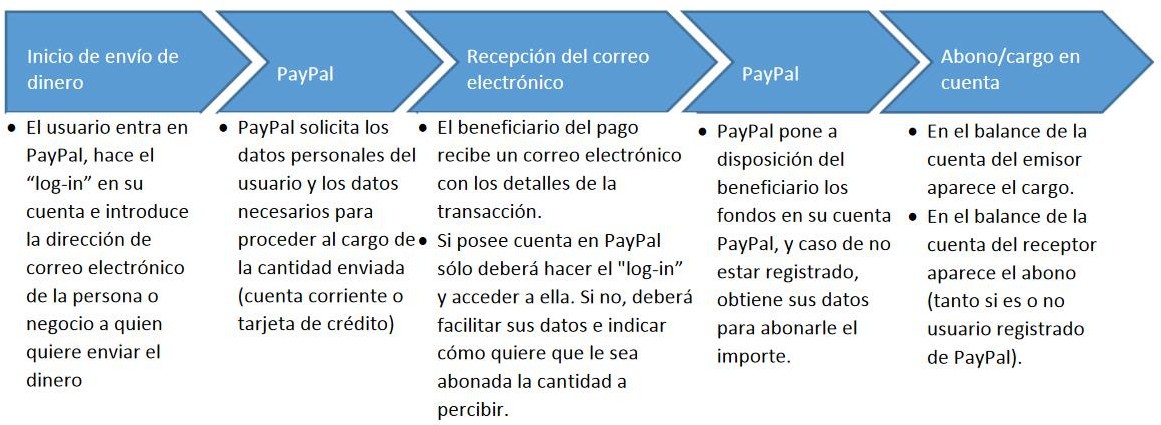
###### 1.4.2.1. PAYPAL

Paypal es una empresa global de comercio electrónico que permite realizar pagos y transferencia de dinero a través del internet entre individuos, pequeñas, medianas y grandes empresas. Cuando un cliente realiza un pago, Paypal procesa el pago y transfiere los fondos a la cuenta del vendedor. Paypal también ofrece

servicios de carrito de compras y un sistema de checkout. Estos servicios se pueden incluir con códigos prefabricados, adheriéndoles en las páginas web.

PayPal está basado en un sistema de cuentas de correo electrónico, en el que el usuario se registra previamente. PayPal permite enviar los pagos de forma segura a otra persona o empresa que disponga de una dirección de correo electrónico, utilizando para ello su tarjeta de crédito o cuenta corriente. Esta empresa permite realizar transacciones a través de una página web o un teléfono móvil, operando a nivel mundial. (Mantilla Garcia, 2015)

La principal diferencia frente a otros medios de pago es que PayPal no se centra en los comercios exclusivamente, sino que ofrece intercambios monetarios entre personas y permite que los datos bancarios o de tarjeta de crédito nunca se muestren al realizar una operación.



***FIGURA 5.*** *PROCESO DE ENVIO Y RECEPCIÓN DE DINERO A TRAVÉS DE PAYPAL*

***FUENTE.*** (Mantilla Garcia, 2015)

###### LENGUAJE UNIFICADO DE MODELADO (UML)

El Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual de propósito general que es usado para especificar, visualizar, construir y documentar los artefactos de un sistema software (Booch, Rambaugh, & Jacobson, 2004).Un sistema es modelado como una colección de objetos discretos que interactúan para desempeñar un trabajo que en última instancia beneficia a un usuario externo.

Los tres autores fijaron cuatro objetivos:

* Representar sistemas completos (más allá de un solo programa) por conceptos de objetos.
* Establecer una relación explícita entre los conceptos y los artefactos ejecutables.
* ejecutables. Tener en cuenta los factores de escala inherentes a los sistemas complejos y críticos.
* Crear un lenguaje de modelado utilizable tanto por los humanos como por las máquinas.

Para comprender mejor esta metodología, se debe crear un modelo conceptual del lenguaje, de los cuales se requiere tres elementos especiales: Construcción de bloques básicos, las reglas para que los bloques se puedan interrelacionar, y algunos mecanismos comunes que se aplican en todo el lenguaje.

###### MODELOS

Un modelo es la representación en un cierto medio de algo en el mismo u otro medio. El modelo captura los aspectos importantes del ente que será modelado desde un cierto punto de vista, simplificando u omitiendo el resto.

* + - * **MODELO DE CASOS DE USO:** El modelo de casos de uso permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requisitos. Contiene actores, casos de uso y sus relaciones.
      * **MODELO DE ANÁLISIS:** El modelo de análisis se representa mediante un análisis que denota el paquete de más alto nivel del modelo. La utilización de otros paquetes de análisis es por tanto una forma de organizar el modelo de análisis en partes más manejables que representan abstracciones de subsistemas y posiblemente capas completas del diseño del sistema.
      * **MODELO DE DISEÑO:** El modelo de diseño es un modelo de objetos que describe la realización física de los casos de usos centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como una entrada fundamental de las actividades de implementación. (Garcia, 2008)
      * **MODELO DE DESPLIEGUE:** El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos del cómputo. El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño

e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. (Díaz, 2013)

* + - * **MODELO DE IMPLEMENTACIÓN:** El modelo de implementación describe cómo los elementos del modelo de diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables, entre otros. El modelo de implementación describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles. (Espinoza, 2007)
      * **MODELO DE PRUEBA:** El modelo de prueba describe principalmente cómo se prueban los componentes ejecutables (como las construcciones) en el modelo de implementación con pruebas de integración y de sistema. El modelo de pruebas puede describir también cómo han de ser probados aspectos específicos del sistema.

###### DIAGRAMAS

De acuerdo a la Real Academia de la Lengua Española un diagrama es: un Dibujo geométrico que sirve para demostrar una proposición, resolver un problema o representar de una manera gráfica la ley de variación de un fenómeno. En el que se muestran las relaciones entre las diferentes partes de un conjunto o sistema.

###### DIAGRAMA DE CASOS DE USOS

Un diagrama de caso de uso es un diagrama que muestra un conjunto de casos de uso, actores y sus relaciones. Los mismos muestran el comportamiento del sistema a partir de los usuarios que interactúan con el sistema. Un diagrama de casos de uso representa las interacciones entre el sistema y los sistemas externos

y los usuarios. En otras palabras, describe gráficamente quien utiliza el sistema y la forma en que los usuarios esperan interaccionar con este. (Fernandez Alarcon, 2006)

A continuación se explica los elementos del diagrama de casos de usos con detalle:

* + - * + **CASOS DE USOS:** Un caso de uso representa un requisito funcional de un sistema. Los casos de uso son descriptores de las interacciones típicas entre los usuarios de un sistema y ese mismo sistema. Representan el interfaz externo del sistema y especifican qué requisitos de funcionamiento debe tener este sistema.
        + **ACTOR (ES):** Un actor es la representación de un conjunto coherente de roles que los usuarios de los casos de uso juegan cuando interactúan con estos. Por lo general, representan el papel desempeñado por una persona, un dispositivo, un objeto e incluso otro sistema que interactúa con el sistema propuesto.

Un actor puede ser representado con un rectángulo con el estereotipo actor. El estereotipo estándar es un icono representado por una persona dibujada con líneas. Existen varias relaciones estándares entre casos de uso o entre actores y casos de uso: asociación, generalización y uso.

* + - * + **RELACIÓN DE DEPENDENCIA:** Es una conexión de uso que indica que cualquier cambio en un elemento puede afectar a otro elemento que la utiliza, pero no necesariamente de modo inverso. Esta es representada por una flecha, de línea no continua, orientada hacia el elemento del que se depende. (Afonso & Segnini, 2009)
        + **RELACIÓN DE GENERALIZACIÓN:** Es la relación que existe entre un elemento general y un caso específico de ese mismo elemento. La generalización significa que los objetos hijos se pueden emplear en cualquier lugar donde pueda aparecer el padre, más no a la inversa. Esta relación se representa por una flecha continua con punta vacía orientada hacia el padre. (Afonso & Segnini, 2009)
        + **RELACIÓN DE ASOCIACIÓN*:*** La relación entre un actor y un caso de uso representa la interacción entre ellos. Este tipo de relación se denomina asociación y se representa gráficamente a través de una línea solida entre un actor y un caso de uso. Para diferenciar al actor que inicia un caso de uso del resto de actores que se ven involucrados en él, su línea asociativa acaba con la flecha en el caso de uso. Para el resto de actores que intervienen, pero que no lo han iniciado, la asociación se representa con un alinea sin flecha. (Fernandez Alarcon, 2006)

En la siguiente figura se muestra los componentes del diagrama de casos de uso:

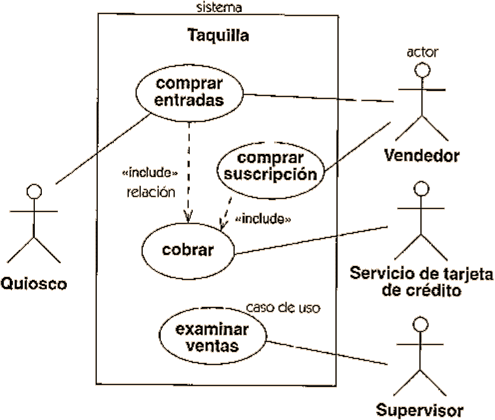


***FIGURA 6.*** *COMPONENTES DEL DIAGRAMA DE CASOS DE USO*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)

De la misma manera se detalla un ejemplo de un diagrama de casos uso a

continuación:



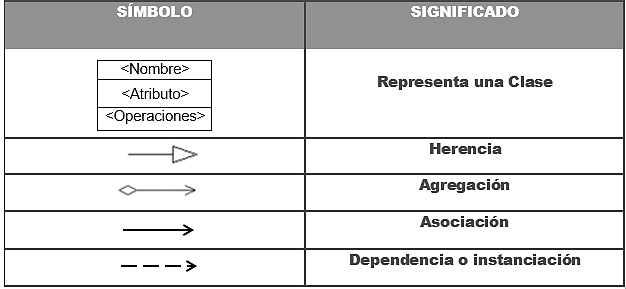
***FIGURA 7.*** *DIAGRAMA DE CASOS DE USO*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)

###### DIAGRAMAS DE CLASES

El diagrama de clases sirve para representar la visión estática del sistema. Se puede representar con varios niveles de detalle, no siendo necesario representarlo siempre con el máximo nivel. Contiene la estructura de clases (con sus métodos y atributos) y las relaciones entre ellas, con sus propiedades (cardinalidad, roles, herencia y pertenencia). Existen diferentes maneras de representar una clase, dependiendo de su nivel de detalle. (Barranco de Areba, 2001)

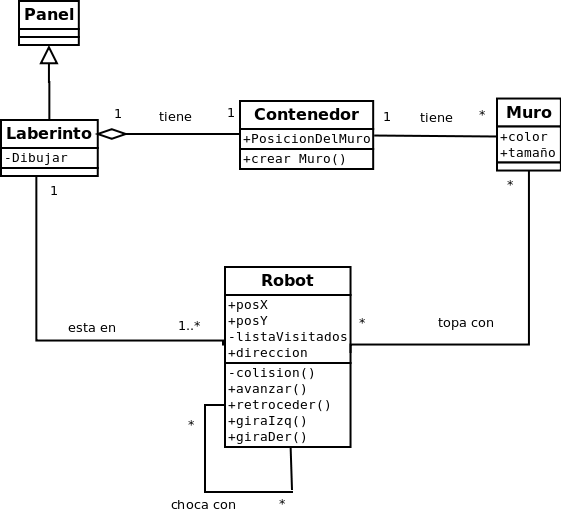
A continuación se detalla la simbología del diagrama de clases:



***FIGURA 8.*** *SIMBOLOGÍA DEL DIAGRAMA DE CLASES*

***FUENTE****. Elaboración Propia*

Del mismo modo para que se pueda entender de mejor manera el diagrama de clases tenemos la siguiente figura:



***FIGURA 9.*** *DIAGRAMA DE CLASES*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)

Los atributos de la clase pueden representarse dentro del símbolo de la clase. El nombre de cada atributo puede ir seguido por detalles opcionales, tales como el tipo y el valor por omisión. El tipo ira precedido por dos puntos, y el valor por omisión por un signo igual. (Barranco de Areba, 2001)

Por lo general, un diagrama de clases está compuesto por: clases, interfaces, colaboraciones y relaciones de dependencia, generalización y asociación.

* **CLASE:** Una clase describe un conjunto de objetos con estructura similar, su comportamiento y sus relaciones. Representa un concepto dentro del sistema a ser modelado; las clases tienen estructura de datos, comportamiento y relaciones con otros elementos. Una clase es dibujada como un rectángulo sólido con tres compartimentos separados por líneas horizontales. El primer compartimiento contiene el nombre de la clase y otras propiedades generales de la clase. El segundo compartimiento contiene los atributos de la clase. El tercer compartimiento contiene la lista de operaciones de la clase.
* **INTERFACE:** Una interface utiliza un tipo para describir el comportamiento visible de una clase, de un componente o de un paquete. Una interface es un estereotipo de un tipo. (Muller, 1997) Pueden ser mostradas con un rectángulo que se identifica con la palabra clave “interface” y que además tiene compartimentos. La lista de operaciones soportadas por la interface se coloca en el compartimiento de operaciones.
* **RELACIÓN DE DEPENDENCIA:** Una relación de dependencia se establece entre clases cuando un cambio en el elemento independiente del modelo puede requerir un cambio en el elemento dependiente.
* **RELACIÓN DE ASOCIACIÓN:** Las asociaciones representan relaciones estructurales entre clases y objetos. Una asociación simboliza una información cuya duración de vida no es relevante respecto a la dinámica

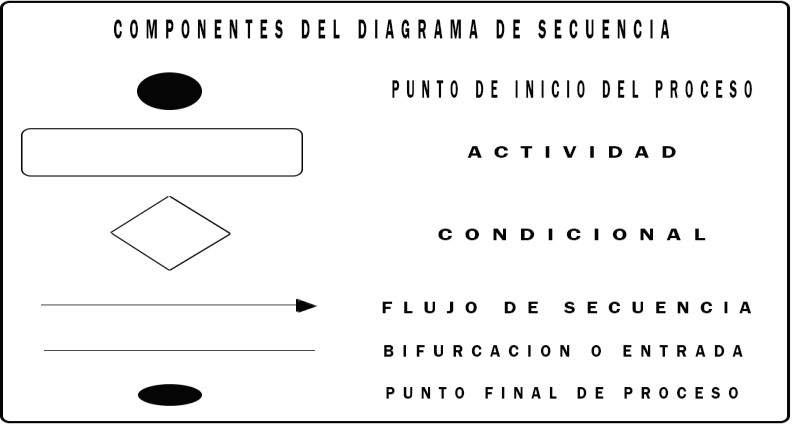
general de los objetos instancias de las clases asociadas. (Booch, Rambaugh, & Jacobson, 2004)

* **RELACIÓN DE GENERALIZACIÓN:** La generalización entre dos clases consiste en considerar a una clase como superclase y a la otra como la subclase. Puede haber más de una clase que se comporte como subclase.

###### DIAGRAMAS DE SECUENCIA

(Booch, Rambaugh, & Jacobson, 2004) Exponen que: Un diagrama de secuencia muestra una interacción como un gráfico bidimensional. La dimensión vertical es el eje del tiempo; El tiempo pasa por la página. La dimensión horizontal muestra los roles que representan objetos individuales en la colaboración. Cada rol está representado por una columna vertical que contiene un símbolo de cabeza y una línea vertical, una cuerda de salvamento.

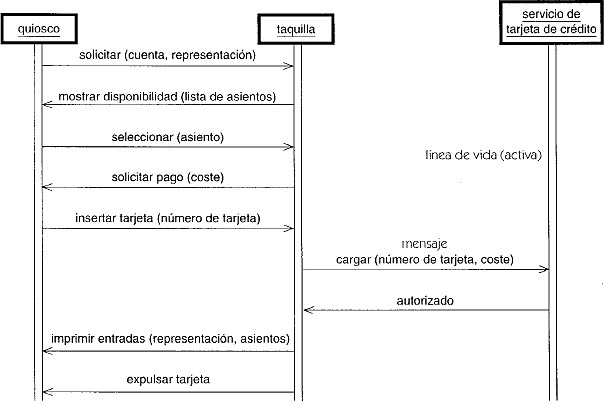
A continuación se detalla los componentes del diagrama de secuencia:



***FIGURA 10.*** *COMPONENTES DEL DIAGRAMA DE SECUENCIA*

***FUENTE.*** *ELABORACIÓN PROPIA*

Para mejor comprensión un ejemplo de diagrama de secuencia:



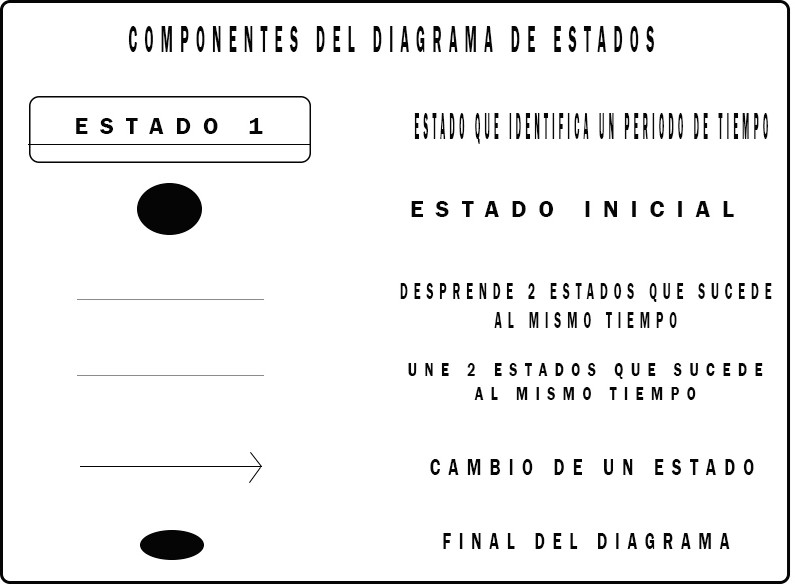
***FIGURA 11.*** *DIAGRAMA DE SECUENCIA*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)*.*

###### DIAGRAMA DE ESTADOS

Los diagramas de estados se usan para representar gráficamente máquinas de estados finitos. Una máquina de estados es un comportamiento que especifica las secuencias de estados por las que pasa un objeto a lo largo de su vida en respuesta a eventos, junto con sus respuestas a esos eventos. (Booch, Rambaugh, & Jacobson, 2004)

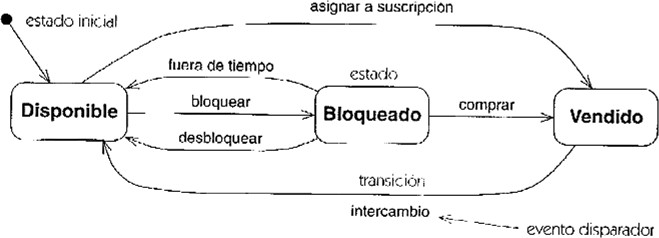
Los elementos básicos de notación que pueden usarse para componer un diagrama son:

* + - * + Círculo lleno, apuntando a un estado inicial.
        + Círculo hueco que contiene un círculo lleno más pequeño en el interior, indicando el estado final (si existiera).
        + Rectángulo redondeado, denotando un estado. En la parte superior del rectángulo está el nombre del estado.
        + Flecha, denotando transición. El nombre del evento (si existiera) que causa esta transición etiqueta el cuerpo de la flecha.

***FIGURA 12.*** *COMPONENTES DEL DIAGRAMA DE ESTADOS*

***FUENTE.*** *ELABORACION PROPIA*

A continuación se muestra un ejemplo del diagrama de estados:



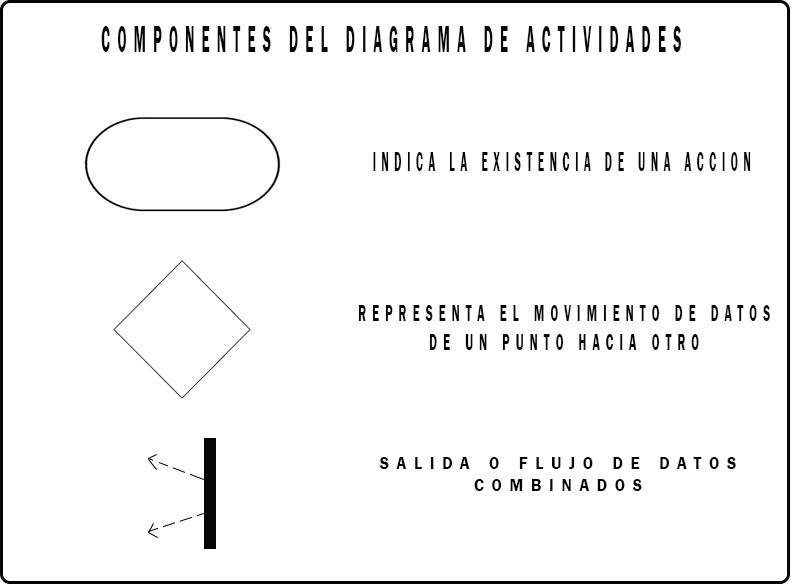
***FIGURA 13.*** *DIAGRAMA DE ESTADOS*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)

###### DIAGRAMA DE ACTIVIDADES

Según (Fowler, 1999) un diagrama de actividades es una técnica que describe un procedimiento lógico, proceso de negocio o un flujo de trabajo. La interpretación de este tipo de diagramas depende de su ubicación en el proyecto. En una fase conceptual las actividades pueden ser vistas como tareas que deben

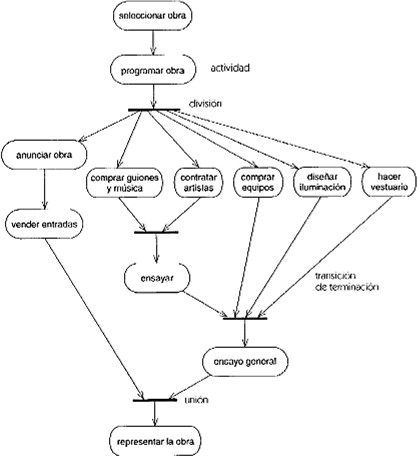
llevar a cabo por una computadora o un ser humano. Para una mejor comprensión se detalla a continuación los componentes del diagrama de actividades:



***FIGURA 14.*** *COMPONENTES DEL DIAGRAMA DE ACTIVIDADES*

***FUENTE.*** *ELABORACION PROPIA*

Una vez entendido los componentes del diagrama de actividades se representa mediante un ejemplo del diagrama de actividades*:*



***FIGURA 15.*** *DIAGRAMA DE ACTIVIDADES*

***FUENTE.*** (Booch, Rambaugh, & Jacobson, 2004)

###### TECNICA DE MODELADO DE OBJETOS (OMT)

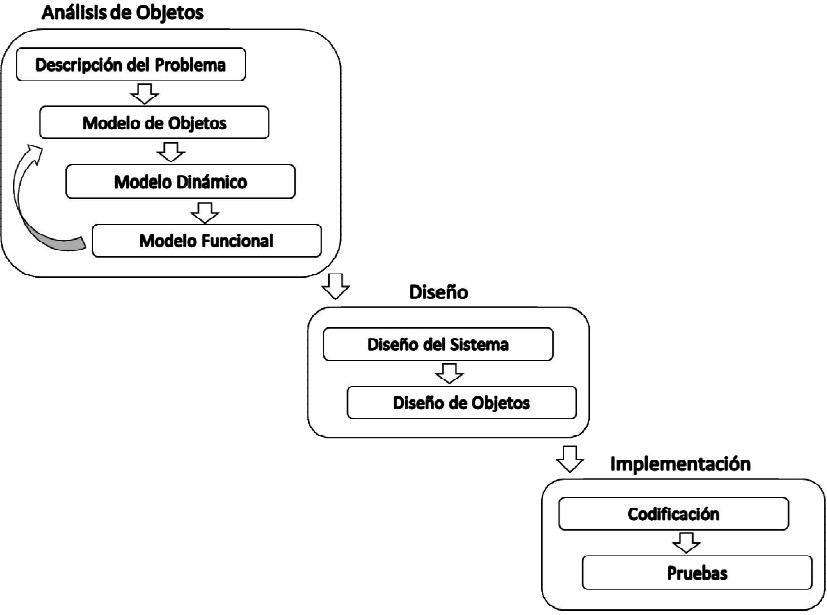
OMT (*Object Modeling Technique*) fue creado por James Rumbaugh y Michael Blaha en 1991, mientras James dirigía un equipo de investigación de los laboratorios General Electric. (Medina, 2005)

La esencia del análisis y diseño orientado a objetos es la identificación y organización de conceptos del dominio de la aplicación, y no de su presentación final en un lenguaje de programación, es decir, es un proceso conceptual independiente de sí el lenguaje es orientado a objetos. (Rumbaugh & Blaha, 1991)

Para construir sistemas complejos, el desarrollador debe abstraer distintas vistas del sistema, construir modelos utilizando notaciones precisas, verificar que los modelos satisfacen los requisitos del sistema y añadir, gradualmente, detalles para trasformar los modelos en una implementación. (Rumbaugh & Blaha, 1991)

OMT divide el ciclo de vida del software en cuatro fases consecutivas:

1. **ANALISIS DE OBJETOS:** Se centra en entender y modelar el problema en el dominio de la aplicación.
2. **DISEÑO DEL SISTEMA:** Se determina la arquitectura del sistema en términos de subsistemas.
3. **DISEÑO DE OBJETOS:** Se refina y optimiza el análisis de objetos para implementarlo.
4. **IMPLEMENTACION:** Se codifica y prueba lo ya diseñado.



***FIGURA 16****. CICLO DE VIDA OMT*

***FUENTE.*** *(Rumbaugh & Blaha, 1991)*

###### ANALISIS DE OBJETOS

En primer lugar se describe el problema. Se obtienen los requisitos que no den lugar a dudas. En segundo lugar se hacen los diagramas de objetos. En él se define la estructura de los objetos y clases así como las relaciones que los unen.

Posteriormente se crea un modelo dinámico para describir los aspectos de control y evolución del sistema. Comprende de un diagrama de estado por cada clase que tenga un comportamiento dinámico, y el diagrama de global de flujo de sucesos (Diagrama de actividades).Después se crea un modelo funcional que describa las funciones, los valores de entrada y salida. Se suelen usar los diagramas de flujo de datos orientados a objetos.

Por último se verifican todos los modelos creados y se itera para conseguir un refinamiento de los tres.

###### EL MODELO DE OBJETOS

El modelo de objetos es el modelo más importante, ya que en él se identifican las clases dentro del sistema junto con sus relaciones, así como sus atributos y operaciones, lo que representa la estructura estática del sistema.

###### EL MODELO DINAMICO

Representa los aspectos temporales de comportamiento “control” del sistema, mediante la secuencia de operaciones en el tiempo.

###### EL MODELO FUNCIONAL

Representa los aspectos transformacionales “de función” del sistema mediante la transformación de valores de los datos. Se representa mediante un diagrama de flujo.

###### MODELO DE OBJETOS

Los pasos para construir el modelo de objetos son los siguientes:

1. Identificación de objetos y/o clases
2. Crear un diccionario de datos
3. Identificación de atributos y enlaces para el realizar el diagrama de clases u objetos
4. Realizar las iteraciones necesarias para el refinamiento del modelo

***Modelo de objetos = Diagrama de modelo de objetos + Diccionario de datos***

###### MODELO DINAMICO

Los pasos para construir el modelo dinámico son los siguientes:

1. Preparación de escenarios de secuencias típicas de iteración.
2. Identificación de sucesos que actúan entre objetos.
3. Construcción de un diagrama de estado para cada objeto.
4. Construcción del diagrama global de flujo de sucesos (Diagrama de secuencia)

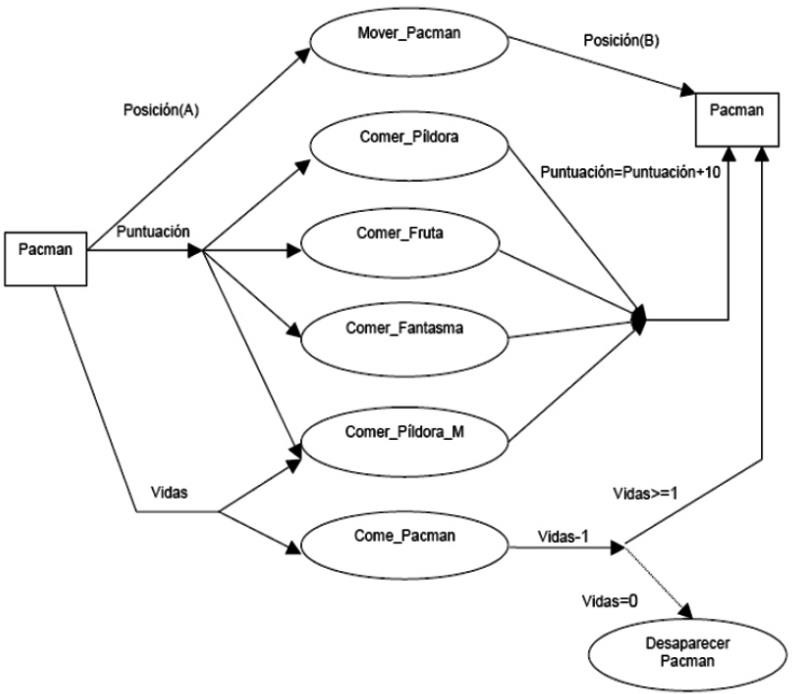
***Modelo dinámico = Diagrama global de flujo de suceso + Diagrama de estados***

###### MODELO FUNCIONAL

Los pasos para construir el modelo funcional son los siguientes:

1. Identificación de los valores de entrada y de salida.
2. Construcción de diagramas de flujo de datos que muestren las dependencias funcionales.
3. Descripción de las funciones.
4. Identificación de restricciones

***Modelo Funcional = Diagrama de flujo de datos + restricciones***



***FIGURA 17****. DIAGRAMA DE FLUJO DE DATOS – OMT*

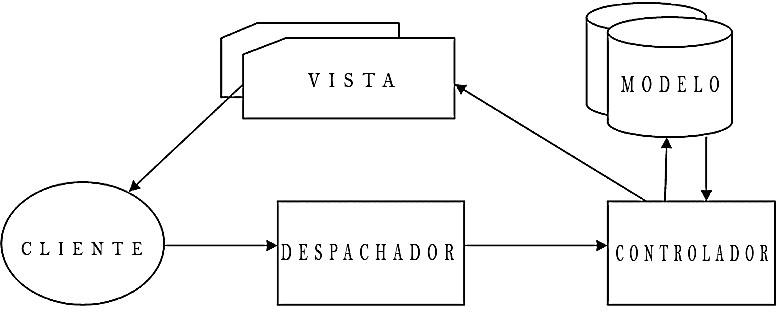
***FUENTE****. (Medina, 2005)*

###### ARQUITECTURA DEL SOFTWARE

(Clements, 1996) Define que: La Arquitectura del Software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo,

la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema. La vista arquitectónica es una vista abstracta, aportando el más alto nivel de comprensión y la supresión o diferimiento del detalle inherente a la mayor parte de las abstracciones.

###### PATRON MODELO VISTA CONTROLADOR

El modelo vista controlador (MVC) es un patrón de arquitectura de software que separa los datos y la aplicación de la interfaz de usuario y el modulo encargado de gestionar los eventos y las comunicaciones. Para ello, MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador. En este tipo de aplicaciones, la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema Gestor de Base de Fatos y la lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista. (Talledo, 2015)

***FIGURA 18.*** *MODELO VISTA CONTROLADOR*

***FUENTE.*** (Talledo, 2015)

###### EL MODELO

El modelo es responsable de:

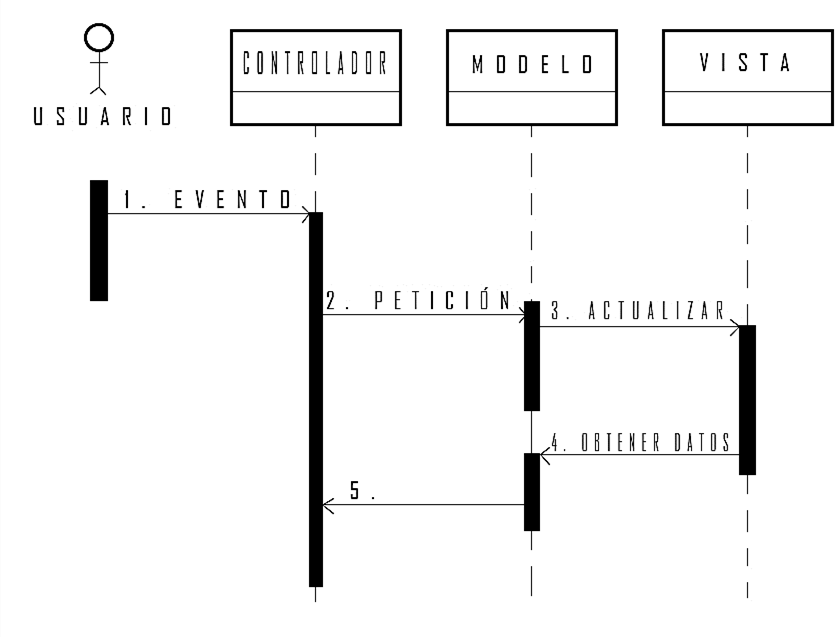
* + - * + Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
        + Define reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor". Es opcional, pues las reglas de negocio, pueden estar también en los controladores, directamente en las acciones.
        + Notificará a las vistas los cambios que en los datos pueda producir un agente externo si se está ante un modelo activo (por ejemplo, un fichero bath que actualiza los datos, un temporizador que desencadena una inserción, etc.). (Fernández Romero & Díaz Gonzáles, 2012)

###### LA VISTA

Las vistas son responsables de:

* + - * + Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.
        + Tienen un registro de su controlador asociado.
        + Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes.

Un ejemplo de MVC con un modelo pasivo (aquel que no notifica cambios en los datos) es la navegación web, que responde a las entradas del usuario, pero no

detecta los cambios en datos del servidor. El Diagrama de Secuencia que se muestra en la FIGURA 15 ilustra la interrelación de los elementos del patrón. (Fernández Romero & Díaz Gonzáles, 2012)

***FIGURA 19.*** *MODELO VISTA CONTROLADOR- DIAGRAMA DE SECUENCIA*

***FUENTE.*** (Fernández Romero & Díaz Gonzáles, 2012)

###### EL CONTROLADOR

El controlador es responsable de:

* Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
* Contiene reglas de gestión de eventos, del tipo "Si Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar ()". Una petición al modelo puede ser "Obtener\_tiempo\_de\_ entrega (nueva\_orden\_de\_venta)". (Fernández Romero & Díaz Gonzáles, 2012)

###### ARQUITECTURA DE 3 CAPAS

El Patrón de arquitectura por capas es una de las técnicas más comunes que los arquitectos de software utilizan para dividir sistemas de software complicados. Al pensar en un sistema en términos de capas, se imaginan los principales

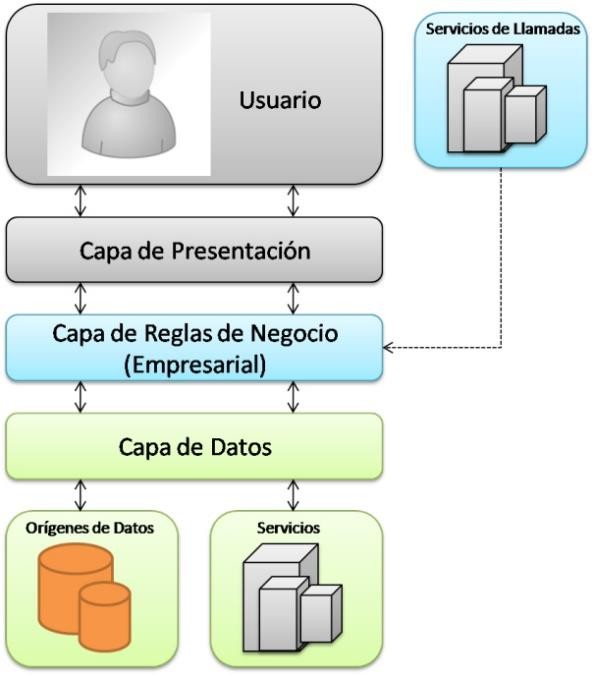
subsistemas de software ubicados de la misma forma que las capas de un pastel, donde cada capa descansa sobre la inferior.

Los beneficios de trabajar un sistema en capas son:

* Se puede entender una capa como un todo, sin considerar las otras.

Las capas se pueden sustituir con implementaciones alternativas de los mismos servicios básicos.

* Se minimizan dependencias entre capas.
* Las capas posibilitan la estandarización de servicios.
* Luego de tener una capa construida, puede ser utilizada por muchos servicios de mayor nivel.

La imagen que se muestra a continuación presenta el esquema de una arquitectura siguiendo este patrón:

***FIGURA 20.*** *ARQUITECTURA 3 CAPAS*

***FUENTE.*** (Sommerville, 2005)

###### CAPA DE PRESENTACIÓN

Referente a la interacción entre el usuario y el software. Puede ser tan simple como un menú basado en líneas de comando o tan complejo como una aplicación basada en formas. Su principal responsabilidad es mostrar información al usuario, interpretar los comandos de este y realizar algunas validaciones simples de los datos ingresados.

###### CAPA DE REGLAS DE NEGOCIO (EMPRESARIAL)

También denominada Lógica de Dominio, esta capa contiene la funcionalidad que implementa la aplicación. Involucra cálculos basados en la información dada por el usuario y datos almacenados y validaciones. Controla la ejecución de la capa de acceso a datos y servicios externos. Se puede diseñar la lógica de la capa de negocios para uso directo por parte de componentes de presentación o su encapsulamiento como servicio y llamada a través de una interfaz de servicios que coordina la conversación con los clientes del servicio o invoca cualquier flujo o componente de negocio. (Sommerville, 2005)

###### CAPA DE DATOS

Esta capa contiene la lógica de comunicación con otros sistemas que llevan a cabo tareas por la aplicación. Estos pueden ser monitores transaccionales, otras aplicaciones, sistemas de mensajerías, etc. Para el caso de aplicaciones empresariales, generalmente está representado por una base de datos, que es responsable por el almacenamiento persistente de información. Esta capa debe abstraer completamente a las capas superiores (negocio) del dialecto utilizado para comunicarse con los repositorios de datos (PL/SQL, Transact-SQL, etc.).

###### SEGURIDAD

La seguridad informática es la disciplina que se ocupa de diseñar las normas, procedimientos, métodos y técnicas destinados a conseguir un sistema de información seguro y confiable. Un sistema de información, no obstante las medidas de seguridad que se le apliquen no deja de tener siempre un margen de riesgo. (Aguilera López, 2010)

###### PROPIEDADES DE UN SISTEMA DE INFORMACION SEGURO

###### INTEGRIDAD

Este principio garantiza la autenticidad y precisión de la información sin importar el momento en que esta se solicita, o dicho de otra manera, una garantía de que los datos no han sido alterados ni destruidos de modo no autorizado. Para evitar este tipo de riesgos se debe dotar al sistema de mecanismos que prevengan y detecten cuándo se produce un fallo de integridad y que puedan tratar y resolver los errores que se han descubierto. (Gárcia-Cervigón Hurtado & Alegre Ramos, 2011)

###### CONFIDENCIALIDAD

La OCDE (Organización para la Cooperación y el Desarrollo Económico), en sus Directrices para la Seguridad de los Sistemas de Información define la confidencialidad como: el hecho de que los datos o informaciones estén únicamente al alcance del conocimiento de las personas, entidades o mecanismos autorizados, en los momentos autorizados y de una manera autorizada. Para prevenir errores de confidencialidad debe diseñarse un control de accesos al sistema: quién puede acceder, a qué parte del sistema, en qué m omento y para realizar qué tipo de operaciones. (Gárcia-Cervigón Hurtado & Alegre Ramos, 2011)

###### DISPONIBILIDAD

La información ha de estar disponible para los usuarios autorizados cuando la necesiten. Situación que se produce cuando se puede acceder a un sistema de información en un periodo de tiempo considerado aceptable. La disponibilidad está asociada a la fiabilidad técnica de los componentes del sistema de información. Se deben aplicar medidas que protejan la información, así como crear copias de seguridad y mecanismos para restaurar los datos que accidental o intencionadamente se hubiesen dañado o destruido.

###### MÉTODOS

###### CORTAFUEGOS (FIREWALL)

Un cortafuegos o firewall es la parte de un sistema informático que se utiliza para impedir un acceso a través de la red no autorizado, pero permitiendo sin problemas todos los accesos autorizados. Los cortafuegos pueden ser software o hardware. (Gárcia-Cervigón Hurtado & Alegre Ramos, 2011)

###### AUTENTICACIÓN Y AUTORIZACIÓN

La autenticación y la autorización están estrechamente relacionadas debido a que la autenticación es el establecimiento de confirmación para dar paso a la autorización de acceso a las zonas restringidas, realizar acciones permitidas con su correspondiente nivel de privilegio, controlar el acceso a recursos protegidos, para prevenir ataques de escalada de privilegios.

###### CIFRADO DE DATOS

Constituye uno de los métodos de protección más fiable. Consiste en la transformación de los datos, de forma que una persona que no deba tener acceso a ellos no sea capaz de entenderlos. El cifrado de datos se puede realizar a través

de elementos lógicos o físicos, pero en cualquiera de los casos es un proceso que consume bastantes recursos.

###### SESIÓN

Una sesión es una serie de comunicaciones entre un cliente y un servidor en la que se realiza un intercambio de información. Por medio de una sesión se puede hacer un seguimiento de un usuario a través de la aplicación. El tiempo de vida de una sesión comienza cuando un usuario se conecta por primera vez a un sitio web pero su finalización puede estar relacionada con tres circunstancias.

###### MODELO DE ESTIMACION DE COSTOS COCOMO

El Modelo Constructivo de Costos (Constructive Cost Model) fue desarrollado por B.W. Boehm a finales de los 70 y comienzos de los 80, exponiéndolo detalladamente en su libro “Software Engineering Economics” (Prentice-Hall, 1981). COCOMO es una jerarquía de modelos de estimación de costes software que incluye submodelos básico, intermedio y detallado. Para este proyecto utilizaremos el modelo básico.

Cocomo pertenece a la categoría de modelos de subestimaciones basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el “tamaño” del proyecto, en líneas de código principalmente. Las ecuaciones que se utilizan en los tres modelos son:

𝑬 = 𝒂(𝑲𝑳𝑫𝑪)𝒃 ∗ 𝑭𝑨𝑬, 𝒆𝒏 𝒑𝒆𝒓𝒔𝒐𝒏𝒔𝒂 − 𝒎𝒆𝒔

𝑻𝒅𝒆𝒗 = 𝒄(𝑬)𝒅, 𝒆𝒏 𝒎𝒆𝒔𝒆𝒔

𝑷 =

𝑬

𝑻𝒅𝒆𝒗

, 𝒆𝒏 𝒑𝒆𝒓𝒔𝒐𝒏𝒂𝒔

Donde:

* E = Es el esfuerzo requerido por el proyecto, en persona-mes
* Tdev = Es el tiempo requerido por el proyecto, en meses
* P = Es el número de personas requerido por el proyecto
* a,b,c,d = Son constantes con valores definidos en una tabla, según cada submodelo.
* KLDC = Es la cantidad de líneas de código, en miles
* FAE = Es un multiplicador que depende de 15 atributos

A la vez, cada submodelo también se divide en modos que representan el tipo de proyecto, y puede ser:

###### MODELO ORGANICO

Un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles (medio).

###### MODELO SEMILIBRE O SEMIENCAJADO

Corresponde a un esquema intermedio entre el orgánico y el rígido; el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.

###### MODO RIGIDO O EMPOTRADO

El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia, puesto que puede no haberla.

###### MODELO BASICO

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODO** | **A** | **B** | **C** | **D** |
| **ORGANICO** | 2,40 | 1,05 | 2,50 | 0,38 |
| **SEMI-**  **ENCAJADO** | 3,00 | 1,12 | 2,50 | 0,35 |
| **EMPOTRADO** | 3,60 | 1,20 | 2,50 | 0,32 |

***FIGURA 21.*** *TABLA DE CONSTANTES PARA CALCULAR DIFERENTES ASPECTOS DE COSTES*

***FUENTE.*** (Afonso & Segnini, 2009)

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2,4 a 3,6 que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno.

###### MODELO INTERMEDIO

Este añade al modelo básico quince modificadores opcionales para tener en cuenta en el entorno de trabajo, incrementando así la precisión de la estimación.

Para este ajuste, al resultado de la formula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Dentro del modelo intermedio se utiliza la siguiente formula:

𝑬 = 𝒂(𝑲𝑳𝑫𝑪)𝒃 ∗ 𝑭𝑨𝑬

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño; mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio alrededor del semi-libre con respecto al efecto multiplicador de los atributos de coste.

El valor de cada atributo, de acuerdo a su calificación, se muestra en la siguiente tabla:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Valor** | | | | | | |
| **Atributos** | **Muy**  **bajo** | **Bajo** | **Nomin**  **al** | **Alto** | **Muy**  **Alto** | **Extra**  **Alto** |
| **Atributos de software** | | | | | | |
| **Fiabilidad** | 0,75 | 0,88 | 1,00 | 1,15 | 1,40 |  |
| **Tamaño de base de datos** |  | 0,94 | 1,00 | 1,08 | 1,16 |  |
| **Complejidad** | 0,70 | 0,85 | 1,00 | 1,15 | 1,30 | 1,65 |
| **Atributos de hardware** | | | | | | |
| **Restricciones de tiempo de**  **ejecución** |  |  | 1,00 | 1,11 | 1,30 | 1,66 |
| **Restricciones de memoria virtual** |  |  | 1,00 | 1,06 | 1,21 | 1,56 |
| **Volatilidad de la máquina virtual** |  | 0,87 | 1,00 | 1,15 | 1,30 |  |
| **Tiempo de respuesta** |  | 0,87 | 1,00 | 1,07 | 1,15 |  |
| **Atributos de personal** | | | | | | |
| **Capacidad de análisis** | 1,46 | 1,19 | 1,00 | 0,86 | 0,71 |  |
| **Experiencia en la aplicación** | 1,29 | 1,13 | 1,00 | 0,91 | 0,82 |  |
| **Calidad de los programadores** | 1,42 | 1,17 | 1,00 | 0,86 | 0,70 |  |
| **Experiencia en la máquina virtual** | 1,21 | 1,10 | 1,00 | 0,90 |  |  |
| **Experiencia en el lenguaje** | 1,14 | 1,07 | 1,00 | 0,95 |  |  |
| **Atributos del proyecto** | | | | | | |
| **Técnicas actualizadas de**  **programación** | 1,24 | 1,10 | 1,00 | 0,91 | 0,82 |  |
| **Utilización de herramientas de**  **software** | 1,24 | 1,10 | 1,00 | 0,91 | 0,83 |  |
| **Restricciones de tiempo de**  **desarrollo** | 1,22 | 1,08 | 1,00 | 1,04 | 1,10 |  |

***FIGURA 22.*** *TABLA DE VALORES MODO INTERMEDIO – COCOMO*

***FUENTE.*** (Afonso & Segnini, 2009)

###### MODELO DETALLADO

Presenta principalmente dos mejoras respecto al anterior:

* + - * Los factores correspondientes a los atributos son sensibles o dependientes de la fase sobre la que se realizan las estimaciones. Aspectos tales como la experiencia en la aplicación, utilización de herramientas de software, etc.,

tienen mayor influencia en unas fases que en otras y además van variando de una etapa a otra.

###### METRICAS DE CALIDAD DEL SOFTWARE

Los desarrolladores de software más hastiados estarán de acuerdo que el software de alta calidad es una de las metas más importantes. La calidad del software es una compleja mezcla de factores que varían a través de diferentes aplicaciones y según el cliente que las pida (Pressman, 2010).

###### EL ESTANDAR ISO/IEC 25010

La ISO (International Organization for Standarization) y la *IE* (International Electrotechnical Commission) forman el sistema especializado para la estandarización a nivel mundial. En el campo de tecnologías de información, ISO e IEC han establecido un comité técnico conjunto denominado ISO/IEC JTC, cuya tarea principal es preparar estándares internacionales ( Pérez Medina & Sánchez, 2012).

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las ocho características de calidad que se muestran en la siguiente figura:



***FIGURA 23.*** *CALIDAD DEL PRODUCTO SOFTWARE – ISO/IEC 25010*

***FUENTE.*** (ISO 25000, 2015)

###### ADECUACIÓN FUNCIONAL

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### COMPLETITUD FUNCIONAL

Grado en el cual el conjunto de funcionalidades cubre todas las tareas y los objetivos del usuario especificados.

###### CORRECCIÓN FUNCIONAL

Capacidad del producto o sistema para proveer resultados correctos con el nivel de precisión requerido.

###### PERTINENCIA FUNCIONAL

Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

###### EFICIENCIA DE DESEMPEÑO

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### COMPORTAMIENTO TEMPORAL

Los tiempos de respuesta y procesamiento y los ratios de throughput de un sistema cuando lleva a cabo sus funciones bajo condiciones determinadas en relación con un banco de pruebas (benchmark) establecido.

###### UTILIZACIÓN DE RECURSOS

Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.

###### CAPACIDAD

Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

###### COMPATIBILIDAD

(ISO 25000, 2015) Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### COEXISTENCIA

Capacidad del producto para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes sin detrimento.

###### INTEROPERABILIDAD

Capacidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada.

###### USABILIDAD

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### CAPACIDAD PARA RECONOCER SU ADECUACIÓN

Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.

###### CAPACIDAD DE APRENDIZAJE

Capacidad del producto que permite al usuario aprender su aplicación.

###### CAPACIDAD PARA SER USADO

Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.

###### PROTECCIÓN CONTRA ERRORES DE USUARIO

Capacidad del sistema para proteger a los usuarios de hacer errores.

###### ESTÉTICA DE LA INTERFAZ DE USUARIO

Capacidad de la interfaz de usuario de agradar y satisfacer la interacción con el usuario.

###### ACCESIBILIDAD

Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

###### FIABILIDAD

Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### MADUREZ

Capacidad del sistema para satisfacer las necesidades de fiabilidad en condiciones normales.

###### DISPONIBILIDAD

Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.

###### TOLERANCIA A FALLOS

Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.

###### CAPACIDAD DE RECUPERACIÓN

Capacidad del producto software para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de interrupción o fallo. (ISO 25000, 2015)

###### SEGURIDAD

Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### CONFIDENCIALIDAD

Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.

###### INTEGRIDAD

Capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.

###### NO REPUDIO

Capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.

###### RESPONSABILIDAD

Capacidad de rastrear de forma inequívoca las acciones de una entidad.

###### AUTENTICIDAD

Capacidad de demostrar la identidad de un sujeto o un recurso.

###### MANTENIBILIDAD

Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### MODULARIDAD

Capacidad de un sistema o programa de ordenador (compuesto de componentes discretos) que permite que un cambio en un componente tenga un impacto mínimo en los demás.

###### REUSABILIDAD

Capacidad de un activo que permite que sea utilizado en más de un sistema software o en la construcción de otros activos.

###### ANALIZABILIDAD

Facilidad con la que se puede evaluar el impacto de un determinado cambio sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.

###### CAPACIDAD PARA SER MODIFICADO

Capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.

###### CAPACIDAD PARA SER PROBADO

Facilidad con la que se pueden establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.

###### PORTABILIDAD

(ISO 25000, 2015) Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

###### ADAPTABILIDAD

Capacidad del producto que le permite ser adaptado de forma efectiva y eficiente a diferentes entornos determinados de hardware, software, operacionales o de uso.

###### CAPACIDAD PARA SER INSTALADO

Facilidad con la que el producto se puede instalar y/o desinstalar de forma exitosa en un determinado entorno.

###### CAPACIDAD PARA SER REEMPLAZADO

Capacidad del producto para ser utilizado en lugar de otro producto software determinado con el mismo propósito y en el mismo entorno.

###### HERRAMIENTAS DE DESARROLLO WEB

###### SISTEMA GESTOR DE BASE DE DATOS POSTGRESQL

PostgreSQL es un avanzado sistema de bases de datos relacionales basado en Open Source. Esto quiere decir que el código fuente del programa está disponible a cualquier persona libre de cargos directos, permitiendo a cualquiera colaborar con el desarrollo del proyecto o modificar el sistema para ajustarlo a sus necesidades. PostgreSQL está bajo licencia BSD. Un sistema de base de datos relacionales es un sistema que permite la manipulación de acuerdo con las reglas del ́algebra relacional. Los datos se almacenan en tablas de columnas y renglones.

Con el uso de llaves, esas tablas se pueden relacionar unas con otras. (Denzer, 2002)

###### FUNCIONAMIENTO

En la jerga de bases de datos, PostgreSQL usa el modelo cliente/servidor.

Una sesión en PostgreSQL consiste en ejecución de los siguientes procesos.

* + - * + El servidor, que maneja archivos de bases de datos, acepta conexiones a las aplicaciones cliente, y realiza acciones en la base de datos. El programa servidor de bases de datos se conoce como *Postmaster.*
        + La aplicación cliente, que necesita realizar operaciones en la base de datos.

Las aplicaciones cliente pueden ser de la más diversa naturaleza: pueden ser aplicaciones de texto en una consola, aplicaciones gráficas, un servidor web que accede a la base de datos para mostrar una página, o herramientas especializadas de mantenimiento de bases de datos.

###### BASE DE DATOS RELACIONAL

Una base de datos relacional desde el punto de vista del usuario podemos decir que es como una colección de tablas interrelacionadas que permiten almacenar información para que esta pueda ser utilizada posteriormente, y se basa en el modelo de datos relacional para la manipulación de las tablas, el que a su vez se basa en elementos de la teoría de conjuntos para establecer las relaciones. (Denzer, 2002)

###### LENGUAJE DE PROGRAMACION PHP (“PREPROCESADOR DE HIPERTEXTO”)

(Capuñay Uceda, 2013) PHP es un lenguaje de “código abierto” interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor. Este lenguaje

se caracteriza por que solo es interpretado pero no compilado, y es embebido en el código HTML, lo que le da un alto rendimiento y potencia.

A diferencia de otros lenguajes script como JavaScript, PHP es un lenguaje script que se ejecuta en el servidor Web, de tal manera que, solamente el resultado de su ejecución es enviado al cliente Web (navegador) tomando en cuenta lo escrito anteriormente podemos decir que, el código fuente escrito en PHP no aparecerá en el código fuente de la página Web que muestra el navegador.

Viendo el lenguaje desde el punto de vista del programador podemos decir que es un lenguaje con una sintaxis similar a C; se puede usar en tres campos: el primer el más tradicional es en los scripts del lado del servidor, el segundo es la ejecución de scripts en la línea de comandos del sistema operativo (Linux o Windows); y el tercer en el desarrollo de aplicaciones de interfaz gráfica con PHP-GTK.

###### FRAMEWORK DE DESARROLLO CODEIGNITER

Es un programa para desarrollar otros programas, CodeIgniter, por tanto, es un programa o aplicación web desarrollada en PHP para la creación de cualquier tipo de aplicación web bajo PHP. Es un producto de código libre, libre de uso para cualquier aplicación.

(Alvarez, 2010) Como cualquier otro framework, Codeigniter contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que debemos seguir para obtener provecho de la aplicación.

Esto es, marca una manera específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de

crear y mantener. CodeIgniter implementa el proceso de desarrollo llamado Model View Controller (MVC), que es un estándar de programación de aplicaciones, utilizado tanto para hacer sitios web como programas tradicionales. Este sistema tiene sus características, que veremos en artículos siguientes.

###### CARACTERISTICAS GENERALES DE CODEIGNITER

* + - * + **VERSATILIDAD**

Quizás la característica principal de CodeIgniter, en comparación con otros frameworks PHP. CodeIgniter es capaz de trabajar la mayoría de los entornos o servidores, incluso en sistemas de alojamiento compartido, donde sólo tenemos un acceso por FTP para enviar los archivos al servidor y donde no tenemos acceso a su configuración. (Alvarez, 2010)

###### COMPATIBILIDAD

CodeIgniter, al menos en el momento de escribir este artículo de desarrolloweb.com, es compatible con la versión PHP 4, lo que hace que se pueda utilizar en cualquier servidor, incluso en algunos antiguos. Por supuesto, funciona correctamente también en PHP 5.

###### FACILIDAD DE INSTALACIÓN

No es necesario más que una cuenta de FTP para subir CodeIgniter al servidor y su configuración se realiza con apenas la edición de un archivo, donde debemos escribir cosas como el acceso a la base de datos. Durante la configuración no necesitaremos acceso a herramientas como la línea de comandos, que no suelen estar disponibles en todos los alojamientos.

###### FLEXIBILIDAD

CodeIgniter es bastante menos rígido que otros frameworks. Define una manera de trabajar específica, pero en muchos de los casos podemos seguirla o no y sus reglas de codificación muchas veces nos las podemos saltar para trabajar como más a gusto encontremos. Algunos módulos como el uso de plantillas son totalmente opcionales. Esto ayuda muchas veces también a que la curva de aprendizaje sea más sencilla al principio.

###### LIGEREZA

El núcleo de CodeIgniter es bastante ligero, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. La mayoría de los módulos o clases que ofrece se pueden cargar de manera opcional, sólo cuando se van a utilizar realmente. (Alvarez, 2010)

###### HTML (“LENGUAJE DE MARCADO DE HIPERTEXTO”)

Es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). El lenguaje HTML es un estándar reconocido en todo el mundo y cuyas normas define un organismo sin ánimo de lucro llamado W3C. El propio W3C define el lenguaje HTML como "un lenguaje reconocido universalmente y que permite publicar información de forma global". Por convención, los archivos de formato HTML usan la extensión .HTM o

.HTML. (Marroquín, 2010)

###### JAVASCRIPT

JavaScript fue diseñado para ser un lenguaje de elaboración de script que pudieran incrustarse en archivos HTML. No es compilado, sino que, en vez de ello, es interpretado por el navegador. A diferencia de Java, que primero es convertido a código de byte fácil de interpretar, JavaScript es leído por el navegador como código fuente. Esto facilita el aprendizaje de JavaScript mediante ejemplos, debido a que se puede ver la manera en que otros usan JavaScript en sus páginas. (Sánchez Maza, 2012)

El lenguaje HTML permite dotar las páginas Web de una atractiva información visual, no obstante le falta cierto grado de interactividad para el usuario, es decir, en la mayoría de casos, una página Web es un mero escaparate. Para llenar el vacío existente nace JavaScript.

###### JQUERY

Es una biblioteca o framework de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas web. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

###### AJAX (JAVASCRIPT ASÍNCRONO Y XML)

Es una técnica de desarrollo web para crear aplicaciones interactivas (RIA). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en

segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

### CAPITULO 2. JUSTIFICACIÓN DEL PROYECTO

###### JUSTIFICACION CIENTIFICO-TECNICA

El sistema web utiliza lenguaje de programación libre (Open Source) PHP, el Sistema Gestor de base de Datos PostgreSQL y el patrón modelo vista controlador (MVC) centralizando toda la información de la base de datos de todos los usuario registrados, compras y ventas, este sistema mejorara las transacciones que realiza dentro de la empresa permitiendo la elaboración de reportes de compras y ventas, registros de clientes y usuarios del sistema de una forma sencilla automática y centralizada.

###### JUSTIFICACION ECONOMICA

La implementación de un sistema web dentro de la empresa para la adquisición de productos online es económicamente factible, debido a que las herramientas tecnológicas son de uso libre lo cual contribuye a la optimización de costos del proyecto, reduciendo el tiempo de procesamiento de información en el momento de realizar los registro de los productos, compras y ventas, reportes de compra y venta, a partir de estas operaciones y/o transacciones se reducirá los gastos en personal y material escritorio.

###### JUSTIFICACION SOCIAL

El sistema de web proporcionara al cliente la información seleccionada al instante, permitiendo conocer los productos que se está adquiriendo, las ventas que se están efectuando, asimismo el entorno web brindará al cliente la facilidad de poder realizar los pagos de cada uno de sus adquisiciones, y proyectando las compras y ventas para ayudar a la empresa en la toma de decisiones.

### CAPITULO 3. DISEÑO TEORICO DE LA INVESTIGACIÓN

###### PLANTEAMIENTO DEL PROBLEMA

Dentro de la empresa GROUPWARE SYSTEMS los registros de ventas y compras se lo realizan de manera manual y esto causa demora al momento de realizar un registro o buscar información del mismo. Del mismo modo este proceso influye económicamente dentro de la empresa debido al costo invertido en material de escritorio y papelería. Por otro lado la empresa GROUPWARE SYSTEMS carece de algún medio para poder promocionar sus productos y dar facilidad al cliente de comprar desde cualquier parte del territorio Nacional.

###### FORMULACION DEL PROBLEMA

***¿Cómo promocionar la adquisición de productos tecnológicos vía online en la empresa GROUPWARE SYSTEMS a nivel nacional?***

###### DELIMITACION DEL CONTENIDO

El presente proyecto refleja la elaboración de un sistema de información, comercio electrónico para la adquisición de productos online, registro de personal encargado del sistema, asignación de cargos y usuarios, registro de productos con sus respectivos subtipos, marcas y categorías para clasificar los productos, registro de clientes, registro de compras, ventas online y pagos a través de Paypal, generación de reportes y gráficos estadísticos para la toma de decisiones dentro de la empresa.

###### DELIMITACION TEMPORAL

El proyecto de grado se desarrolló en un periodo de tiempo de 8 meses empezando con el análisis de requerimientos, siguiendo con el diseño, desarrollo, prueba e implementación a través de los sprints propios de la metodología SCRUM.

###### DELIMITACION ESPACIAL

El proyecto tiene una delimitación espacial dentro de la Ciudad de El Alto, específicamente en la siguiente dirección: Av. Jorge Carrasco entre calle 1 y 2, Galería Inti Piso Nº3 Oficina 3-0 en inmediaciones de la empresa GROUPWARE SYSTEMS.

###### OBJETIVOS DE LA INVESTIGACION

###### OBJETIVO GENERAL

Desarrollar un sistema de información web para el comercio electrónico de los

productos que posee el GROUPWARE SYSTEMS, con la finalidad de promocionar

y efectuar la venta de los mismos y también la elaboración de reportes de alertas

tempranas para la toma de decisiones.

###### OBJETIVOS ESPECIFICOS

* + - * Analizar los requerimientos de la empresa para determinar las necesidades del cliente.
      * Diseñar el sistema de adquisición de productos para definir los módulos y funciones necesarias que albergará el sistema.
      * Desarrollar el sistema web con el uso de herramientas Open Source para vender productos online y registrar las compras efectuadas.
      * Implementar el sistema de información web dentro de la empresa para mejorar el registro de compra y venta automatizando el proceso.
      * Realizar pruebas del sistema web con el objetivo de verificar la aceptación del cliente y las fallas que el software pueda tener.
      * Capacitar al personal de la empresa con el fin de que el usuario tenga una mejor experiencia y haga un mejor uso del sistema.

### CAPITULO IV. DISEÑO O DISPOSITIVO DE PRUEBA

###### ARGUMENTACION DEL TIPO DE INVESTIGACION Y TIPO DE DISEÑO DE INVESTIGACION

###### TIPO DE INVESTIGACION

Debido a que existe documentación, proyectos, trabajos previos referentes al tema de este Proyecto de Grado se utilizara el MÉTODO DESCRIPTIVO, ya que consiste en llegar a conocer las situaciones predominantes a través de las actividades exactas.

###### DISEÑO DE LA INVESTIGACION

El diseño de la investigación se cataloga por NO EXPERIMENTAL ya que no hay manipulación de las variables de forma intencional para ver su efecto sobre otras variables, es decir, que no se manipula en diversos tipos de escenario y/o varias condiciones.

###### MÉTODOS TÉCNICAS E INSTRUMENTOS DE INVESTIGACIÓN A UTILIZAR

###### MÉTODO

El método usado en el presente Proyecto es el METODO DEDUCTIVO debido a que de algo general como un sistema de compras y ventas online, se implementará una empresa en particular GROUPWARE SYSTEMS.

###### TÉCNICAS

Como técnica de la investigación se definió la ENTREVISTA por el hecho de que para el desarrollo del presente software se requiere consultar, saber, conocer,

los requerimientos del cliente. Siendo así que para tal fin la entrevista es una técnica muy útil.

###### HERRAMIENTAS

Se utilizó los CUESTIONARIOS FORMULADOS CON PREGUNTAS

ABIERTAS debido a que lo primordial es lo que desea el cliente de manera explícita y completa para así cumplir con la metodología usada, y satisfacer las necesidades del cliente.

### CAPITULO V. DESCRIPCIÓN DEL PROYECTO

###### METODOLOGIA SCRUM

En el presente documento los principales motivos por la cual se escogió la metodología de desarrollo ágil SCRUM para la ejecución de este sistema son:

* + - Las características del sistema de información permiten desarrollar un producto que pueda ir incrementando las funcionalidades del sistema de acuerdo a las necesidades del usuario.
    - Entregas de productos o módulos terminados (probados, concluidos), que estén listos para la implementación, y a partir de ahí un incremento y mejora continua del sistema.
    - Apta para cambios imprevistos pero siempre con límites.

###### ROLES DEL PROYECTO SCRUM

|  |  |  |
| --- | --- | --- |
| **PERSONA** | **DESCRIPCION** | **ROL SCRUM** |
| **CARLOS BAINA** | Persona que toma decisiones, y es la  que realmente conoce el negocio del cliente y su visión del producto. | **PRODUCT OWNER** |

|  |  |  |
| --- | --- | --- |
| **CARLOS BAINA** | Encargada de comprobar que el  modelo y la metodología funciona | **SCRUM MASTER** |
| **CARLOS BAINA** | Encargado del desarrollo | **DEVELOPMENT**  **TEAM** |
| **GERENTE (ADMINISTRADOR), EMPLEADOS DE**  **GROUPWARE SYSTEMS** | Personas a la que el proyecto les producirá un beneficio | **STAKEHOLDERS** |

***TABLA 4.*** *ROLES DEL PROYECTO SCRUM*

***FUENTE.*** *ELABORACION PROPIA*

###### ARTEFACTOS SCRUM

###### HISTORIAS DE USUARIO

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | PRIOR | RIESG | USUARIO | TITULO | DESCRIPCION | ESTIM |
| **1** | Muy Alta | Alto | *Gerente* | REGISTRO DE PERSONAL | El sistema debe permitir  registrar al personal de la empresa | 2 |
| **2** | Muy Alta | Alto | *Gerente* | CUENTAS DE USUARIOS (EMPLEADOS) | El sistema debe asignar un usuario y contraseña por defecto a cada empleado. | 2 |
| **3** | Muy Alta | Alto | *Gerente* | ASIGNACION DE ROLES (USUARIOS) | El sistema debe permitir asignar un rol a cada empleado de la empresa | 1 |
| **4** | Muy Alta | Alto | *Gerente* | AÑADIR, MODIFICAR, ELIMINAR PRODUCTOS | El sistema debe permitir registrar productos al catálogo, introduciendo el precio, existe precio de oferta, o diferentes tipos  del mismo producto | 2 |
| **5** | Muy Alta | Alto | *Vendedor* | AÑADIR, MODIFICAR, ELIMINAR  CATEGORIAS | El sistema debe tener la capacidad de registrar una categoría y asignar el  producto a cada una. | 1 |
| **6** | Muy Alta | Alto | *Vendedor* | AÑADIR, MODIFICAR, ELIMINAR  MARCAS | El sistema debe permitir registrar marcas para los productos | 1 |
| **7** | Muy Alta | Alto | *Gerente* | REGISTRO DE EMPLEADOS EXTERNOS Y PROVEEDORES | El sistema debe permitir al administrador registrar proveedores y empleados de los mismos para poder | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | realizar compras de productos |  |
| **8** | Muy Alta | Alto | *Gerente* | REGISTRO DE LAS COMPRAS REALIZADAS | El sistema debe permitir registrar las compras de productos realizadas de un determinado proveedor, y para tener un mejor control debe permitir seleccionar el vendedor del proveedor  que vendió el producto | 3 |
| **9** | Muy Alta | Muy ALto | *Gerente* | VENTAS FISICAS | El sistema debe permitir realizar ventas físicas por parte de los empleados y el administrador del sistema | 3 |
| **10** | Alta | Medio | *Mensajer o* | ALERTAS TEMPRANAS | El sistema debe alertar al empleado de la empresa cuando existan productos insuficientes dentro del  catalogo | 2 |
| **11** | Alta | Alto | *Gerente* | REPORTES | El sistema debe generar reportes por producto, por compra y venta de los diferentes productos, para saber los ingresos o  egresos obtenidos | 3 |
| **12** | Alta | Bajo | *Vendedor* | CONTACTOS | El sistema debe tener una sección de contactos en donde el cliente pueda realizar sugerencias,  preguntas o reclamos. | 2 |
| **13** | Alta | Alto | *Gerente* | CATALOGO DE PRODUCTOS | El sistema debe mostrar todos los productos activos al cliente, sin necesidad de iniciar  sesión. | 1 |
| **14** | Alta | Alto | *Vendedor* | NOVEDADES (PRODUCTOS) | El sistema debe mostrar los productos que son novedades y si existen precios de oferta. | 2 |
| **15** | Alta | Alto | *Vendedor* | SELECCIÓN DE PRODUCTOS | El sistema debe mostrar los productos  organizados en  categorías, y/o marcas | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **16** | Alta | Alto | *Vendedor* | DETALLE DE PRODUCTO | El sistema debe mostrar al hacer click sobre un producto el detalle completo del mismo y al mismo tiempo mostrar productos relacionados a este. | 2 |
| **17** | Alta | Muy Alto | *Gerente* | CUENTAS DE  USUARIO (CLIENTE) | El sistema debe permitir  crear una cuenta de usuario por cliente | 2 |
| **18** | Alta | Muy Alto | *Vendedor* | CARRITO DE COMPRAS | El sistema debe permitir eliminar o añadir uno o varios productos al carrito de compras y almacenarlos dentro de la  cuenta del cliente | 4 |
| **19** | Alta | Muy Alto | *Gerente* | VENTAS ONLINE | El sistema debe permitir realizar compras de parte del cliente a través de la pasarela de pagos de PayPal y guardar dentro de la base de datos, siempre y cuando el cliente tenga la sesión  iniciada. | 5 |
| **20** | Media | Medio | *Mensajer o* | REGISTRO DE  DIRECCION DE ENVIO | El sistema debe dar la  opción al cliente de grabar una dirección de envió | 1 |
| **21** | Media | Medio | *Vendedor* | COMENTARIO DEL PRODUCTO | El sistema debe permitir a los clientes generar comentarios dentro de los diferentes productos. Para ver la opinión de  cada uno | 2 |
| **22** | Media | Alto | *Gerente* | GRAFICOS ESTADISTICOS | El sistema de generar gráficos estadísticos para ver el estado de las compras, ventas, y  productos más vendidos. | 5 |

***TABLA 5.*** *HISTORIAS DE USUARIO DEL SISTEMA*

***FUENTE.*** *ELABORACION PROPIA*

###### PRODUCT BACKLOG

Es el inventario en que se almacenan todas las funcionalidades o requisitos en forma de lista priorizada. Serán los requisitos que tendrá el producto, esta lista

será gestionada y creada por el cliente con la ayuda del Scrum Master, la estimación se realizó de acuerdo al método *planning póker* de Scrum medidas en *story points*.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **MODULO** | **PRIOR.** | **ID BACK LOG** | **ELEMENTO DEL BACKLOG** | **DESCRIPCION** | **RESPONSA BLE** | **ESTIM.** |
| **SESIONES** | Muy Alta | **1** | REGISTRO DE PERSONAL | El sistema debe permitir registrar al personal de la empresa | Carlos Baina Limpias | 3 |
| Muy Alta | **2** | CUENTAS DE USUARIOS (EMPLEADOS) | El sistema debe asignar un usuario y contraseña por defecto a cada empleado. | Carlos Baina Limpias | 5 |
| Muy Alta | **3** | ASIGNACION DE ROLES (USUARIOS) | El sistema debe permitir asignar un rol a cada empleado de la empresa | Carlos Baina Limpias | 5 |
| Muy Alta | **17** | CUENTAS DE USUARIO (CLIENTE) | El sistema debe permitir crear una cuenta de usuario por cliente | Carlos Baina Limpias | 5 |
| **GESTION DE PRODUCTOS** | Muy Alta | **5** | AÑADIR, MODIFICAR, ELIMINAR CATEGORIAS | El sistema debe tener la capacidad de registrar una categoría y asignar el producto a cada una. | Carlos Baina Limpias | 3 |
| Muy Alta | **6** | AÑADIR, MODIFICAR, ELIMINAR MARCAS | El sistema debe permitir registrar marcas para los productos. | Carlos Baina Limpias | 3 |
| Muy Alta | **4** | AÑADIR, MODIFICAR, ELIMINAR PRODUCTOS | El sistema debe permitir registrar productos al catálogo, introduciendo el precio, existe precio de oferta, o diferentes tipos del mismo producto | Carlos Baina Limpias | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Muy Alta | **10** | ALERTAS TEMPRANAS | El sistema debe alertar al empleado de la empresa cuando existan productos insuficientes dentro del catalogo | Carlos Baina Limpias | 3 |
| **VENTAS FISICAS** | Alta | **9** | VENTAS FISICAS | El sistema debe permitir realizar ventas físicas por parte de los empleados y el administrador del  sistema | Carlos Baina Limpias | 3 |
| **PRODUCTOS** | Alta | **13** | CATALOGO DE PRODUCTOS | El sistema debe mostrar todos los productos activos al cliente, sin necesidad de iniciar sesión. | Carlos Baina Limpias | 5 |
| Alta | **14** | NOVEDADES (PRODUCTOS) | El sistema debe mostrar los productos que son novedades y si existen precios de oferta. | Carlos Baina Limpias | 3 |
| Alta | **15** | SELECCIÓN DE PRODUCTOS | El sistema debe mostrar los productos organizados en categorías, y/o marcas | Carlos Baina Limpias | 2 |
| Alta | **16** | DETALLE DE PRODUCTO | El sistema debe mostrar al hacer click sobre un producto el detalle completo del mismo y al mismo tiempo mostrar productos relacionados a este. | Carlos Baina Limpias | 3 |
| Alta | **21** | COMENTARIO DEL PRODUCTO | El sistema debe permitir a los clientes generar comentarios dentro de los diferentes productos. Para ver la opinión de  cada uno | Carlos Baina Limpias | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **VENTAS ONLINE** | Alta | **19** | VENTAS ONLINE | El sistema debe permitir realizar compras de parte del cliente a través de la pasarela de pagos de PayPal y guardar dentro de la base de datos, siempre y cuando  el cliente tenga la sesión iniciada. | Carlos Baina Limpias | 8 |
| Alta | **18** | CARRITO DE COMPRAS | El sistema debe permitir eliminar o añadir uno o varios productos al carrito de compras y almacenarlos  dentro de la cuenta del cliente | Carlos Baina Limpias | 5 |
| Alta | **20** | REGISTRO DE DIRECCION DE ENVIO | El sistema debe dar la opción al cliente de grabar una dirección de envió | Carlos Baina Limpias | 3 |
| **COMPRAS Y PROVEEDOR** | Media | **7** | REGISTRO DE EMPLEADOS EXTERNOS Y PROVEEDOR | El sistema debe permitir al administrador registrar proveedores y empleados de los mismos para poder realizar  compras de productos | Carlos Baina Limpias | 3 |
| Media | **8** | REGISTRO DE LAS COMPRAS REALIZADAS | El sistema debe permitir registrar las compras de productos realizadas de un determinado proveedor, y para tener un mejor control debe permitir seleccionar el vendedor del proveedor que  vendió el producto | Carlos Baina Limpias | 3 |
| **CHARTS Y REPORTES** | Media | **11** | REPORTES | El sistema debe generar reportes por producto, por compra y venta de los diferentes productos, para saber los ingresos o egresos obtenidos | Carlos Baina Limpias | 5 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Media | **22** | GRAFICOS ESTADISTICOS | El sistema de generar gráficos estadísticos para ver el estado de las compras, ventas, y productos más vendidos. | Carlos Baina Limpias | 5 |
| **CONTACTO** | Media | **11** | CONTACTOS | El sistema debe tener una sección de contactos en donde el cliente pueda realizar sugerencias,  preguntas o reclamos. | Carlos Baina Limpias | 2 |

***TABLA 6.*** *PRODUCT BACKLOG-SISTEMA DE VENTAS*

***FUENTE.*** *ELABORACION PROPIA*

###### SPRINT BACKLOG

Para la planificación de los sprints se realizó una tabla donde permitirá conocer a detalle las tareas que corresponden a cada uno de ellos, fecha de inicio y final del sprint. Dentro del presente proyecto de grado se planifico que el sistema contendrá 10 módulos, los cuales están desarrollados en 8 sprints. A continuación para una mejor comprensión se detalla el sprint backlog:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nº** | **MODULO** | **TAREAS DEL SPRINT** | **ESFUERZO EN HORAS** | |
| **SPRINT** |  |  | **ESTIMA.** | **REAL** |
| **1** | SESIONES | Análisis de requisitos | 12 | 9 |
| Diseño del módulo | 34 | 30 |
| Desarrollo del módulo | 38 | 32 |
| Pruebas del módulo | 33 | 25 |
| **TIEMPO TOTAL DEL SPRINT** | | | **117** | **96** |
| **2** | GESTION DE PRODUCTOS | Análisis de requisitos | 13 | 10 |
| Diseño del módulo | 36 | 34 |
| Desarrollo del módulo | 37 | 33 |
| Pruebas del módulo | 35 | 30 |
| **TIEMPO TOTAL DEL SPRINT** | | | **121** | **107** |
| **3** | VENTAS FISICAS | Análisis de requisitos | 14 | 10 |
| Diseño del módulo | 28 | 25 |
| Desarrollo del módulo | 32 | 30 |
| Pruebas del módulo | 36 | 30 |
| **TIEMPO TOTAL DEL SPRINT** | | | **110** | **95** |
| **4** | PRODUCTOS | Análisis de requisitos | 12 | 10 |
| Diseño del módulo | 36 | 34 |
| Desarrollo del módulo | 39 | 35 |
| Pruebas del módulo | 32 | 30 |
| **TIEMPO TOTAL DEL SPRINT** | | | **119** | **109** |
| **5** | VENTAS ONLINE | Análisis de requisitos | 15 | 12 |
| Diseño del módulo | 39 | 38 |
| Desarrollo del módulo | 42 | 39 |
| Pruebas del módulo | 39 | 38 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TIEMPO TOTAL DEL SPRINT** | | | **135** | **127** |
| **6** | COMPRAS Y PROVEEDORES | Análisis de requisitos | 12 | 10 |
| Diseño del módulo | 33 | 30 |
| Desarrollo del módulo | 36 | 33 |
| Pruebas del módulo | 38 | 30 |
|  | **TIEMPO TOTAL DEL SPRINT** | | **119** | **103** |
| **7** | CHARTS Y REPORTES | Análisis de requisitos | 7 | 7 |
| Diseño del módulo | 30 | 25 |
| Desarrollo del módulo | 35 | 30 |
| Pruebas del módulo | 37 | 34 |
|  | **TIEMPO TOTAL DEL SPRINT** | | **109** | **96** |
| **8** | CONTACTOS | Análisis de requisitos | 9 | 9 |
| Diseño del módulo | 20 | 19 |
| Desarrollo del módulo | 29 | 19 |
| Pruebas del módulo | 25 | 20 |
|  | **TIEMPO TOTAL DEL SPRINT** | | **83** | **67** |
| **TOTAL** | | | **913** | **800** |

***TABLA 7.*** *SPRINT BACKLOG-SISTEMA DE VENTAS*

***FUENTE.*** *ELABORACION PROPIA*

###### BURNDOWN CHART (QUEMADO DE TAREAS)

Un gráfico de burndown o diagrama de quemado de tareas es una representación gráfica del trabajo por hacer en un proyecto en el tiempo. Es decir, el diagrama representa una serie temporal del trabajo pendiente. Este diagrama es útil para predecir cuándo se completará todo el trabajo. Como se ve en la FIGURA

1. El desarrollo del proyecto se realizó en un tiempo menos que lo estimado, por tal motivo el desarrollo del sistema fue un éxito.

***FIGURA 24.*** *BURNDOWN CHART DEL SISTEMA*



1000

**BURNDOWN CHART**

800

600

400

200

0

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 31 32 33

HECHO

ESTIMADO

REAL LINEA DE TENDENCIA

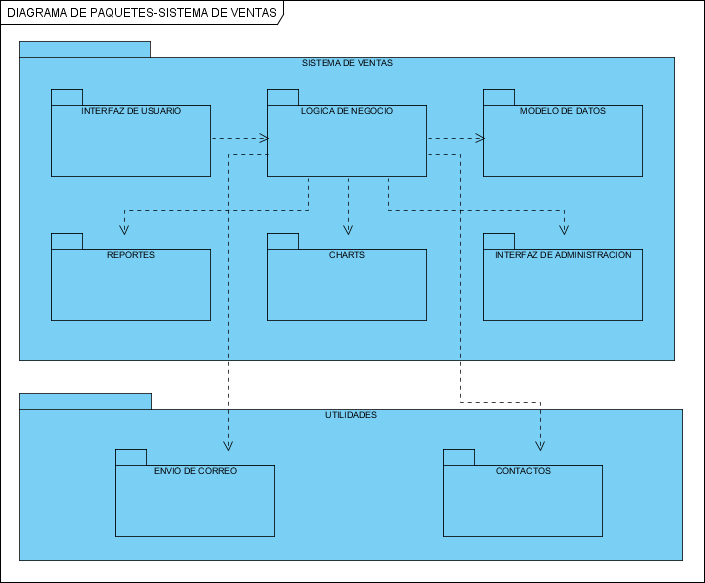
-200

***FUENTE.*** *ELABORACION PROPIA*

Siguiendo las recomendaciones definidas por UML (Lenguaje Unificado de Modelado). Este modelo dispone de una multitud de diagramas que ayudaran a comprender la complejidad del sistema. Aunque UML define una gran cantidad de diagramas para representar los distintos aspectos del desarrollo de la aplicación, se usara 5 tipos de diagramas los cuales son:

* + Diagrama de paquetes.
  + Diagrama de casos de usos.
  + Diagrama de secuencia.
  + Diagrama de actividades
  + Diagrama de estado.
  + Diagrama de clases.

###### DIAGRAMA DE PAQUETES

Por medio del siguiente diagrama se representa la estructura general del sistema, dividida en diferentes paquetes relacionados.

***FIGURA 25.*** *DIAGRAMA DE PAQUETES-SISTEMA DE VENTAS*

***FUENTE.*** *ELABORACION PROPIA*

###### PAQUETE SISTEMA DE VENTAS

Dentro del paquete de Sistema de Ventas existen 6 subpaquetes: Interfaz de usuario, Lógica de negocio, Modelo de Datos, Reportes, Charts, Interfaz de administración, los cuales parten de un núcleo central que es la lógica de negocio dentro de ella, se desarrollara todo el funcionamiento del sistema (backend) y a partir de la misma se generara los demás subpaquetes.

###### PAQUETE UTILIDADES

Dentro de este paquete se realizaran acciones extras como ser el envío de correos electrónicos indican las promociones que se tiene dentro de la empresa, al mismo tiempo se incluye el paquete de contactos para realizar, sugerencias, reclamos, observaciones, etc. Acerca de la empresa.

Una vez descrita la complejidad del sistema se comenzará con los sprints establecidos en el sprint backlog.

###### ACTORES DEL SISTEMA

En la tabla que se detalla a continuación se identifican y describen los actores que son parte del sistema desarrollado.

|  |  |
| --- | --- |
| **ACTOR** | **DESCRIPCIÓN** |
| **ADMINISTRADOR GENERAL DEL SISTEMA DE VENTAS** | Este usuario tiene acceso para crear cuentas de usuarios de los empleados, crear y administrar los productos, realizar ventas, asignar categorías y marcas a los diferentes productos, realizar ventas online y físicas, registrar proveedores, comprar productos de los distintos proveedores, reestablecer cuentas de usuario, generar reportes  y charts. |
| **EMPLEADO DE GROUPWARE SYSTEMS** | El empleado solo tiene acceso a realizar las  ventas online, físicas y poder comprar productos |

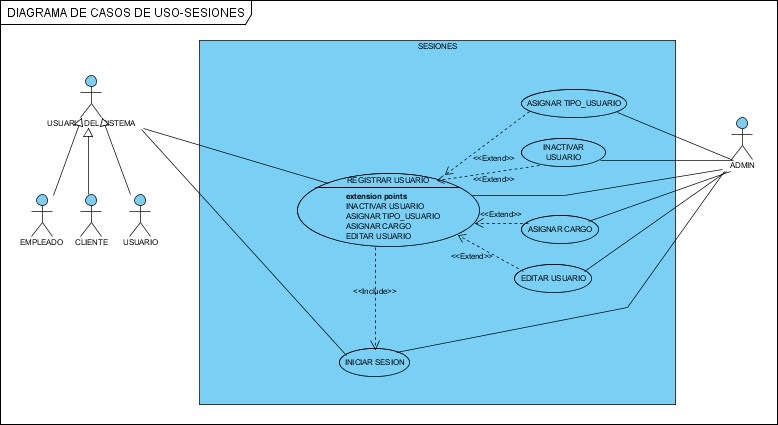
|  |  |
| --- | --- |
|  | de los distintos proveedores, visualizar las ventas  realizadas, actualizar el stock de productos. |
| **CLIENTE** | El cliente tiene acceso a ver el catálogo productos desde la interfaz del usuario, iniciar sesión, añadir productos al carrito de compras, añadir productos a la lista de deseos o favoritos, realizar compras  online. |
| **USUARIO** | Este usuario solo tiene acceso a ver el catálogo de productos, tiene la opción de añadir los  productos al carrito de compras y registrarse. |

***FIGURA 26.*** *ACTORES-SISTEMA DE VENTAS*

***FUENTE****. ELABORACION PROPIA*

###### CASOS DE USO DEL SISTEMA

* **CASO DE USO-SESIONES**



***FIGURA 27.*** *DIAGRAMA DE CASOS DE USO-SESIONES*

***FUENTE****. ELABORACION PROPIA*

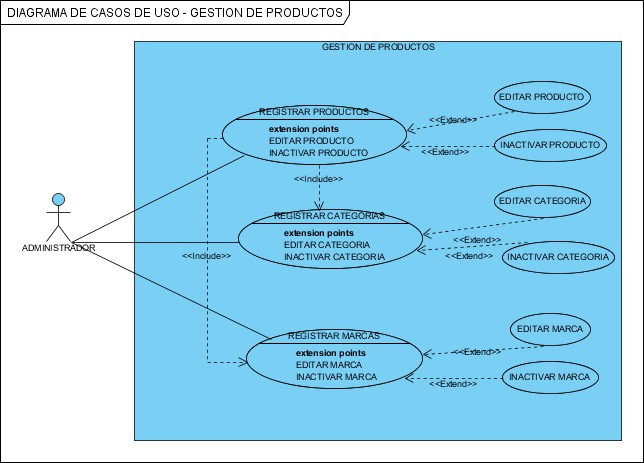
|  |  |  |
| --- | --- | --- |
| **CASO DE USO** | SESIONES | |
| **ACTOR(S)** | ADMINISTRADOR EMPLEADO, CLIENTE, USUARIO | |
| **TIPO** | PRIMARIO Y ESENCIAL | |
| **PROPÓSITO** | Registrar usuarios, para que puedan realizar una determinada acción dentro del sistema, iniciar sesión si el usuario ya se encuentra registrado. | |
| **RESUMEN** | El administrador registra a sus empleados con el fin de que puedan hacer uso del sistema, así mismo elegir el tipo de usuario que será para que pueda tener roles dentro del sistema. Sin embargo el administrador no puede eliminar una cuenta de usuario debido a que puede darse el caso de que en algún registro se esté usando, mas al contrario se puede desactivar. Por otra parte tanto como el empleado el cliente y el usuario pueden iniciar sesión dentro del sistema y al mismo tiempo pueden registrarse. | |
| **REFERENCIAS CRUZADAS** | | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El administrador registra a un nuevo empleado | 2. El sistema verifica si no existe el empleado y graba la información en la base de datos. |
| 3. El administrador asigna un cargo y un tipo de usuario al sistema. |  |
| 4. El administrador asigna un usuario y contraseña al empleado para el inicio de sesión | 5. El sistema guarda la información del empleado |
| 6. El empleado inicia sesión por primera vez | 7. El sistema pide al usuario cambiar de contraseña por temas de seguridad |
| 8. El empleado cambia de contraseña | 9. El sistema guarda en la información en la base de datos |
| 10. El usuario se registra | 11. El sistema verifica si es un usuario nuevo, y graba la  información en la base de datos |

|  |  |  |
| --- | --- | --- |
|  | 12. El usuario se convierte en un cliente e inicia sesión en el sistema |  |
| **CURSO ALTERNATIVO DE EVENTOS.** | **En el paso 2.** Si el sistema verifica que ya existe el empleado mandara un mensaje de advertencia indicando que ya existe el empleado.  **En el paso 11.** Si el sistema verifica que ya existe el cliente mandara un mensaje de advertencia indicando que ya existe el cliente y le preguntara si olvido la contraseña | |
| **PRECONDICIÓN** | El administrador del sistema ha sido aceptado en el sistema con el rol de administrador. | |
| **POSCONDICIÓN** | Se ha registrado los datos de los empleados y los clientes en la base de datos. | |

***TABLA 8****. CASO DE USO-SESIONES*

***FUENTE.*** *ELABORACION PROPIA*

###### CASO DE USO - GESTION DE PRODUCTOS



***FIGURA 28****. DIAGRAMA DE CASOS DE USO- GESTION DE PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

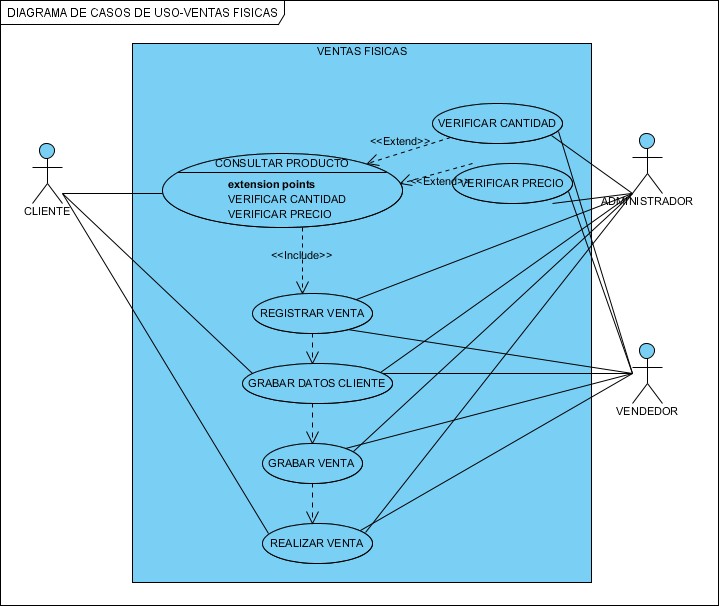
|  |  |  |
| --- | --- | --- |
| **CASO DE USO** | GESTION DE PRODUCTOS | |
| **ACTOR(S)** | ADMINISTRADOR | |
| **TIPO** | PRIMARIO Y ESENCIAL | |
| **PROPÓSITO** | Registrar las marcas, categorías de los diferentes productos, paralelamente realizar el registro de los productos asignando una marca y categoría | |
| **RESUMEN** | El administrador registra las diferentes marcas y categorías que se manejan dentro de GROUPWARE SYSTEMS. Paralelamente se registran los productos asignando una categoría y una marca, registrando los atributos propios del producto y definiendo si el producto tiene algún precio de oferta o alguna variación dentro del stock. No debe existir el mismo producto más de 1 vez | |
| **REFERENCIAS CRUZADAS** | R1 | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El administrador registra una nueva categoría | 2. El sistema verifica si no existe la categoría y graba la información en la base de datos. |
| 3. El administrador registra una nueva marca | 4. El sistema verifica si no existe la marca y graba la información en la base de datos |
| 5. El administrador registra un nuevo producto con su respectiva variación | 6. El sistema solicita al administrador que seleccione la categoría y la marca a la que pertenece el producto |
| 7. El administrador modifica el producto | 8. El sistema valida los datos introducidos y actualiza la base de datos |
| 9. El administrador modifica la categoría | 9. El sistema valida los datos introducidos y actualiza la base de datos |
| 10. El administrador modifica la marca | 11. El sistema valida los datos introducidos y actualiza la base de datos |

|  |  |
| --- | --- |
| **CURSO**  **ALTERNATIVO DE EVENTOS.** | **En el paso 2, 4 y 6.** Si el sistema verifica que ya existe el objeto  indicado mandara un mensaje de advertencia indicando que ya existe. |
| **PRECONDICIÓN** | El administrador del sistema ha sido aceptado en el sistema con el rol de administrador. |
| **POSCONDICIÓN** | Se ha registrado los datos de las categorías, marcas y productos en la base de datos. |

***TABLA 9.*** *CASO DE USO- GESTION DE PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

###### CASOS DE USO – VENTAS FISICAS



***FIGURA 29.*** *DIAGRAMA DE CASOS DE USO – VENTAS FISICAS*

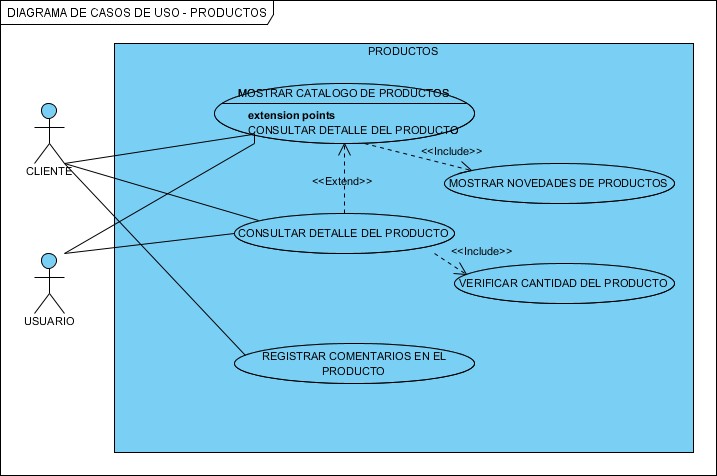
***FUENTE.*** *ELABORACION PROPIA*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CASO DE USO** | VENTAS FISICAS | | | |
| **ACTOR(S)** | ADMINISTRADOR, VENDEDOR Y CLIENTE | | | |
| **TIPO** | PRIMARIO Y ESENCIAL | | | |
| **PROPÓSITO** | Realizar ventas físicas, es decir que el cliente puede apersonarse a la tienda y sin necesidad de estar registrado en el sistema puede adquirir un producto. | | | |
| **RESUMEN** | Tanto el administrador como el vendedor pueden ser los encargados de realizar las ventas físicas, pueden registrar los datos del cliente, verificar el producto deseado y la cantidad de existencias de dicho producto, y registrar las ventas, siempre y cuando el cliente realice el respectivo pago por el producto solicitado. | | | |
| **REFERENCIAS**  **CRUZADAS** | R1, R2 | | | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | | **EVENTO SISTEMA** | |
| 1. El administrado o empleado verifica producto solicitado por  cliente | el el el | 2. El sistema devuelve información del producto | la |
| 3. El administrador o el empleado verifica las existencias y el precio del  producto | | 4. El sistema retorna información solicitada | la |
| 5. El cliente decide realizar la compra | |  | |
| 6. El administrador o empleado registra la venta | el | 7. El sistema verifica los datos ingresados | |
| 8. El cliente registra datos personales | | 9. El sistema graba la información en la base de datos | |
|  | | 10. El sistema graba la venta | |
| **CURSO ALTERNATIVO DE EVENTOS.** | **En el paso 3** si no existen unidades suficientes disponibles se vuelve al paso 1. | | | |
| **PRECONDICIÓN** | El administrador o empleado ha sido aceptado en el sistema  con el rol de administrador. | | | |
| **POSCONDICIÓN** | Se ha registrado la venta en la base de datos. | | | |

***TABLA 10.*** *CASO DE USO –VENTAS FISICAS*

***FUENTE.*** *ELABORACION PROPIA*

###### CASOS DE USO - PRODUCTOS



***FIGURA 30.*** *DIAGRAMA DE CASOS DE USO – PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

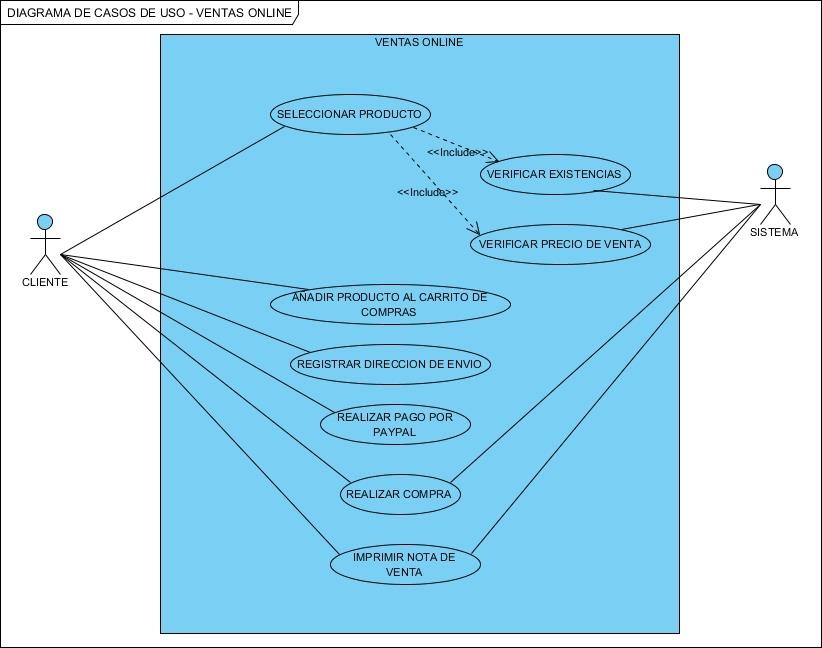
|  |  |  |
| --- | --- | --- |
| **CASO DE USO** | PRODUCTOS | |
| **ACTOR(S)** | CLIENTE, USUARIO | |
| **TIPO** | PRIMARIO E IMPLEMENTACION | |
| **PROPÓSITO** | Mostrar el catálogo de productos que ofrece la empresa con sus respectivos detalles y características, así mismo mostrar varias opciones del mismo. | |
| **RESUMEN** | Tanto el cliente que se encuentra ya registrado del sistema, o cualquier usuario que ingresa a la página pueden ver el catálogo de productos que ofrece la empresa, las variaciones, las marcas, las categorías, sin embargo solo el cliente registrado en el sistema puede realizar comentarios acerca del producto, para valorarlo. | |
| **REFERENCIAS CRUZADAS** | R1, R2 | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El cliente o el usuario ingresan al sitio | 2. El sistema despliega todo el catálogo de productos relevantes |

|  |  |  |
| --- | --- | --- |
|  | 3. El cliente o usuario seleccionan una categoría dentro del catálogo de productos | 4. El sistema retorna la información solicitada |
|  | 5. El sistema filtra los productos por categoría y marca |
| 6. El cliente o usuario selecciona un producto | 7. El sistema despliega todo el detalle del producto con la cantidad existente |
| 8. El cliente ya registrado tiene la opción de añadir un comentario del producto | 9. El sistema graba la información en la base de datos |
| **CURSO ALTERNATIVO DE EVENTOS.** | **En el paso 8** si el cliente no está registrado dentro del sistema no puede realizar ningún comentario acerca del producto. | |
| **PRECONDICIÓN** | Ingresar al sitio web de la empresa GROUPWARE SYSTEMS | |
| **POSCONDICIÓN** | Se ha registrado los comentarios realizados en la base de datos | |

***TABLA 11.*** *CASO DE USO – PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

###### CASOS DE USO – VENTAS ONLINE



***FIGURA 31.*** *CASOS DE USO – VENTAS ONLINE*

***FUENTE****. ELABORACION PROPIA*

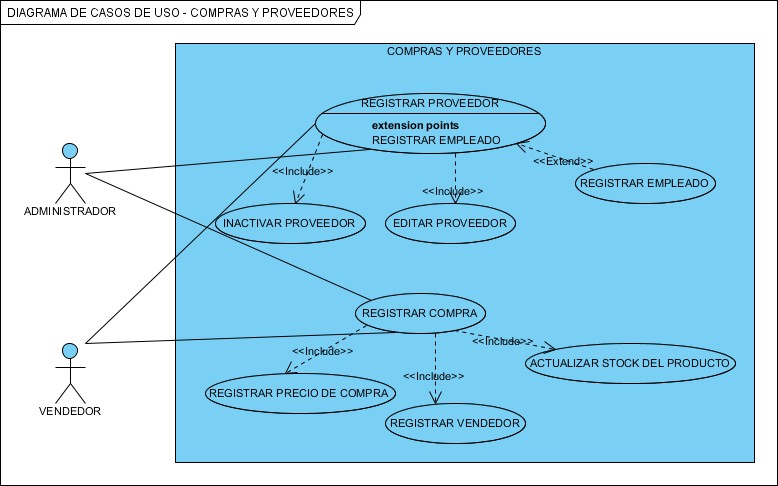
|  |  |
| --- | --- |
| **CASO DE USO** | VENTAS ONLINE |
| **ACTOR(S)** | CLIENTE, SISTEMA |
| **TIPO** | PRIMARIO Y ESENCIAL |
| **PROPÓSITO** | El cliente debe seleccionar los productos que desea comprar y añadirlos al carrito de compras, para posteriormente realizar la compra mediante PayPal. |

|  |  |  |
| --- | --- | --- |
| **RESUMEN** | El cliente debe estar registrado, luego deberá seleccionar los productos que desea añadiendo al carrito de compras, una vez añadidos se pasa al siguiente paso que será seleccionar una dirección de envió, luego deberá realizar el pago mediante PayPal siempre y cuando tenga dinero disponible, posteriormente una vez realizada la compra el sistema desplegara la nota de venta que servirá como comprobante de la compra | |
| **REFERENCIAS**  **CRUZADAS** | R1, R2 | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El cliente inicia sesión en el sistema | 2. El sistema despliega todo el catálogo de productos relevantes |
| 3. El cliente selecciona los productos añadiendo al carrito de compras | 4. El sistema guarda la información del carrito de compras dentro de la cuenta del cliente |
| 5. El cliente decide comprar y pasa al paso de seleccionar dirección de envió |  |
| 6. El cliente realiza el pago mediante la pasarela de pagos de PayPal |  |
|  | 6. El sistema verifica la  transacción |
|  | 7. El sistema despliega la nota de venta |
| **CURSO ALTERNATIVO DE EVENTOS.** | **En el paso 7.** Si el cliente no cuenta con saldo en la cuenta de PayPal el sistema no permitirá realizar la transacción. | |
| **PRECONDICIÓN** | Iniciar sesión en el sistema | |
| **POSCONDICIÓN** | Se ha registrado la venta | |

***FIGURA 32.*** *CASO DE USO – VENTAS ONLINE*

***FUENTE.*** *ELABORACION PROPIA*

###### CASOS DE USO – COMPRAS Y PROVEEDORES



***FIGURA 33.*** *DIAGRAMA DE CASOS DE USO – COMPRAS Y PROVEEDORES*

***FUENTE.*** *ELABORACION PROPIA*

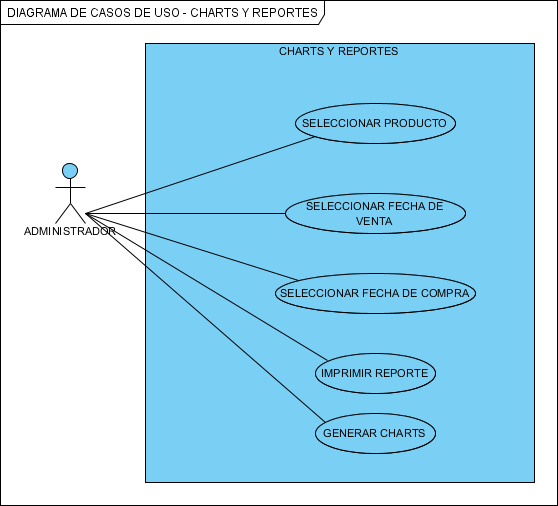
|  |  |  |
| --- | --- | --- |
| **CASO DE USO** | COMPRAS Y PROVEEDORES | |
| **ACTOR(S)** | ADMINISTRADOR Y VENDEDOR | |
| **TIPO** | SECUNDARIO Y ESENCIAL | |
| **PROPÓSITO** | El administrador o el vendedor pueden registrar los productos adquiridos para la empresa, solo si existe un proveedor ya registrado dentro del sistema. Caso contrario se debe registrar. | |
| **RESUMEN** | Tanto como el administrador y el vendedor deben realizar las compras de los productos adquiridos siendo así que se actualiza el stock de los productos y se graban los precios de compra. Asimismo el sistema para un mejor control de las compras añade a los diferentes proveedores que tiene la empresa y sus respectivos empleados los cuales realizan la venta hacia la empresa. | |
| **REFERENCIAS CRUZADAS** | R1, R2 | |
|  | **EVENTO ACTOR** | **EVENTO SISTEMA** |

|  |  |  |
| --- | --- | --- |
| **CURSO NORMAL DE EVENTOS** | 1. El administrador o el vendedor inician sesión dentro del sistema | 2. El sistema despliega el respectivo rol asignado |
| 3. El administrador o vendedor registran al proveedor con los diferentes  datos | 4. El sistema graba la información proporcionada |
| 5. El administrador o proveedor registra al vendedor del proveedor | 6. El sistema graba la información proporcionada |
| 7. El cliente registra la compra con los datos del vendedor y del producto | 8. El sistema graba la información en la base de datos y actualiza el stock |
| **CURSO**  **ALTERNATIVO DE EVENTOS.** | **En el paso 5.** Si no existe el proveedor respectivo que hace la venta a la empresa no se podrá proseguir al siguiente paso. | |
| **PRECONDICIÓN** | Ingresar al sitio web de la empresa GROUPWARE SYSTEMS | |
| **POSCONDICIÓN** | Se ha registrado los proveedores y la compra | |

***TABLA 12.*** *CASO DE USO – COMPRAS Y VENDEDORES*

***FUENTE****. ELABORACION PROPIA*

###### CASOS DE USO CHARTS Y REPORTES



***FIGURA 34.*** *DIAGRAMA DE CASOS DE USO – CHARTS Y REPORTES*

***FUENTE.*** *ELABORACION PROPIA*

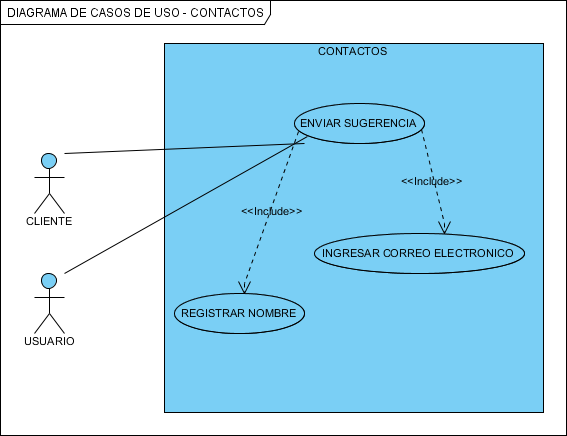
|  |  |  |
| --- | --- | --- |
| **CASO DE USO** | CHARTS Y REPORTES | |
| **ACTOR(S)** | ADMINISTRADOR | |
| **TIPO** | PRIMARIO Y ESENCIAL | |
| **PROPÓSITO** | El administrador puede generar reportes o charts (gráficos estadísticos) para saber el estado de las ventas, las compras, los productos más o menos vendidos | |
| **RESUMEN** | Solo el administrador puede generar reportes o gráficos estadísticos del estado de las ventas, el estado de las compras, o los productos que más se vendió, el administrador puede seleccionar si el reporte se generará en Excel o en Pdf dependiendo el caso. | |
| **REFERENCIAS**  **CRUZADAS** | R1, R2 | |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El administrador inicia sesión dentro del sistema | 2. El sistema despliega el respectivo rol asignado |
| 3. El administrador  selecciona la opción de reportes | 4. El sistema despliega la pantalla de reportes |
| 5. El administrador selecciona de que objeto se realizara el reporte |  |
| 6. El administrador selecciona la fecha de la información solicitada | 7. El sistema pregunta en que formato desea el reporte |
| 8. El administrador selección la opción de charts | 9. El sistema despliega la pantalla de charts |
| 10. El administrador selecciona la fecha y la transacción a ser generar en  forma de chart | 11. El sistema despliega la información solicitada |
| **CURSO ALTERNATIVO DE EVENTOS** | **En el paso 1.** Si el administrador no inicio sesión no podrá generar los reportes y los charts | |
| **PRECONDICIÓN** | Iniciar sesión en el sistema como administrador | |

|  |  |
| --- | --- |
| **POSCONDICIÓN** | Se ha generado los reportes y los charts |

***TABLA 13.*** *CASO DE USO – CHARTS Y REPORTES*

***FUENTE.*** *ELABORACION PROPIA*

###### CASOS DE USO – CONTACTOS



***FIGURA 35.*** *DIAGRAMA DE CASOS DE USO – CONTACTOS*

***FUENTE.*** *ELABORACION PROPIA*

|  |  |
| --- | --- |
| **CASO DE USO** | CONTACTOS |
| **ACTOR(S)** | USUARIO, CLIENTE |
| **TIPO** | SECUNDARIO Y OPCIONAL |
| **PROPÓSITO** | Un usuario no registrado o un cliente registrado puede realizar sugerencia acerca de la empresa o el sitio web. |
| **RESUMEN** | El usuario o cliente pueden realizar sugerencias, observaciones o consultas acerca de la empresa o sitio web, ingresando el nombre, el correo electrónico y la sugerencia. |
| **REFERENCIAS CRUZADAS** |  |

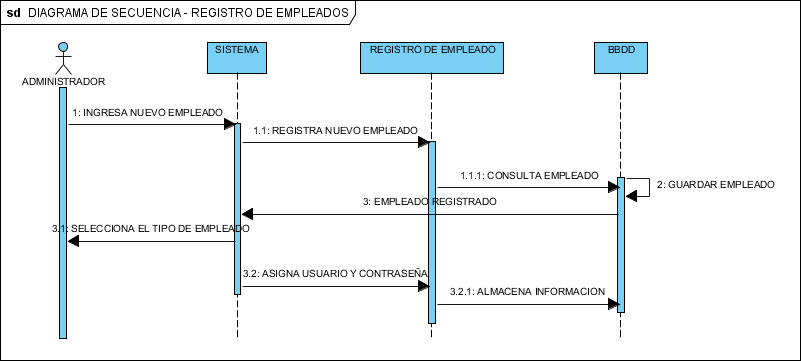
|  |  |  |
| --- | --- | --- |
| **CURSO NORMAL DE EVENTOS** | **EVENTO ACTOR** | **EVENTO SISTEMA** |
| 1. El cliente o el usuario entran a la página de contactos | 2. El sistema despliega la página de contactos |
| 3. El cliente o usuario manda la sugerencia o consulta | 4. El sistema alerta al administrador o al empleado mediante el sistema |
| **CURSO ALTERNATIVO DE EVENTOS** | **En el paso 3.** Si el cliente o empleado no registra alguno de los campos los datos no pueden ser enviados | |
| **PRECONDICIÓN** | Ingresar al sitio web de GROUPWARE SYSTEMS | |
| **POSCONDICIÓN** | Se ha enviado la sugerencia o consulta | |

***TABLA 14.*** *CASO DE USO – CONTACTOS*

***FUENTE.*** *ELABORACION PROPIA*

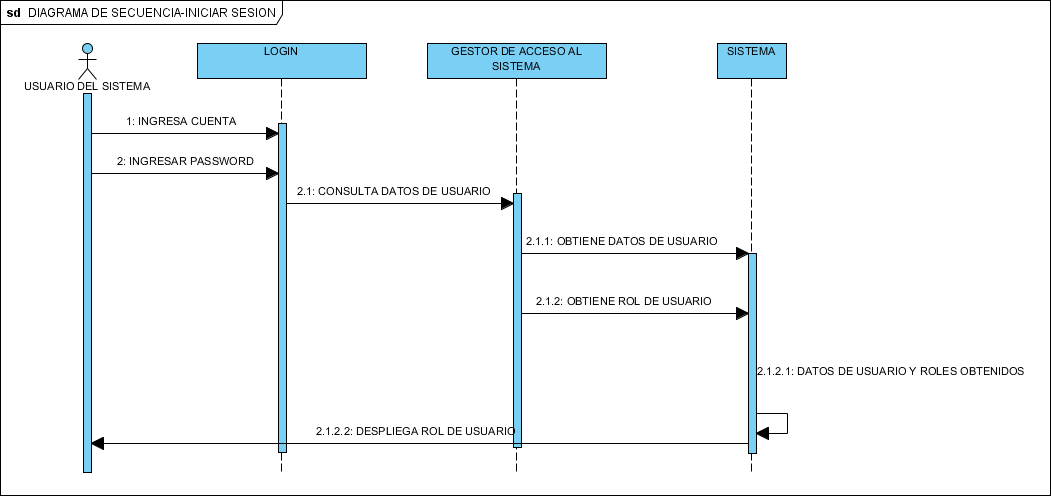
###### DIAGRAMAS DE SECUENCIA DEL SISTEMA

* **DIAGRAMA DE SECUENCIA DEL MODULO DE SESIONES**



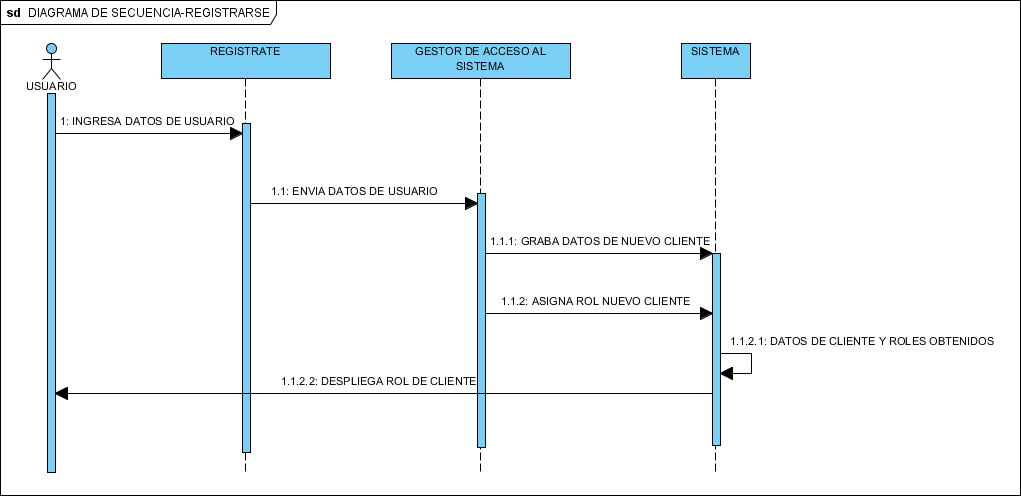
***FIGURA 36.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE EMPLEADOS*

***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 37.*** *DIAGRAMA DE SECUENCIA - INIICIAR SESION*

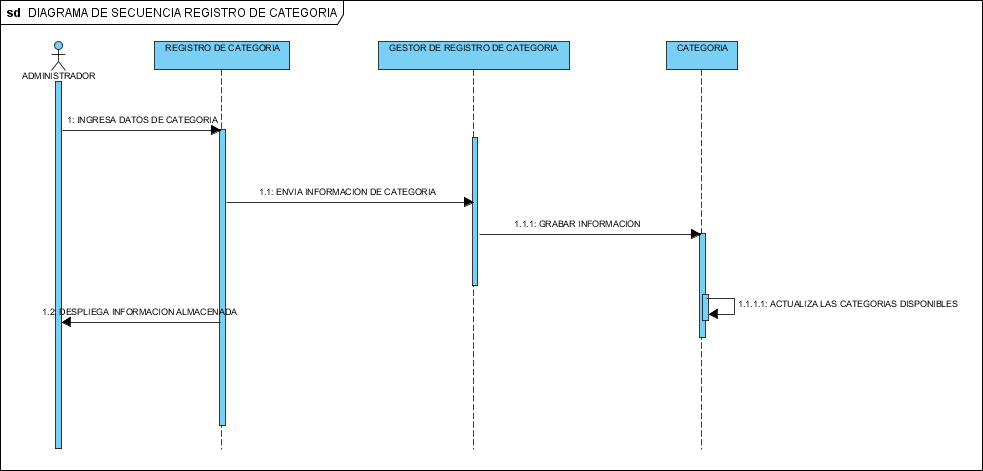
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 38****. DIAGRAMA DE SECUENCIA – REGISTRARSE*

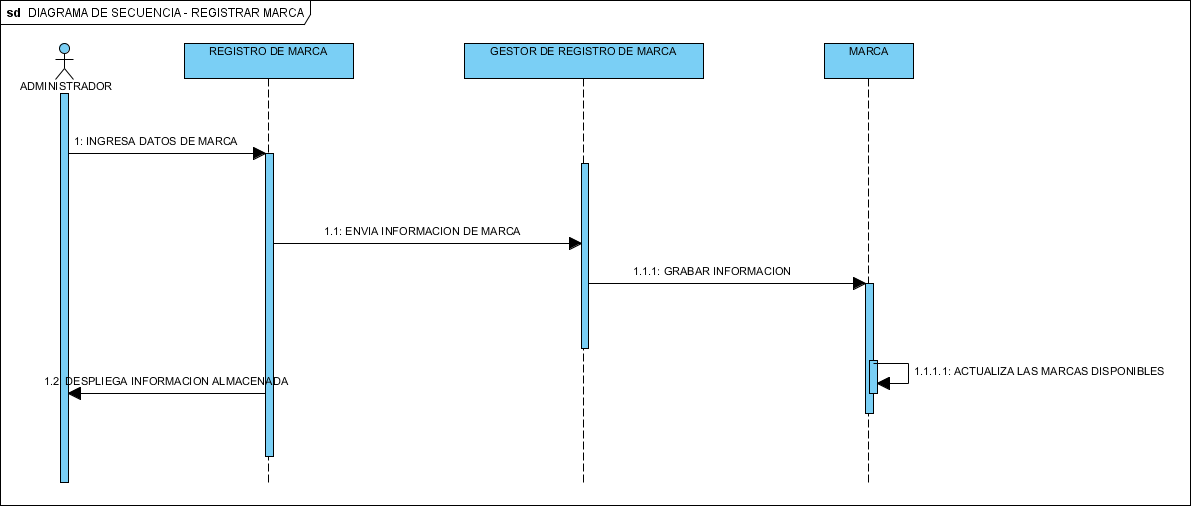
***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE SECUENCIA DEL MODULO GESTION DE PRODUCTOS



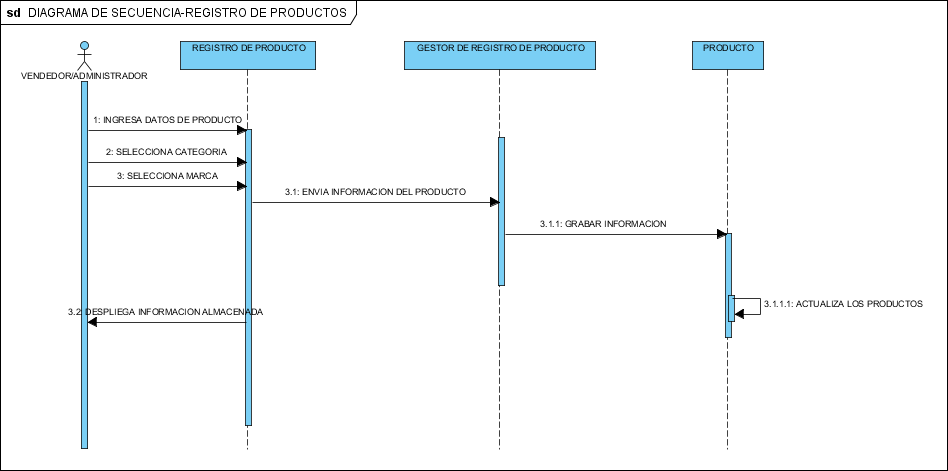
***FIGURA 39.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE CATEGORIA*

***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 40.*** *DIAGRAMA DE SECUENCIA – REGISTRAR MARCA*

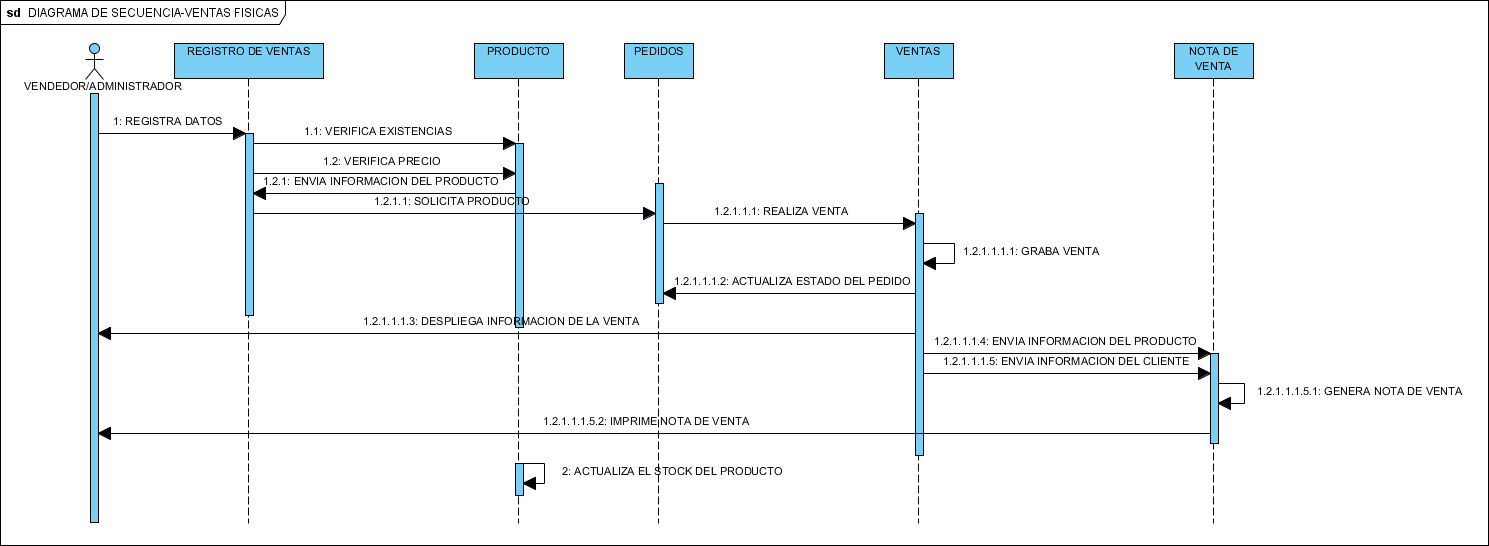
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 41.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

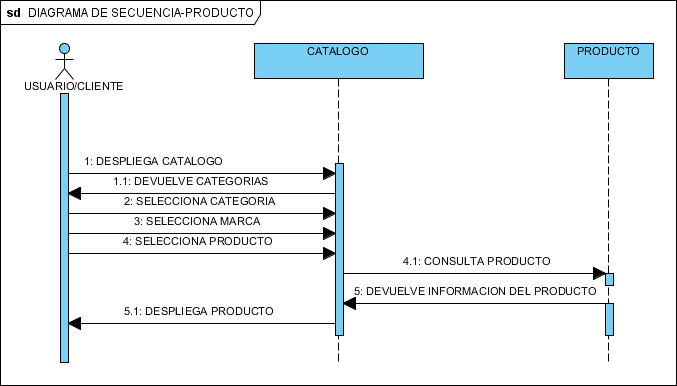
###### DIAGRAMA DE SECUENCIA DEL MODULO DE VENTAS FISICAS



***FIGURA 42.*** *DIAGRAMA DE SECUENCIA – VENTAS FISICAS*

***FUENTE****. ELABORACION PROPIA*

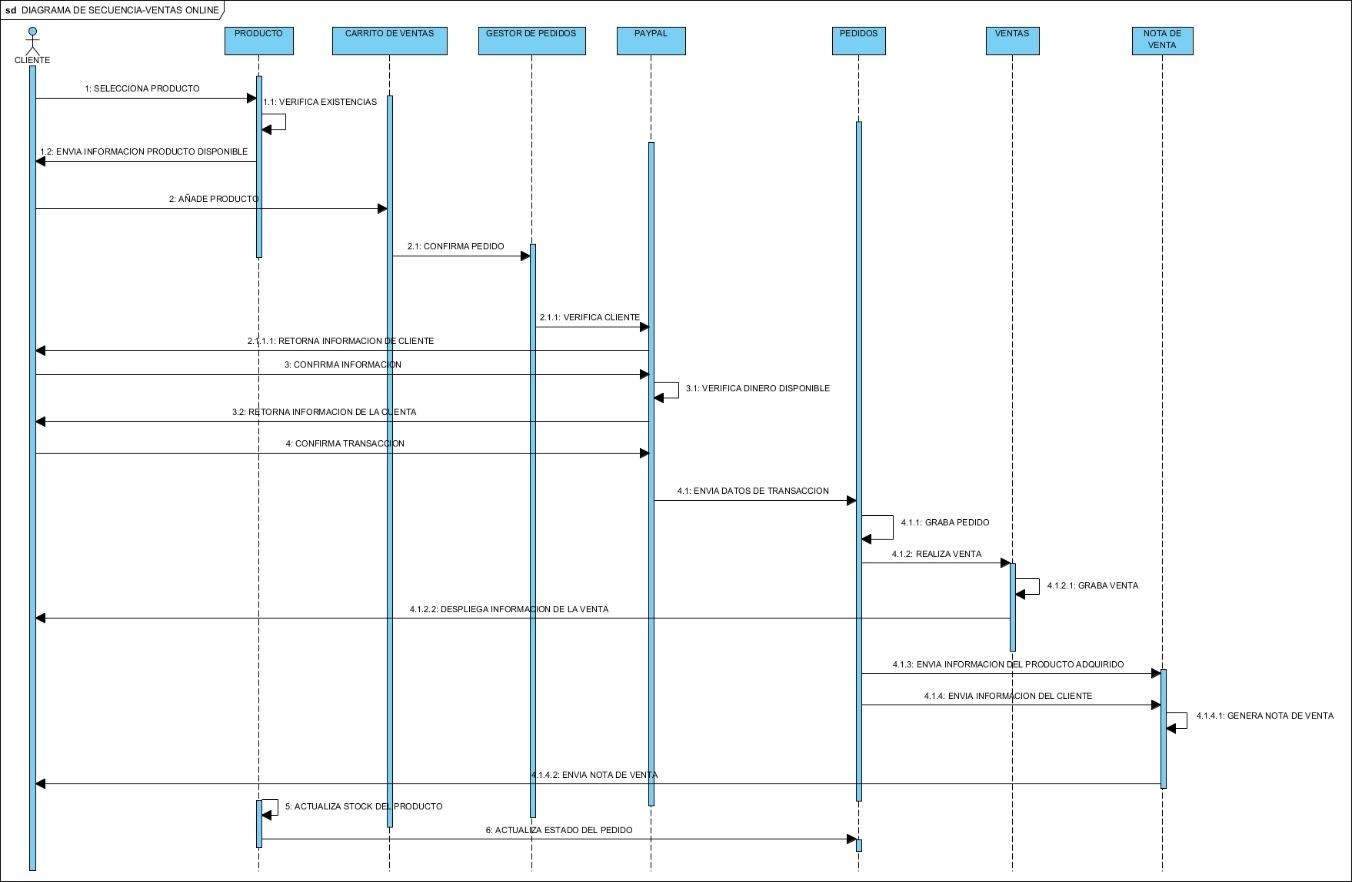
###### DIAGRAMA DE SECUENCIA DEL MODULO PRODUCTOS



***FIGURA 43.*** *DIAGRAMA DE SECUENCIA – PRODUCTOS*

***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE SECUENCIA DEL MODULO DE VENTAS ONLINE



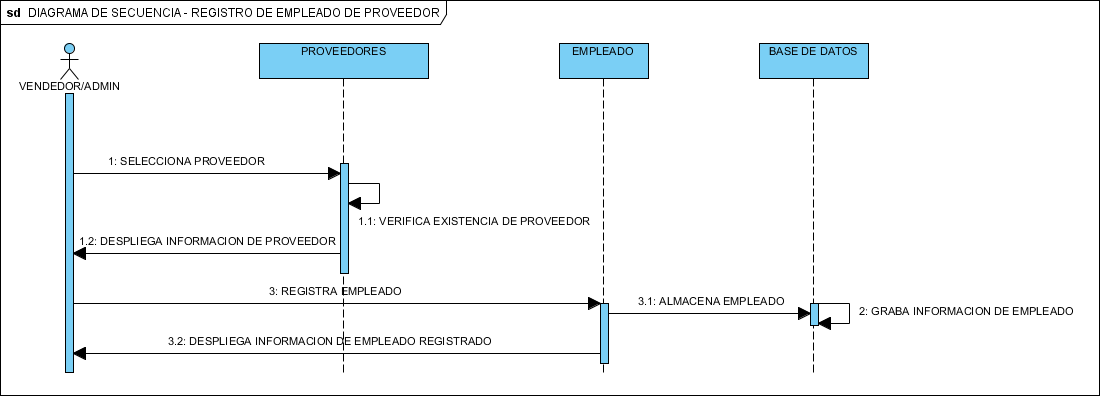
***FIGURA 44.*** *DIAGRAMA DE SECUENCIA – VENTAS ONLINE*

***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE SECUENCIA DEL MODULO DE COMPRAS Y PROVEEDORES

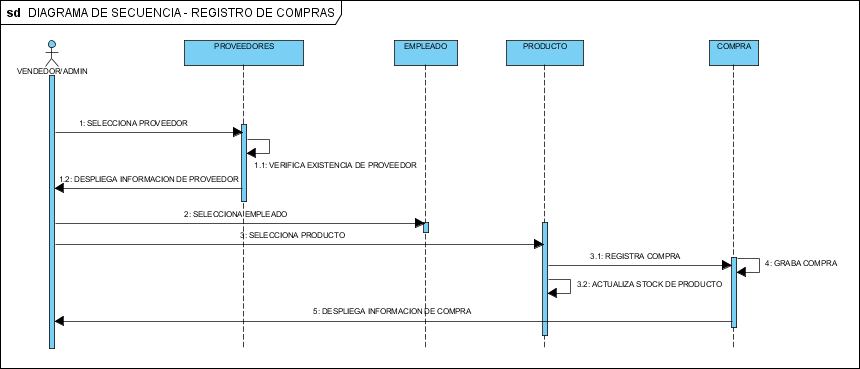
***FIGURA 45.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE PROVEEDORES*

***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 46.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE EMPLEADO DE PROVEEDOR*

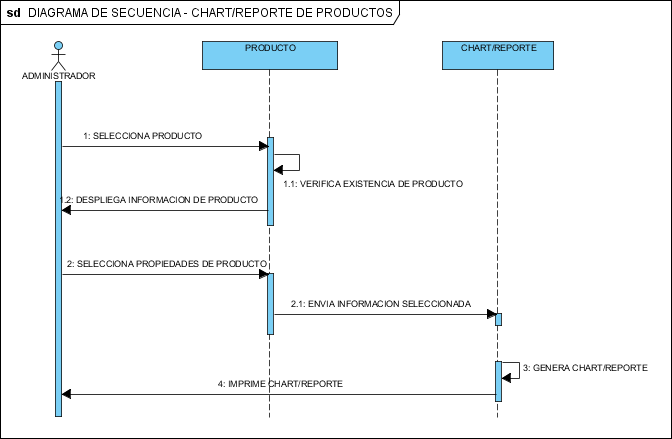
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 47.*** *DIAGRAMA DE SECUENCIA – REGISTRO DE COMPRAS*

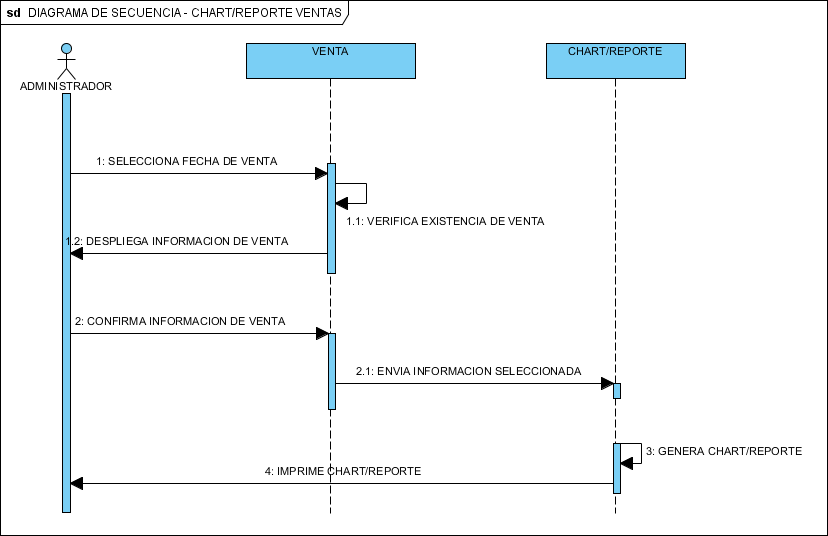
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE SECUENCIA DEL MODULO DE CHARTS Y REPORTES



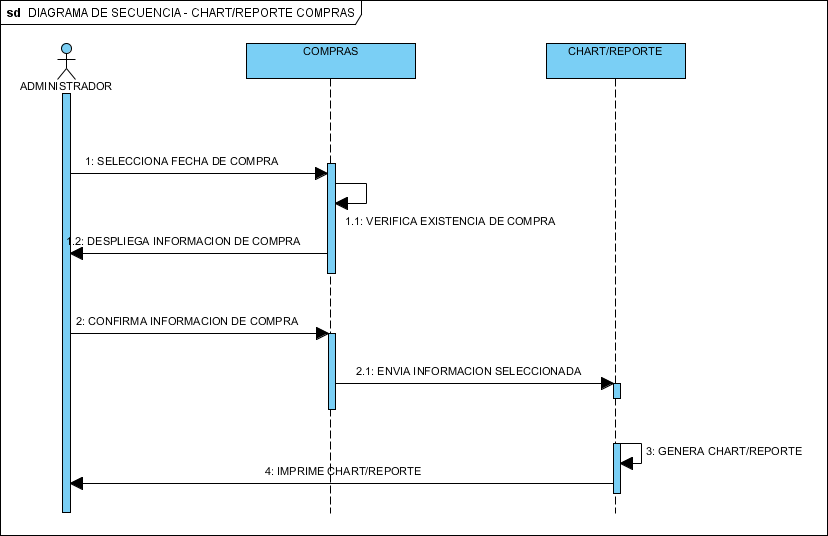
***FIGURA 48****. DIAGRAMA DE SECUENCIA – REPORTE DE PRODUCTOS*

***FUENTE****. ELABORACION PROPIA*



**FIGURA 49.** DIAGRAMA DE SECUENCIA – CHART/REPORTE VENTAS

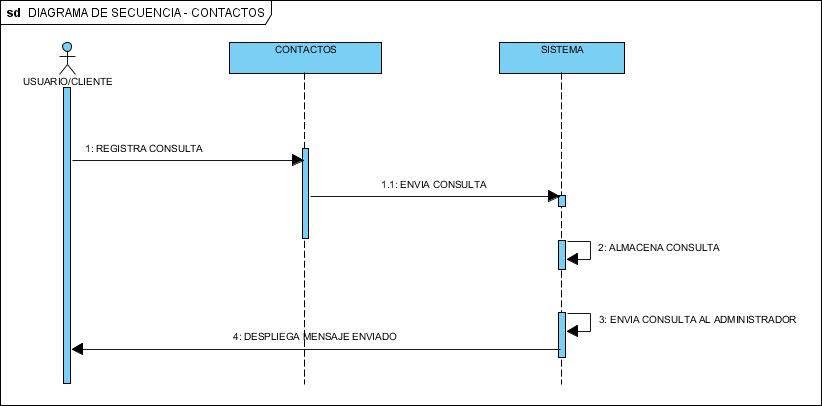
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 50.*** *DIAGRAMA DE SECUENCIA – CHART/REPORTE COMPRAS*

***FUENTE.*** *ELABORACION PROPIA*

* **DIAGRAMA DE SECUENCIA DEL MODULO CONTACTOS**

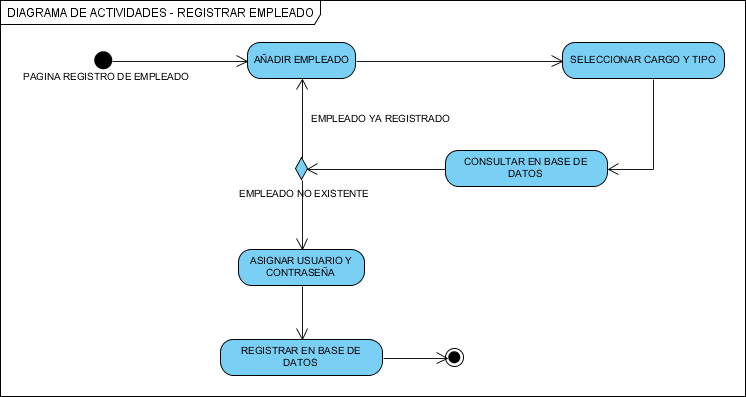


***FIGURA 51****. DIAGRAMA DE SECUENCIA – CONTACTOS*

***FUENTE.*** *ELABORACION PROPIA*

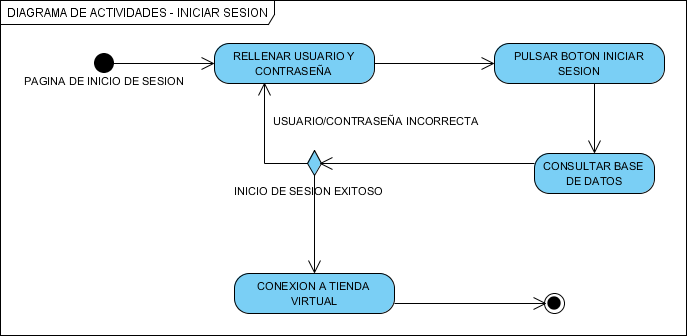
###### DIAGRAMA DE ACTIVIDADES DEL SISTEMA

* **DIAGRAMA DE ACTIVIDADES DEL MODULO SESIONES**



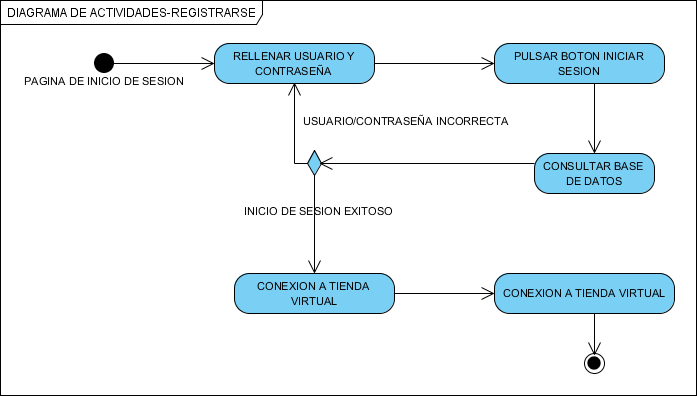
***FIGURA 52.*** *DIAGRAMA DE ACTIVIDADES – REGISTRAR EMPLEADO*

***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 53.*** *DIAGRAMA DE ACTIVIDADES – INICIAR SESION*

***FUENTE****. ELABORACION PROPIA*



***FIGURA 54****. DIAGRAMA DE ACTIVIDADES – REGISTRARSE*

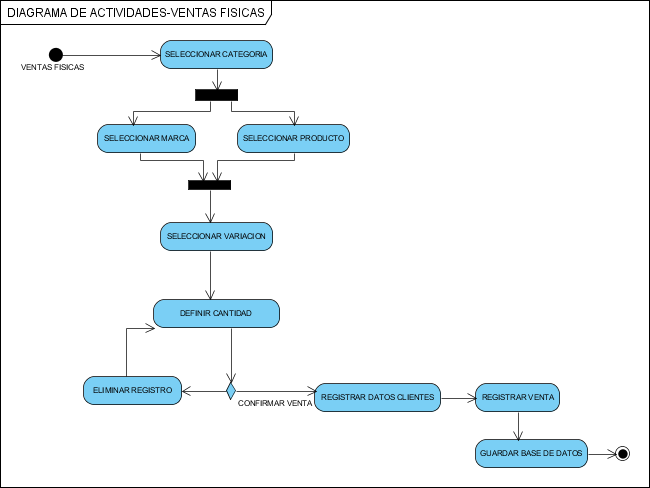
***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE GESTION DE PRODUCTOS

***FIGURA 55.*** *DIAGRAMA DE ACTIVIDADES – GESTION DE PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

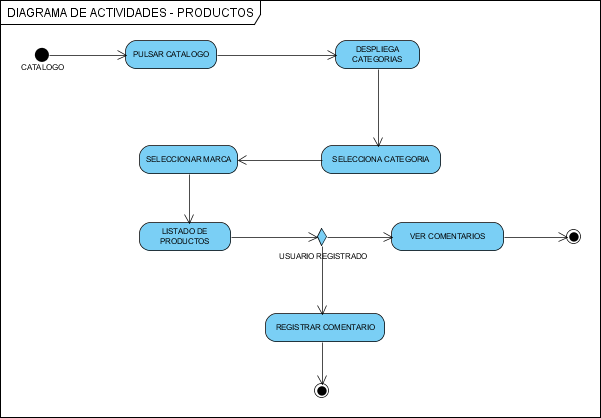
###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE VENTAS FISICAS



***FIGURA 56.*** *DIAGRAMA DE ACTIVIDADES – VENTAS FISICAS*

***FUENTE.*** *ELABORACION PROPIA*

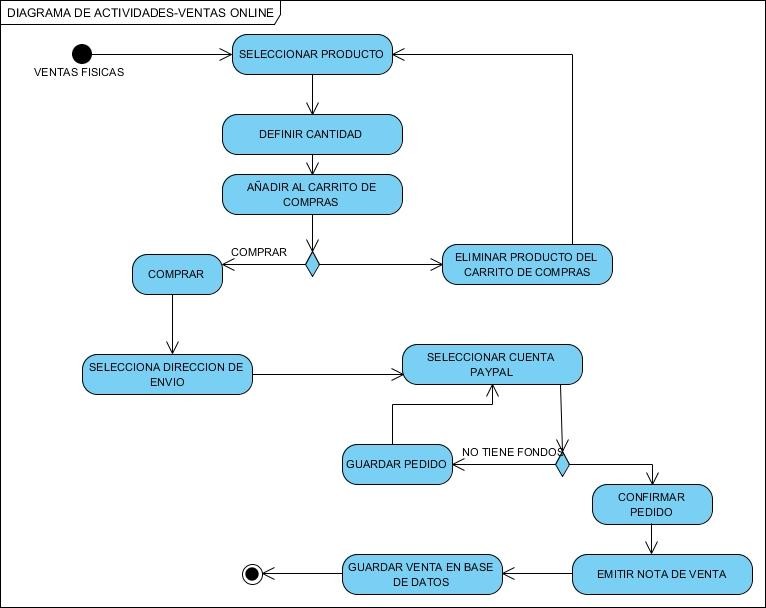
###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE PRODUCTOS



***FIGURA 57.*** *DIAGRAMA DE ACTIVIDADES – PRODUCTOS*

***FUENTE****. ELABORACION PROPIA*

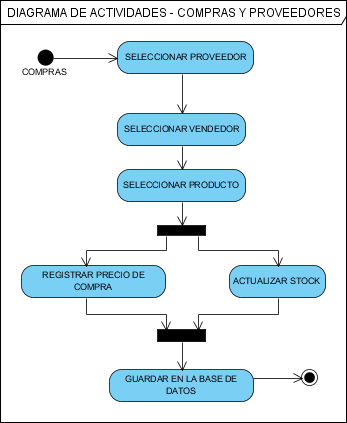
###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE VENTAS ONLINE



***FIGURA 58.*** *DIAGRAMA DE ACTIVIDADES – VENTAS ONLINE*

***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE COMPRAS Y PROVEEDORES



***FIGURA 59.*** *DIAGRAMA DE ACTIVIDADES – COMPRAS Y PROVEEDORES*

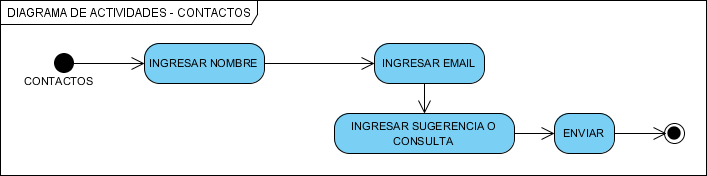
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE CHARTS Y REPORTES

***FIGURA 60.*** *DIAGRAMA DE ACTIVIDADES – CHARTS Y REPORTES*

***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE ACTIVIDADES DEL MODULO DE CONTACTOS

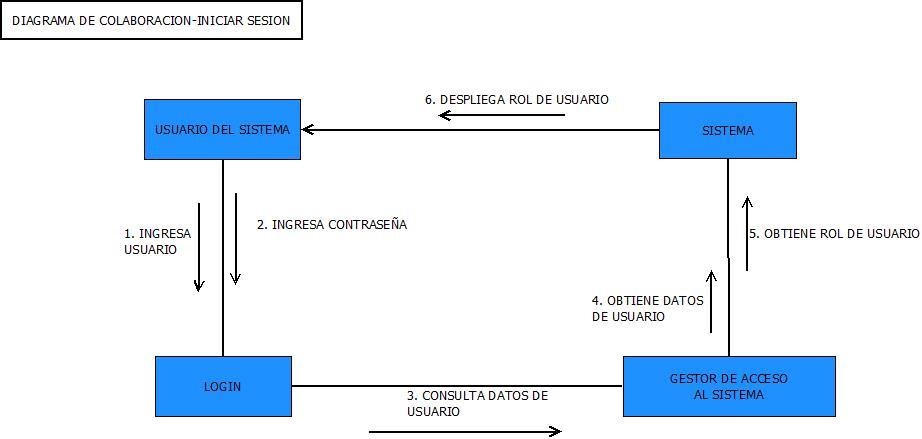


***FIGURA 61.*** *DIAGRAMA DE ACTIVIDADES – CONTACTOS*

***FUENTE.*** *ELABORACION PROPIA*

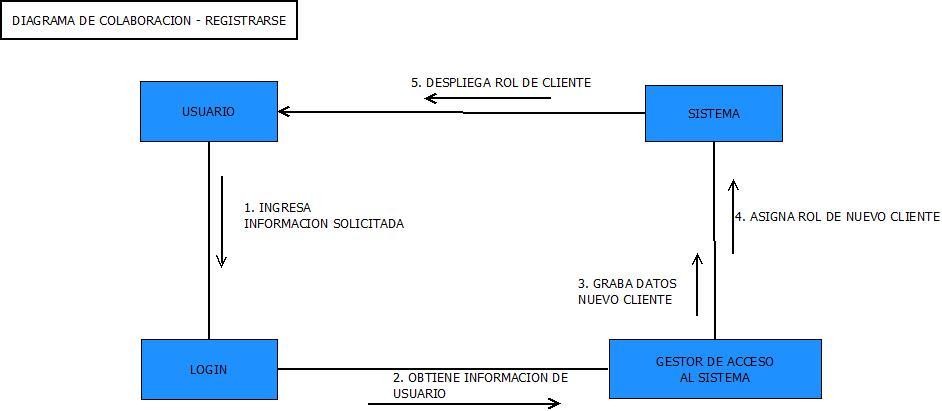
###### DIAGRAMAS DE COLABORACION DEL SISTEMA

* **DIAGRAMA DE COLABORACION DEL MODULO DE SESIONES**



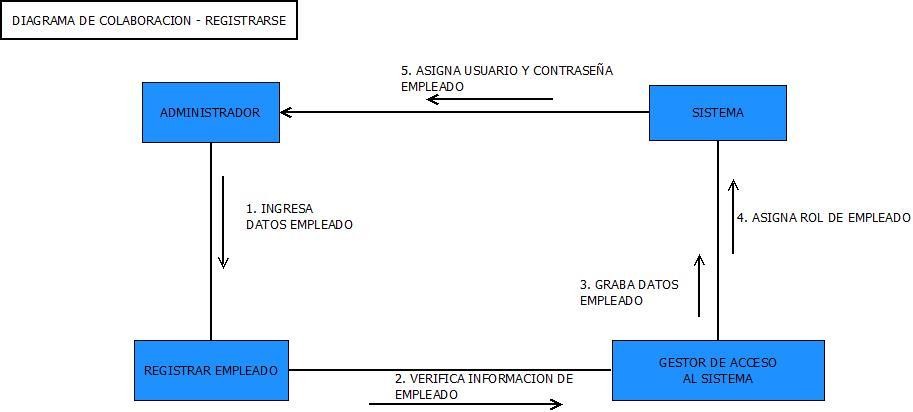
***FIGURA 62****. DIAGRAMA DE COLABORACION – INICIAR SESION*

***FUENTE****. ELABORACION PROPIA*



***FIGURA 63.*** *DIAGRAMA DE COLABORACION – REGISTRARSE*

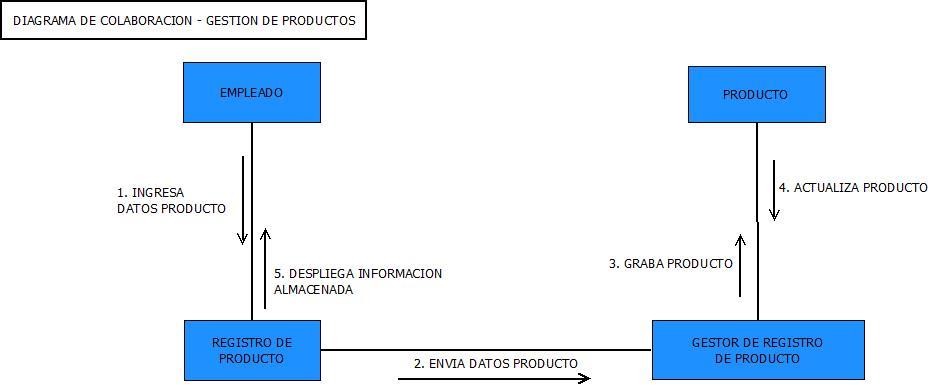
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 64.*** *DIAGRAMA DE COLABORACION – REGISTRARSE*

***FUENTE.*** *ELABORACION PROPIA*

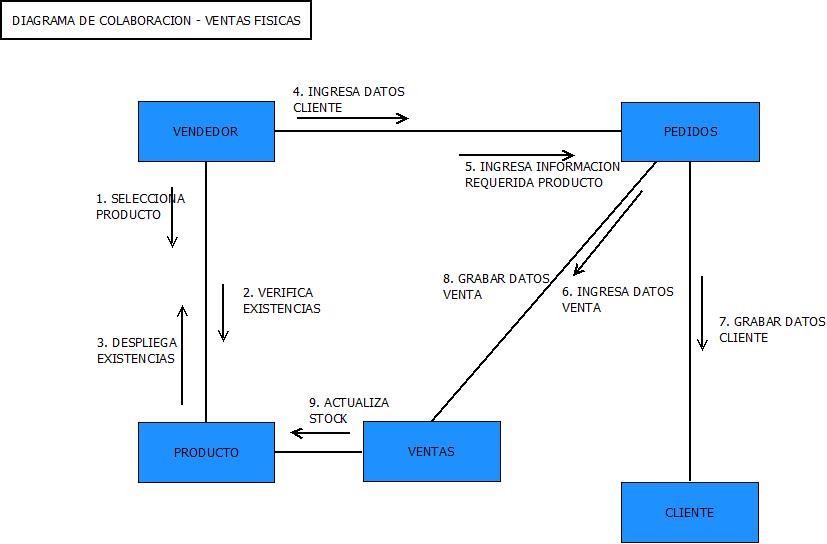
###### DIAGRAMA DE COLABORACION DEL MODULO DE GESTION DE PRODUCTO



***FIGURA 65.*** *DIAGRAMA DE COLABORACION – GESTION DE PRODUCTO*

***FUENTE.*** *ELABORACION PROPIA*

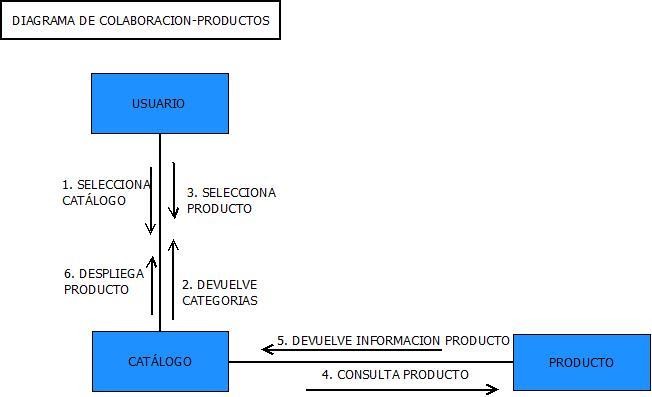
###### DIAGRAMA DE COLABORACION DEL MODULO DE VENTAS FISICAS



***FIGURA 66.*** *DIAGRAMA DE COLABORACION - VENTAS FISICA*

***FUENTE.*** *ELABORACION PROPIA*

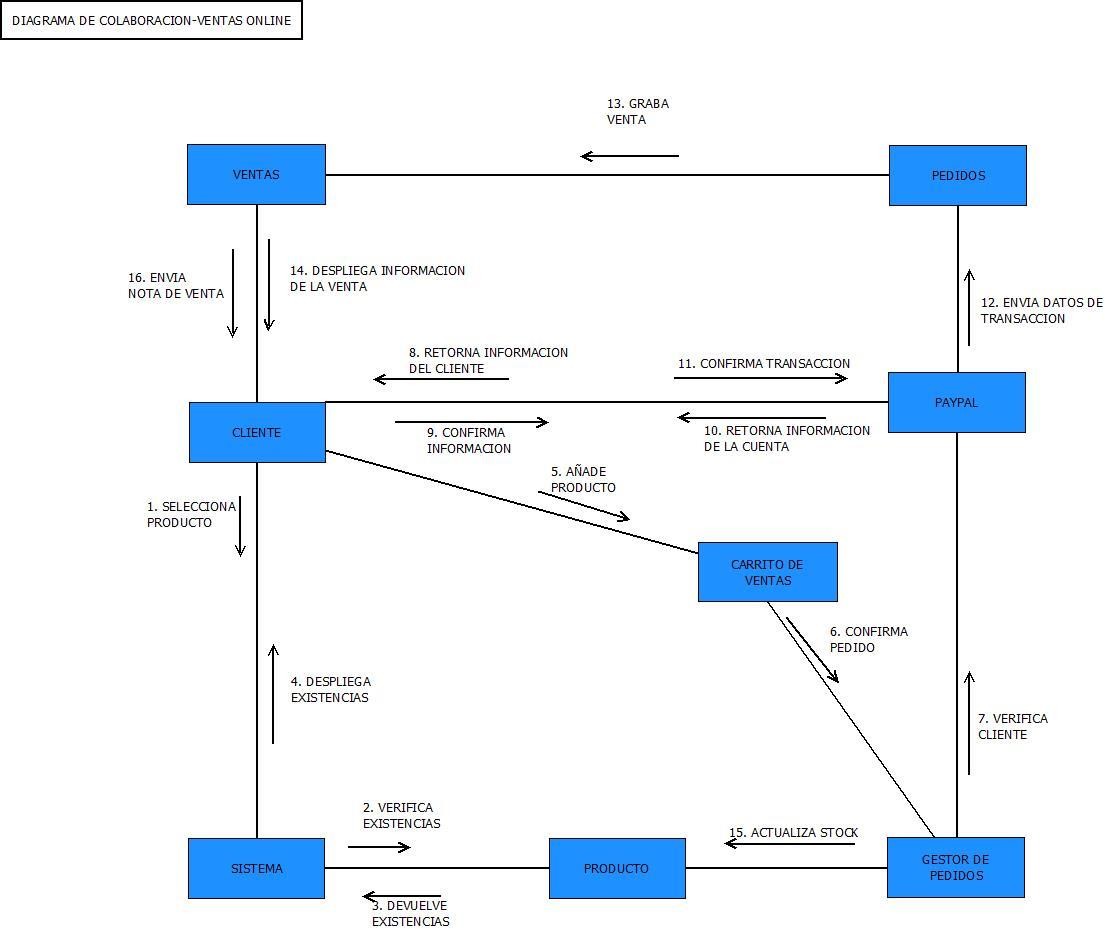
###### DIAGRAMA DE COLABORACION DEL MODULO DE PRODUCTOS



***FIGURA 67.*** *DIAGRAMA DE COLABORACION – PRODUCTOS*

***FUENTE.*** *ELABORACION PROPIA*

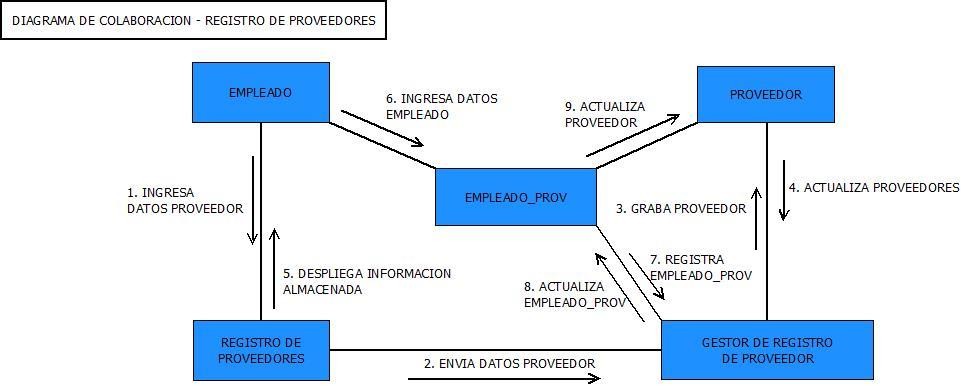
###### DIAGRAMA DE COLABORACION DEL MODULO DE VENTAS ONLINE



***FIGURA 68.*** *DIAGRAMA DE COLABORACION – VENTAS ONLINE*

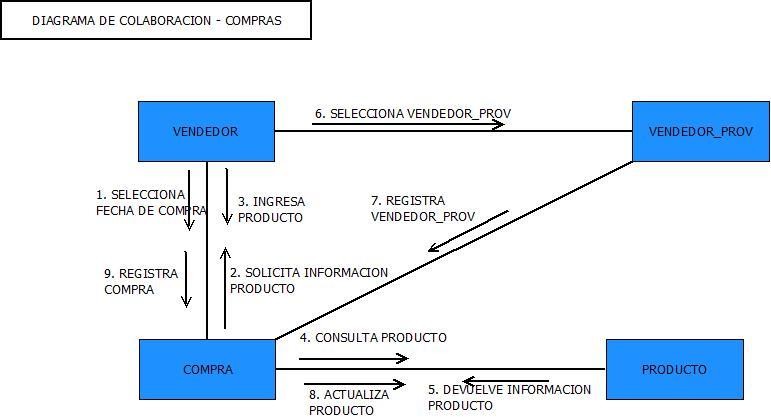
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE COLABORACION DEL MODULO PROVEEDORES Y COMPRAS



***FIGURA 69.*** *DIAGRAMA DE COLABORACION – REGISTRO DE PROVEEDORES*

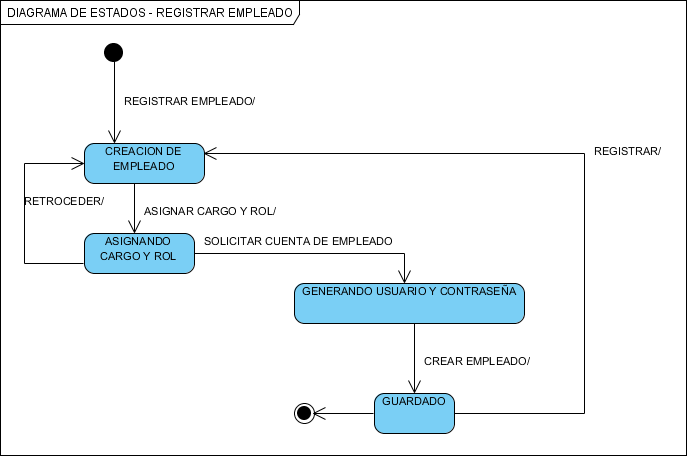
***FUENTE****. ELABORACION PROPIA*



***FIGURA 70.*** *DIAGRAMA DE COLABORACION – COMPRAS*

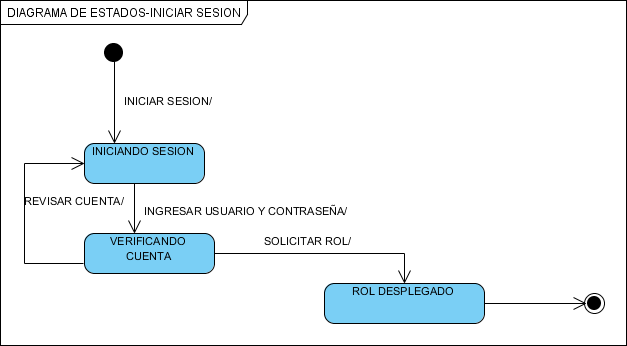
***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL SISTEMA

* **DIAGRAMA DE ESTADOS DEL MODULO DE SESIONES**

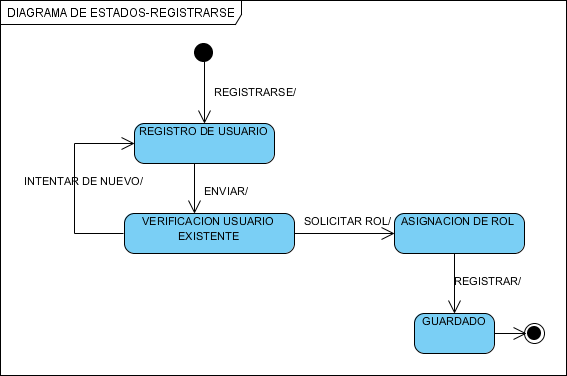
***FIGURA 71.*** *DIAGRAMA DE ESTADOS – REGISTRAR EMPLEADO*

***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 72.*** *DIAGRAMA DE ESTADOS – INICIAR SESION*

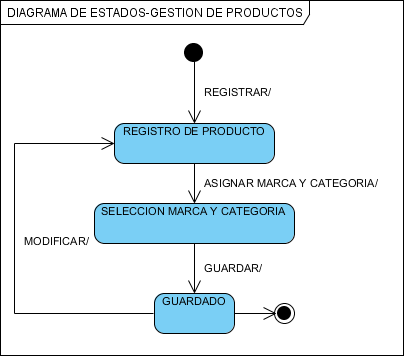
***FUENTE.*** *ELABORACION PROPIA*



***FIGURA 73.*** *DIAGRAMA DE ESTADOS – REGISTRARSE*

***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO DE GESTION DE PRODUCTOS



***FIGURA 74.*** *DIAGRAMA DE ESTA DOS – GESTION DE PRODUCTOS*

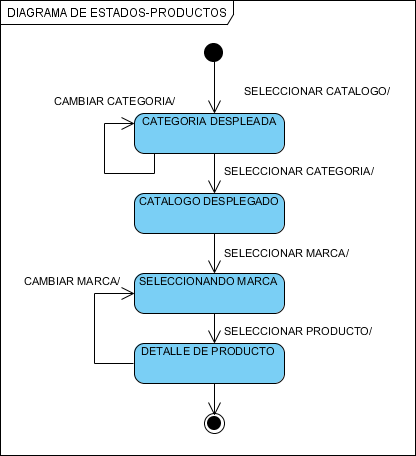
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO DE VENTAS FISICAS

***FIGURA 75.*** *DIAGRAMA DE ESTADOS – VENTAS FISICAS*

***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO DE PRODUCTOS



***FIGURA 76.*** *DIAGRAMA DE ESTADOS – PRODUCTOS*

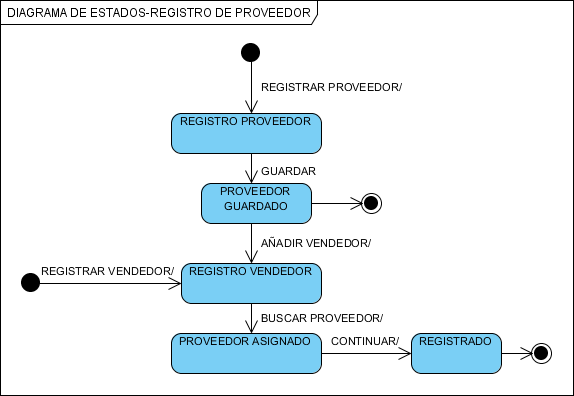
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO DE VENTAS ONLINE

***FIGURA 77.*** *DIAGRAMA DE ESTADOS – VENTAS ONLINE*

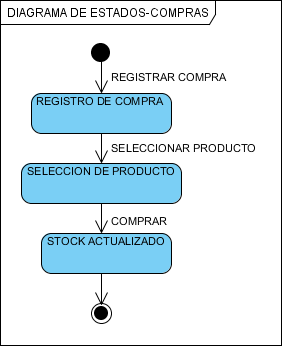
***FUENTE****. ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO DE COMPRAS Y PROVEEDORES



***FIGURA 78****. DIAGRAMA DE ESTADOS – REGISTRO DE PROVEEDOR*

***FUENTE.*** *ELABORACION PROPIA*



**FIGURA 79.** DIAGRAMA DE ESTADOS-COMPRAS

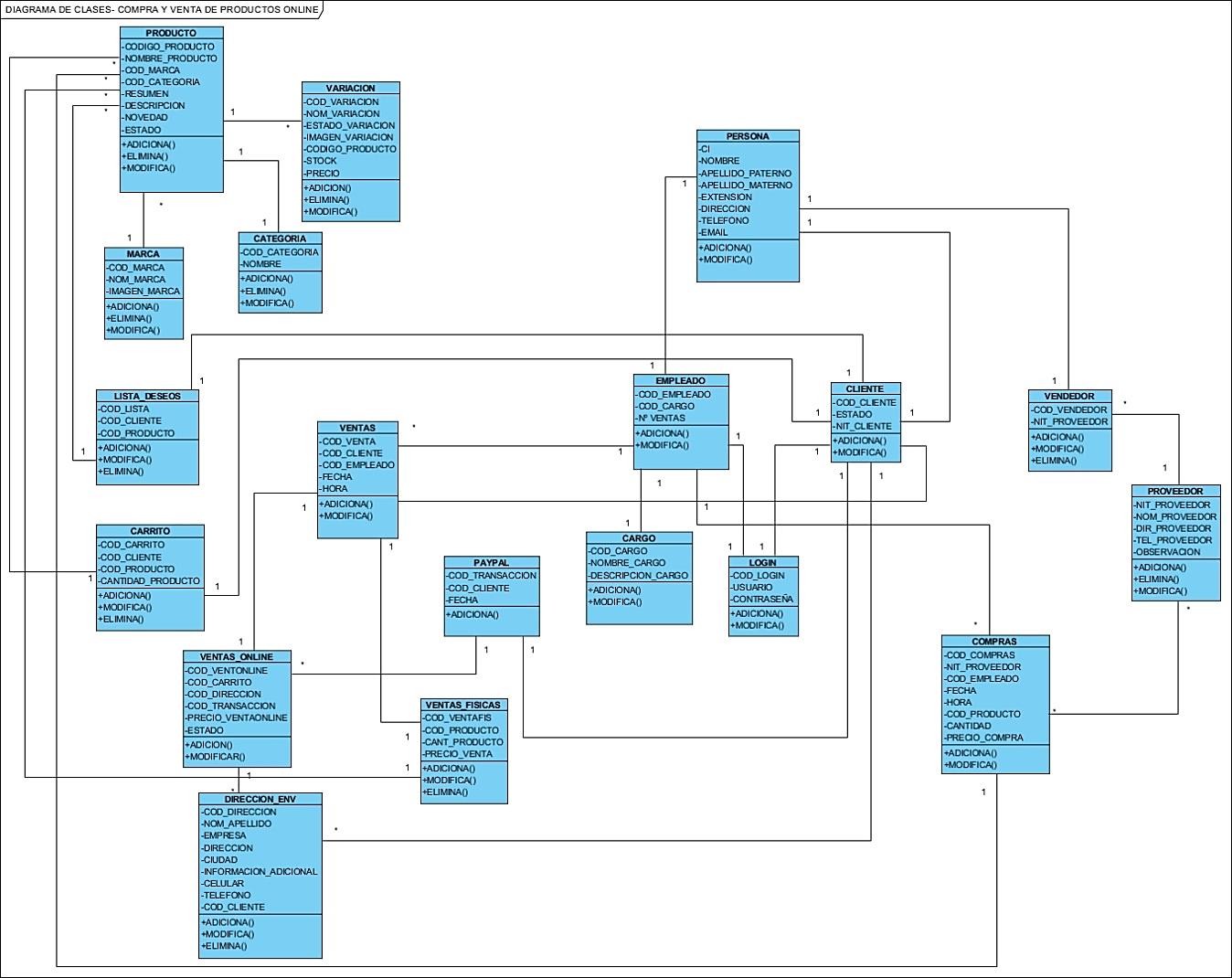
***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE ESTADOS DEL MODULO CHARTS REPORTES

***FIGURA 80.*** *DIAGRAMA DE ESTADOS – CHARTS/REPORTES*

***FUENTE.*** *ELABORACION PROPIA*

###### DIAGRAMA DE CLASES DEL SISTEMA

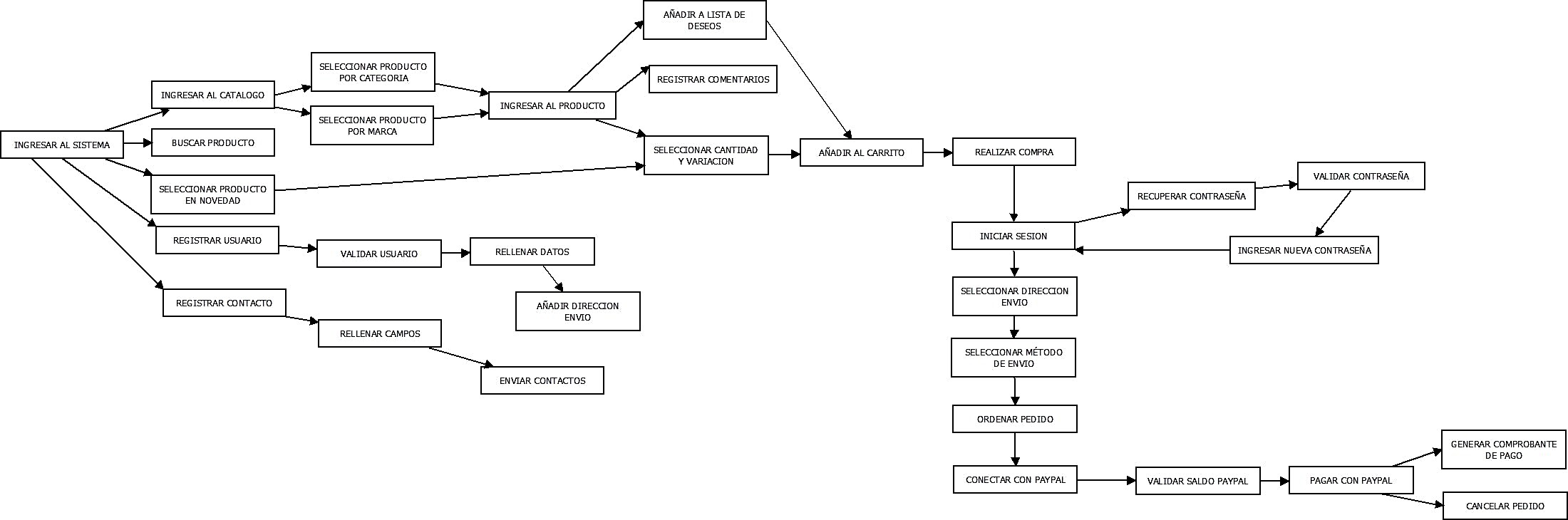


***FIGURA 81.*** *DIAGRAMA DE CLASES- COMPRA Y VENTA DE PRODUCTOS ONLINE*

***FUENTE.*** *ELABORACION PROPIA*

###### MODELO FUNCIONAL

* **VENTAS ONLINE**



***FIGURA 82.*** *MODELO FUNCIONAL – CIENTE*

***FUENTE.*** *ELABORACION PROPIA*

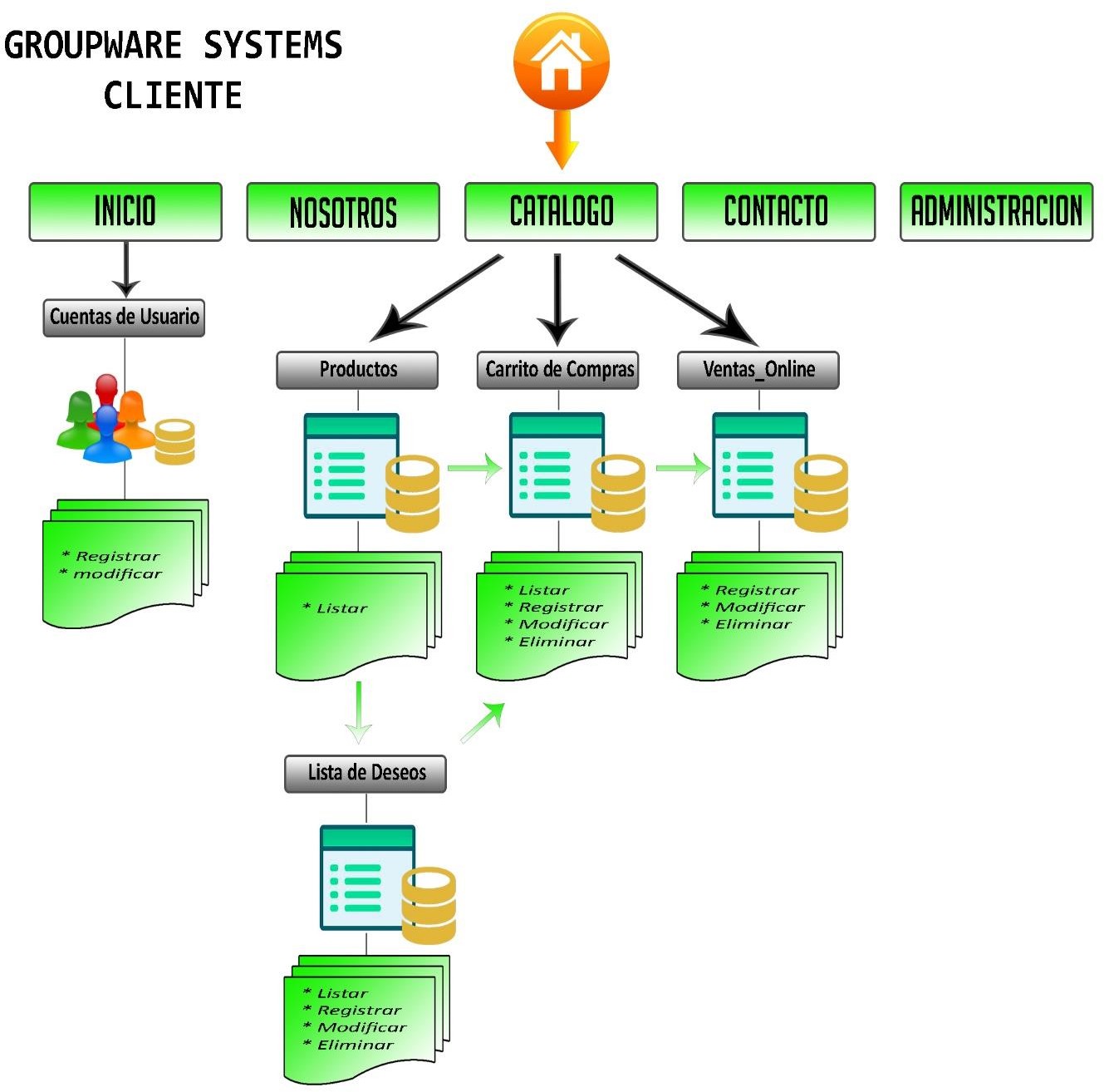
###### C:\Users\Carlitos\Desktop\COMPRAS.jpegCOMPRAS Y VENTAS FISICAS

***FIGURA 83.*** *MODELO FUNCIONAL – COMPRAS Y VENTAS FISICAS*

***FUENTE.*** *ELABORACION PROPIA*

###### MODELO NAVEGACIONAL

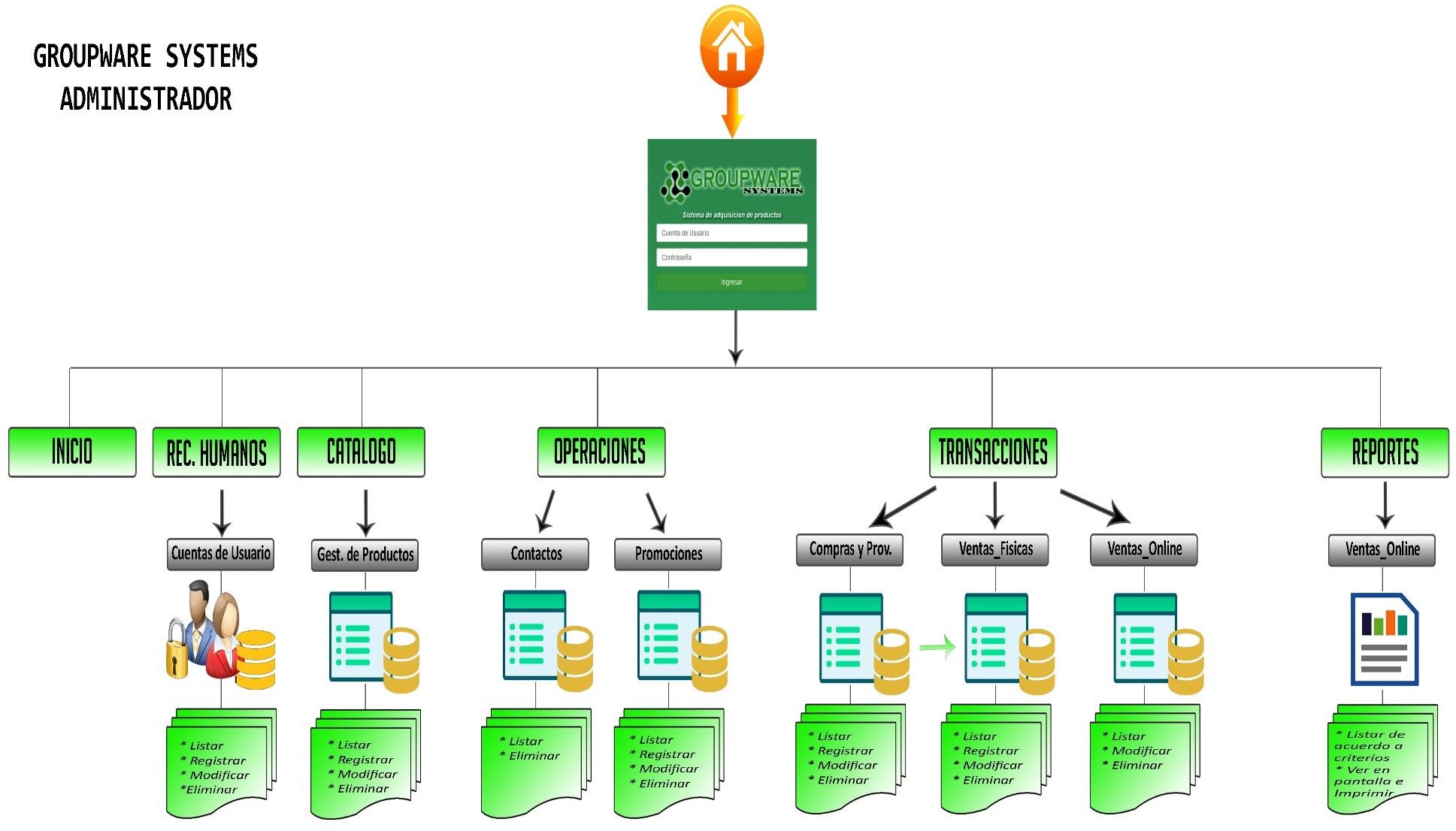
* **CLIENTE**



***FIGURA 84.*** *MODELO NAVEGACIONAL – CLIENTE*

***FUENTE****. ELABORACION PROPIA*

###### ADMINISTRADOR



***FIGURA 85.*** *MODELO NAVEGACIONAL – ADMINISTRADOR*

***FUENTE.*** *ELABORACION PROPIA*

###### ARQUITECTURA DEL SISTEMA DE 3 CAPAS

Se utilizó la arquitectura de 3 capas debido a que el objetivo fundamental es separar la lógica de negocios de la lógica de diseño, es decir, separar la capa de datos de la capa de presentación al usuario.

A continuación se describen cada una de las capas que conforman la arquitectura:

###### CAPA DE PRESENTACIÓN

Se enfoca en las peticiones que realizan los usuarios a través de una interfaz gráfica, las interfaces se encuentran codificadas en las siguientes tecnologías de desarrollo cuando son mostradas al cliente: XHTML, JavaScript, XML y CSS.

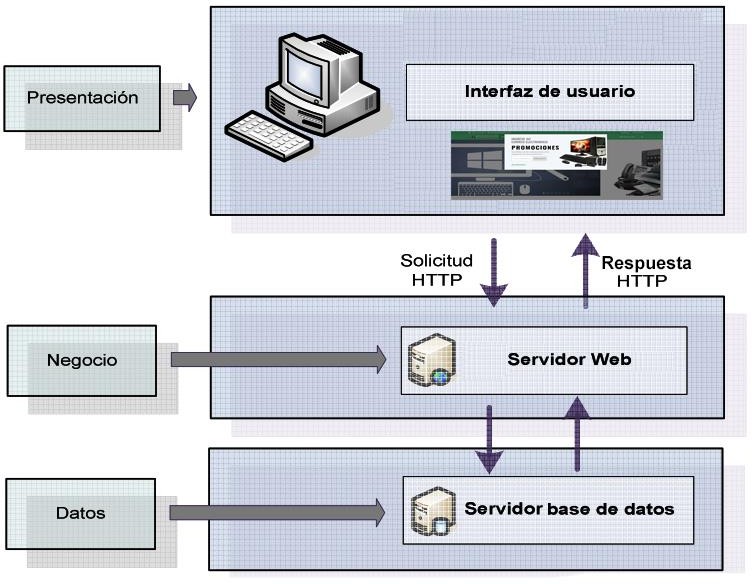
###### CAPA DE NEGOCIO

Abarca todos los procesos de negocio del sistema, es decir toda la lógica de programación, automatización, los procedimientos, funciones, operaciones, etc. S su vez todos estos procesos se encuentran codificados en el Lenguaje de Programación PHP.

###### 4.1.4. CAPA DE DATOS

Agrupa todos los datos y su estructura correspondiente (Base de Datos), los cuales se encuentran administrados por el Sistema Gestor de Base de Datos PostgreSQL.

A continuación se detalla una representación de la arquitectura de 3 capas:



***FIGURA 86.*** *ARQUITECTURA DEL SISTEMA - 3 CAPAS*

***FUENTE.*** *(Afonso & Segnini, 2009)*

###### PRUEBAS DE SOFTWARE

###### PRUEBA DE CAJA NEGRA

Debido a que la metodología de desarrollo de software elegida se basa en subdividir un sistema y entregar al cliente un subproducto totalmente listo para la implementación se optó por realizar pruebas de caja negra con el fin de dar la mejor experiencia posible de manejo al cliente o usuario, la cual se detalla a continuación:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ELEMENTO DEL BACKLOG** | **ENTRADAS CONTROLA DAS** | **SALIDAS ESPERADAS** | **SALIDAS OBTENIDAS** | **CUM PLE** | **NO CUM PLE** |
| REGISTRO DE PERSONAL | El administrador registra al personal de la  empresa | El sistema debe permitir registrar al personal de la empresa | El sistema permite registrar al personal de la  empresa | **100** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CUENTAS DE USUARIOS (EMPLEADOS) | El administrador selecciona la opción de generar usuario y contraseña | El sistema debe asignar un  usuario y  contraseña por defecto a cada empleado. | El sistema asigna un  usuario y contraseña por defecto a cada  empleado. |  |  |
| ASIGNACION DE ROLES(USUARI OS) | El administrador selecciona un tipo de  empleado | El sistema debe permitir asignar un rol a cada empleado de la empresa | El sistema permite asignar un rol a cada  empleado de  la empresa |
| CUENTAS DE USUARIO(CLIE NTE) | El usuario se registra en el sistema, introduciendo sus datos  personales | El sistema debe permitir crear una cuenta de usuario por cliente | El sistema permite crear una cuenta de usuario por cliente |
| AÑADIR, MODIFICAR, ELIMINAR CATEGORIAS | Creación/asig nación/ modificación/ eliminación de categoría | El sistema debe tener la  capacidad de  registrar una  categoría y asignar a cada producto | El sistema tiene la  capacidad de registrar una categoría y  asignar a  cada producto | **100%** |  |
| AÑADIR, MODIFICAR, ELIMINAR MARCAS | Creación/asig nación/ modificación/ eliminación de marca | El sistema debe permitir registrar marcas para cada producto | El sistema permite registrar marcas para cada producto |
| AÑADIR, MODIFICAR, ELIMINAR PRODUCTOS | Creación/modi ficación  /eliminación de producto | El sistema debe permitir registrar productos al catálogo, introduciendo el precio, existe precio de oferta, o diferentes tipos del mismo producto | El sistema permite registrar productos al catálogo, introduciendo el precio, existe precio de oferta, o diferentes tipos del mismo  producto |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ALERTAS TEMPRANAS | Logueado dentro del  módulo de administración | El sistema debe alertar al  empleado de la empresa cuando existan productos insuficientes dentro del catalogo | El sistema alerta al empleado de la empresa cuando existen productos insuficientes dentro del catalogo |  |  |
| VENTAS FISICAS | Escoger Productos y cantidades una vez  logueado el administrador o el vendedor | El sistema debe permitir realizar ventas físicas por parte de los empleados y el administrador del sistema | El sistema permite realizar ventas físicas por parte de los empleados y el  administrador  del sistema | **100%** |  |
| CATALOGO DE PRODUCTOS | Ingreso a la página web | El sistema debe mostrar todos los productos activos al cliente, sin necesidad de iniciar sesión. | El sistema muestra todos los productos activos al  cliente, sin  necesidad de iniciar sesión. | **100%** |  |
| NOVEDADES (PRODUCTOS) | Ingreso a la página web | El sistema debe mostrar los  productos que son novedades y si existen precios de oferta. | El sistema muestra los productos que son novedades y si existen precios de oferta. |
| SELECCIÓN DE PRODUCTOS | Ingreso al  catálogo de productos | El sistema debe mostrar los productos organizados en categorías, y/o  marcas | El sistema muestra los productos organizados en categorías,  y/o marcas |
| DETALLE DE PRODUCTO | Ingresa al producto dentro del  catálogo de productos | El sistema debe mostrar al hacer click sobre un producto el detalle completo del mismo y al | El sistema muestra al hacer click  sobre un  producto el detalle |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | mismo tiempo mostrar productos relacionados a este. | completo del mismo y al mismo tiempo muestra productos relacionados a este. |  |  |
| COMENTARIOS DEL PRODUCTO | Se introduce un comentario del producto | El sistema debe permitir a los clientes generar comentarios dentro de los diferentes productos. Para ver la opinión de cada uno | El sistema permite a los clientes generar comentarios dentro de los diferentes productos.  Para ver la opinión de  cada uno |
| VENTAS ONLINE | Se realiza la compra de varios productos | El sistema debe permitir realizar compras de parte del cliente a través de la  pasarela de pagos de PayPal y guardar dentro de la base,  siempre y cuando el cliente tenga la sesión iniciada. | El sistema permite realizar compras de  parte del  cliente a través de la pasarela de  pagos de  PayPal y guarda dentro de la base de datos, siempre y  cuando el cliente tenga la sesión  iniciada. | **95%** |  |
| CARRITO DE COMPRAS | Se añade los productos al  carrito de compras | El sistema debe permitir eliminar o añadir uno o varios productos al carrito de compras y almacenarlos dentro de la cuenta del cliente | El sistema permite eliminar o añadir uno o varios productos al carrito de  compras y  almacenarlos dentro de la |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | cuenta del cliente |  |  |
| REGISTRO DE DIRECCION DE ENVIO | Se registra la dirección de envió | El sistema debe dar la opción al cliente de grabar una dirección de envió | El sistema da la opción al cliente de  grabar una  dirección de envió |
| REGISTRO DE EMPLEADOS EXTERNOS Y PROVEEDORES | Se registra al proveedor, y a continuación se realiza la asignación del empleado al proveedor | El sistema debe permitir al administrador registrar proveedores y  empleados de los mismos para poder realizar compras de productos | El sistema permite al administrador registrar proveedores y empleados de los mismos  para poder realizar compras de  productos | **98%** |  |
| REGISTRO DE LAS COMPRAS REALIZADAS | Se realiza una compra a un determinado proveedor | El sistema debe permitir registrar las compras de productos realizadas de un determinado proveedor, y para tener un mejor control  debe permitir seleccionar el  vendedor del proveedor que vendió el  producto | El sistema permite registrar las  compras de productos realizadas de un determinado proveedor, y permite seleccionar el vendedor del proveedor que vendió el  producto |
| REPORTES | El administrador introduce los parámetros de productos, ventas, compras para generar un reporte | El sistema debe generar reportes por producto, por compra y venta de los diferentes productos, para saber los  ingresos o | El sistema genera reportes por producto, por compra y venta de los diferentes productos, para saber los | **100%** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | egresos obtenidos | ingresos o egresos obtenidos |  |  |
| GRAFICOS ESTADISTICOS | El administrador introduce los parámetros de productos, ventas, compras para generar el respectivo grafico estadístico para verificar el estado de  los mismos | El sistema de generar gráficos estadísticos para ver el estado de las compras, ventas, y  productos más vendidos. | El sistema genera gráficos estadísticos para ver el estado de las compras, ventas, y productos más vendidos. |
| CONTACTOS | El usuario o cliente envía una consulta al sistema | El sistema debe tener una  sección de  contactos en donde el cliente pueda realizar sugerencias, preguntas o reclamos. | El sistema contiene una sección de contactos en donde el cliente pueda realizar sugerencias, preguntas o  reclamos. | **100%** |  |

***TABLA 15.*** *PRUEBA DE CAJA DE NEGRA DEL SISTEMA*

***FUENTE****. ELABORACION PROPIA*

Para hallar el porcentaje total de la prueba de caja negra se debe realizar el promedio de todos los módulos analizados:

%𝑪𝑵𝑬𝑮𝑹𝑨 = 𝟏𝟎𝟎% + 𝟏𝟎𝟎% + 𝟏𝟎𝟎% + 𝟏𝟎𝟎% + 𝟗𝟓% + 𝟗𝟖% + 𝟏𝟎𝟎% + 𝟏𝟎𝟎%

%𝑪𝑵𝑬𝑮𝑹𝑨 = 𝟗𝟗, 𝟏𝟐𝟓%

Por lo tanto el resultado obtenido muestra el porcentaje de la prueba de caja negra, enfocándose en las necesidades del cliente.

SEGURIDAD

###### METRICAS DE CALIDAD DEL SOFTWARE

El estándar ISO/IEC 9126 define la calidad del software como un conjunto de seis características globales:

* **FUNCIONALIDAD.** Las funciones del software son aquellas que buscan satisfacer las necesidades del usuario/cliente.
* **CONFIABILIDAD.** La capacidad del software de mantener su rendimiento bajo ciertas condiciones durante un período de tiempo específico.
* [**USABILIDAD**](http://www.4rsoluciones.com/blog/que-es-la-usabilidad-web-2/)**.** Basada en la capacidad de entender el manejo y el funcionamiento del *software* por parte del usuario.
* **EFICIENCIA.** Basada en la relación entre el nivel de rendimiento del *software*

y el volumen de recursos utilizado, bajo ciertas condiciones. “FIABILIDAD”

* **CAPACIDAD DE MANTENIMIENTO**. Basada en el esfuerzo necesario para realizar modificaciones específicas.
* **PORTABILIDAD.** Basada en la capacidad del software para ser transferido de un entorno a otro.

El cuidado de estos aspectos durante todo el ciclo de vida del software redundará en productos que no sólo satisfarán las exigencias del usuario, sino que además serán más fáciles de mantener y modificar una vez realizada la entrega al cliente.

###### FUNCIONALIDAD

Es el grado en que el software satisface las necesidades. Los puntos de función (PF) se describen como medidas básicas desde donde calculan métricas de productividad. Los de datos de Puntos de Función se utilizan de dos formas durante la estimación del proyecto de software:

* + - * Como una variable de estimación que se utiliza para dimensionar cada elemento del software
      * Como métricas de línea base recopilada de proyectos anteriores, y utilizados junto con variables de estimación para desarrollar proyección de costo y esfuerzo

Para estimaciones de PF, la descomposición funciona de la siguiente manera:

|  |  |
| --- | --- |
| **DOMINIO DE INFORMACION** | **DESCRIPCION** |
| ***Número de entradas de usuario*** | Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado dentro de un informe no  se cuentan de forma separada |
| ***Número de archivos*** | Se cuenta cada archivo maestro lógico (esto es un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente). |
| ***Número de interfaces externas*** | Se cuenta todas las interfaces legibles por la maquina (por ejemplo: Archivos de datos de disco) que se utilizan para transmitir  información a otros sistemas. |
| ***Número de salidas de usuario*** | Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error. Los elementos  de datos particular |

***TABLA 16.*** *DOMINIOS DE INFORMACION DE PUNTO FUNCION*

***FUENTE.*** (Pressman, 2010)

De acuerdo a la tabla mencionada con anterioridad se tiene los siguientes resultados del Sistema:

|  |  |
| --- | --- |
| **Número de entradas de usuario** | **58** |
| **Número de salidas de usuario** | **70** |
| **Número de peticiones de usuarios** | **35** |
| **Número de archivos** | **6** |
| **Número de Interfaces externas** | **1** |

***TABLA 17.*** *ENTRADAS PARA EL CALCULO DE FUNCIONALIDAD*

***FUENTE.*** *ELABORACION PROPIA*

Los puntos de función se calculan con los datos obtenidos, considerando un factor de ponderación medio:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PARAMETROS** | **CANTIDAD** | **FACTOR DE PONDERACION** | | | **RESULTADO** |
| **DE MEDICION**  **# de entradas de usuario** |  | **SIMPLE** | **MEDIO** | **COMPLEJO** |  |
| 58 | 3 | 4 | 6 | 232 |
| **# de salidas de usuario** | 70 | 4 | 5 | 7 | 350 |
| **# de**  **peticiones** | 35 | 3 | 4 | 6 | 140 |
| **# de archivos** | 6 | 7 | 10 | 15 | 60 |
| **# de Interfaces**  **Externas** | **1** | **5** | **7** | **10** | **7** |
|  |  |  |  | **TOTAL** | **789** |

Entonces:

***TABLA 18.*** *CALCULO DE PUNTOS DE CUENTA TOTAL*

***FUENTE.*** *ELABORACION PROPIA*

## CUENTA TOTAL = 789

Para obtener la sumatoria de todas las respuestas se tiene la siguiente tabla:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Importancia** | **0%** | **20%** | **40%** | **60%** | **80%** | **100%** |
| **Escala** | | No Influencia | Incidental | Moderado | Medio | Significativo | Esencial |
| **Factor** | | 0 | 1 | 2 | 3 | 4 | 5 |
| **1** | ¿Requiere el sistema copias de seguridad y de recuperación fiables? |  |  |  |  | X |  |
| **2** | ¿Se requiere comunicación de datos? |  |  |  |  |  | X |
| **3** | ¿Existen funciones de procesamiento distribuido? |  |  |  | X |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **4** | ¿Es Crítico el rendimiento? |  |  | X |  |  |  |
| **5** | ¿Se ejecuta el sistema en un  entorno operativo existente y fuertemente utilizado? |  |  |  | X |  |  |
| **6** | ¿Requiere el sistema entrada de datos interactiva? |  |  |  |  |  | X |
| **7** | ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas  operaciones? |  |  |  |  | X |  |
| **8** | ¿Se actualizan los archivos maestros de forma interactiva? |  |  |  | X |  |  |
| **9** | ¿Son complejas las entradas, las Salidas o las peticiones? |  |  |  | X |  |  |
| **10** | ¿Es complejo el Procesamiento interno? |  |  |  |  | X |  |
| **11** | ¿Se ha diseñado el código para ser reutilizable? |  |  |  |  | X |  |
| **12** | ¿Están incluidas en le diseño la conversión y la instalación? | X |  |  |  |  |  |
| **13** | ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes  organizaciones? |  |  |  |  | X |  |
| **14** | ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el  usuario? |  |  |  |  | X |  |

Luego:

***TABLA 19.*** *CALCULO DE AJUSTES DE COMPLEJIDAD DEL PUNTO DE FUNCION*

***FUENTE.*** *ELABORACION PROPIA*

###### ΣFi=F1+F2+F3+F4+F5+F6+F7+F8+F9+F10+F11+F12+F13+F14 ΣFi=4+5+3+2+3+5+4+3+3+4+4+0+4+4

Entonces:

# ΣFi=48

Para medir la funcionalidad del software se utiliza la siguiente formula:

##### PF (real)= CUENTA TOTAL\*(0.65+0.01\* Σ (F (i))

Reemplazando en la formula se tiene:

##### PF (real)= 789\*(0.65+0.01\* 48) PF (real)=891.57

Entonces podemos concluir que la funcionalidad del sistema es **ÓPTIMO** de acuerdo al resultado obtenido ya que es mayor a 300 de acuerdo a la tabla de escala de punto de función:

|  |  |
| --- | --- |
| **ESCALA** | **OBSERVACION** |
| **PF>=300** | OPTIMO |
| **200<PF<300** | BUENO |
| **100<PF<200** | SUFICIENTE |
| **PF<100** | DEFICIENTE |

***TABLA 20.*** *ESCALA DE PUNTO DE FUNCION*

***FUENTE.*** *ELABORACION PROPIA*

Sin embargo el resultado que se muestra está representado con un valor de confianza del 65%, para calcular el porcentaje de funcionalidad del sistema se calculara con un valor de confianza de un 100% y se dividirán ambos.

##### PF (esperado) = 789\*(1+0.01\* 48) PF (esperado)=1167.72

El porcentaje del punto de función:

###### %PF=

𝑷𝑭 𝒓𝒆𝒂𝒍

###### =

𝑷𝑭 𝒆𝒔𝒑𝒆𝒓𝒂𝒅𝒐

𝟖𝟗𝟏.𝟓𝟕

𝟏𝟏𝟔𝟕.𝟕𝟐

###### =0,76351

Por lo tanto el sistema desarrollado tiene un 76% de funcionalidad.

###### CONFIABILIDAD

Es la cantidad de tiempo que el software está disponible para su uso, definida por la probabilidad libre de fallos que se encuentra en una escala de 0 a 1.

La confiabilidad se calcula mediante la siguiente función:

(−𝜆∗𝟏𝟐)

𝑭(𝒕) = 𝑭𝒄 ∗ (𝒆 𝟕 )

Donde:

Fc= Funcionalidad del sistema

λ = tasa de fallos en 7 ejecuciones dentro de 1 mes Reemplazando en la formula se tiene:

𝟏

𝑭(𝒕) = 𝟎. 𝟕𝟔 ∗ (𝒆(−𝟕∗𝟏𝟐))

La probabilidad de hallar una falla es la siguiente:

**F (t)=0.14**

Probabilidad de no hallar una falla:

### 1 - F (t)

**1 - 0.14 = 0.86**

Por lo tanto, el sistema presenta una confiabilidad de 0.86, lo que quiere decir que el 86% de las ocasiones el sistema funciona sin presentar fallos y el 14% presentan fallos que no representan un peligro en el sistema.

###### FIABILIDAD

La fiabilidad es fundamental para la reutilización de software, para la obtención de la fiabilidad se aplica la siguiente ecuación:

𝑭𝒊𝒂𝒃𝒊𝒍𝒊𝒅𝒂𝒅 = 𝟏 − (

#𝒆𝒓𝒓𝒐𝒓𝒆𝒔

)

#𝒍𝒊𝒏𝒆𝒂𝒔 𝒅𝒆 𝒄𝒐𝒅𝒊𝒈𝒐

Reemplazando valores se obtiene:

𝟕

𝑭𝒊𝒂𝒃𝒊𝒍𝒊𝒅𝒂𝒅 = 𝟏 − ( )

𝟓𝟔𝟒𝟎

𝑭𝒊𝒂𝒃𝒊𝒍𝒊𝒅𝒂𝒅 = 𝟎. 𝟗𝟗𝟖

Por lo tanto se llega a la conclusión de que el sistema es un 99.8% fiable.

###### FACILIDAD DE USO

La facilidad de uso es el grado en la que el usuario del sistema llega a comprender el funcionamiento del sistema, en esta sección se realizó una calificación de 0 a 100%, de acuerdo al nivel de comprensión del usuario, para tal fin se evaluó a 8 usuarios.

|  |  |  |
| --- | --- | --- |
| **USUARIOS** | **FACILIDAD DE COMPRENSION** | **FACILIDAD DE OPERACIÓN** |
| **USUARIO 1** | 91,00% | 90,00% |
| **USUARIO 2** | 85,00% | 91,00% |
| **USUARIO 3** | 87,00% | 93,00% |
| **USUARIO 4** | 85,00% | 90.5% |
| **USUARIO 5** | 84,00% | 92,00% |
| **USUARIO 6** | 92,00% | 88,00% |
| **USUARIO 7** | 95,00% | 89,00% |
| **USUARIO 8** | 99,00% | 90,00% |
| **PROMEDIO** | 89,75% | 90,43% |
| **TOTAL** | **90,09%** | |

***TABLA 21.*** *RESULTADOS DE LA FACILIDAD DE USO DEL SISTEMA*

***FUENTE.*** *ELABORACION PROPIA*

La facilidad de uso de acuerdo a los resultados obtenidos en la *TABLA 40* es igual a un 90.09% sobre 100%, por lo cual se concluye que el sistema casi en su totalidad es comprensible.

###### CAPACIDAD DE MANTENIMIENTO

La capacidad de mantenimiento es la facilidad con la que una modificación puede realizarse dentro del sistema, ya sea correctiva, de mejora o un mantenimiento preventivo.

Para medir la facilidad de mantenimiento se aplica la siguiente formula:

𝑻𝑪𝑴 = 𝑻𝑨 + 𝑻𝑫 + 𝑻𝑰 + 𝑻𝑷

Donde:

TCM : Tiempo de capacidad de mantenimiento

TA : Tiempo en que se tarda analizar la petición de cambio. TD : Tiempo empleado en diseñar una modificación adecuada. TI : Tiempo necesario para implementar el cambio.

TP : Tiempo en probar y distribuir el cambio a todos los usuarios.

###### MATENIMIENTO CORRECTIVO

El mantenimiento correctivo del sistema fue realizado mientras se estaba desarrollando el sistema, asimismo durante la fase de pruebas, debido a que siempre se presentan algún tipo de fallas.

A continuación se realiza el cálculo de los tiempos de corrección en el sistema:

###### EN EL MEJOR DE LOS CASOS

Los datos expresados se encuentran en minutos.

𝑇𝐶𝑀 = 22 + 46 + 63 + 94

𝑇𝐶𝑀 = 225 𝑚𝑖𝑛𝑢𝑡𝑜𝑠 = 3.75 ℎ𝑜𝑟𝑎𝑠

###### EN EL PEOR DE LOS CASOS

Los datos expresados se encuentran en horas.

𝑇𝐶𝑀 = 7 + 13 + 22 + 68

𝑇𝐶𝑀 = 110 ℎ𝑜𝑟𝑎𝑠

Por lo tanto el promedio de los casos reflejaran el tiempo en que se puede tardar al realizar un mantenimiento correctivo que son: *56.87 horas*.

###### MANTENIMIENTO ADAPTATIVO

El mantenimiento adaptativo es la facilidad con la que el sistema se acomoda a los cambios de su entorno externo, es decir, se adapta a un nuevo sistema operativo, servidor, y sobre todo en las políticas internas de la empresa.

A continuación se realiza el cálculo de los tiempos de corrección en el sistema:

###### EN EL MEJOR DE LOS CASOS

Los datos expresados se encuentran en minutos.

𝑇𝐶𝑀 = 52 + 36 + 48 + 97

𝑇𝐶𝑀 = 233 𝑚𝑖𝑛𝑢𝑡𝑜𝑠 = 3.88 ℎ𝑜𝑟𝑎𝑠

###### EN EL PEOR DE LOS CASOS

Los datos expresados se encuentran en horas.

𝑇𝐶𝑀 = 8 + 24 + 50 + 30

𝑇𝐶𝑀 = 112 ℎ𝑜𝑟𝑎𝑠

Por lo tanto el promedio de los casos reflejaran el tiempo en que se puede tardar al realizar un mantenimiento adaptativo que son: *57.94 horas*.

###### MANTENIMIENTO PREVENTIVO

El mantenimiento preventivo no fue aplicado al sistema, ya que este fue recientemente implementado, al mismo tiempo, la reingeniería de software se aplica a sistemas que ya tienen cierto tiempo de funcionamiento.

###### PORTABILIDAD

Es la capacidad del producto para ser usado en lugar de otro producto software, para el mismo propósito, dentro del mismo las características más relevantes son: la facilidad de instalación, facilidad de ajuste, y la facilidad de adaptación al cambio.

###### FACILIDAD DE INSTALACION

Con el avance y la aparición de nuevas tecnologías, el sistema debe adaptarse a las mismas. El sistema de adquisición de productos online solo tendrá un requisito fundamental a la hora de ser implementado: el servidor que almacenará el sistema deberá soportar la integración de una base de datos PostgreSQL, debido a que el sistema fue desarrollado con una base de datos PostgreSQL por el hecho de que es una gestor de base de datos que no necesita licencia, habiendo cumplido con el requisito fundamental, el sistema está desarrollado para poder ser implementado en cualquier tipo de Sistema Operativo y servidor.

###### FACILIDAD DE AJUSTE

Un ajuste sucede cuando el sistema no cumple con los requerimientos o con los objetivos por el cual fue creado. Los ajustes se realizan en la creación de un nuevo módulo del sistema o el rediseño de los existentes. Un ajuste de todo el sistema se realizará en promedio de 25 días hábiles una vez que haya pasado el sistema el periodo de adecuación.

###### FACILIDAD DE ADAPTACION

El sistema se adaptara a los cambios en los registros de la base de datos y a los cambios que se realizan dentro del mismo, debido a que este es un sitio web

dinámico, y que el 90% de sus funcionalidades son modificables, es decir que el usuario puedo modificar cada una de ellas.

En la siguiente tabla se detalla los valores obtenidos de las métricas de calidad:

|  |  |
| --- | --- |
| **METRICAS DE CALIDAD** | |
| **METRICA** | **VALOR** |
| **FUNCIONALIDAD** | 76% |
| **CONFIABILIDAD** | 86% |
| **FIABILIDAD** | 99,8% |
| **FACILIDAD DE USO** | 90,09% |
| **CAPACIDAD DE MANTENIMIENTO** | 57,94 Hrs. |

***TABLA 22.*** *METRICAS DE CALIDAD DEL SISTEMA*

***FUENTE.*** *ELABORACION PROPIA*

###### RESULTADOS ESPERADOS, APORTES, IMPACTOS Y OPORTUNIDADES

###### APORTES

El aporte del presente proyecto, es el de automatizar los procesos manuales de las compras y ventas para ayudar al personal a tener un mejor control de las mismas, aumentando la productividad de la empresa. El sistema coadyuva la buena toma de decisiones y a la planificación de gestiones posteriores en base a los clientes, elaborando reportes e historiales confiables y precisos hacia el personal que es parte de la empresa.

###### IMPACTO

El sistema ayudara al cliente a poder realizar pagos mediante la pasarela de pagos de Paypal, es decir que el cliente no necesita estar presente en la tienda, las compras lo realiza de manera online (a distancia), asimismo recibe el producto en

la dirección que desee. Permitiendo almacenar todas las transacciones e información procesada dentro de una base de datos, por otro lado permite al empleado de la empresa estar al pendiente del stock de los productos con alertas tempranas, indicando que productos son los faltantes.

La generación de reportes y gráficos estadísticos ayudara a la empresa en la toma de decisiones.

###### OPORTUNIDADES

Con el sistema desarrollado se podrá obtener, un mejor control de las adquisiciones de los productos, asimismo con la venta de los productos online la empresa tendrá un mejor registro de las transacciones, por otra parte debido a que es un sistema online el cliente y empleado, pueden realizar sus actividades dentro del sistema de manera remota.

###### PRESUPUESTO DEL PROYECTO

En esta sección se detalla todo el presupuesto del software, también los gastos que se realizó durante todo el proyecto.

###### MODELO COCOMO

Para la estimación de costos del proyecto se tomaron en cuenta varios factores dentro del modelo Cocomo:

* + - * El sistema es considerado como un proyecto pequeño debido a que contiene menos de 50.000 líneas de código.
      * Se hará uso del modelo Intermedio en el Modo Orgánico, que se utiliza en proyectos pequeños, donde trabajan pequeños equipos, y es más exacto que el modelo básico.

La ecuación que ayudará a hallar el esfuerzo, se detalla a continuación:

𝑬 = 𝒂(𝑲𝑳𝑫𝑪)𝒃 ∗ 𝑭𝑨𝑬

Donde:

E= Esfuerzo

A y b = constantes empíricas KLDC = Kilo de Línea de Código FAE = Factor de ajuste

El factor de ajuste se calcula considerando 15 factores o “guías de coste” que se agrupan en cuatro categorías que se detallara a continuación:

###### CALCULO DE LA VARIABLE FAE

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Valor** | | | | | | |
| **Atributos** | **Muy**  **bajo** | **Bajo** | **Nomi**  **nal** | **Alto** | **Muy**  **Alto** | **Extra**  **Alto** |
| **Atributos de software** | | | | | | |
| **Fiabilidad** | 0,75 | 0,88 | 1,00 | **1,15** | 1,40 |  |
| **Tamaño de base de datos** |  | **0,94** | 1,00 | 1,08 | 1,16 |  |
| **Complejidad** | 0,70 | **0,85** | 1,00 | 1,15 | 1,30 | 1,65 |
| **Atributos de hardware** | | | | | | |
| **Restricciones de tiempo de**  **ejecución** |  |  | **1,00** | 1,11 | 1,30 | 1,66 |
| **Restricciones de memoria**  **virtual** |  |  | **1,00** | 1,06 | 1,21 | 1,56 |
| **Volatilidad de la máquina**  **virtual** |  | 0,87 | **1,00** | 1,15 | 1,30 |  |
| **Tiempo de respuesta** |  | 0,87 | **1,00** | 1,07 | 1,15 |  |
| **Atributos de personal** | | | | | | |
| **Capacidad de análisis** | 1,46 | 1,19 | 1,00 | **0,86** | 0,71 |  |
| **Experiencia en la aplicación** | 1,29 | 1,13 | 1,00 | **0,91** | 0,82 |  |
| **Calidad de los programadores** | 1,42 | 1,17 | **1,00** | 0,86 | 0,70 |  |
| **Experiencia en la máquina**  **virtual** | 1,21 | 1,10 | 1,00 | **0,90** |  |  |
| **Experiencia en el lenguaje** | 1,14 | 1,07 | **1,00** | 0,95 |  |  |
| **Atributos del proyecto** | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Técnicas actualizadas de**  **programación** | 1,24 | 1,10 | **1,00** | 0,91 | 0,82 |  |
| **Utilización de herramientas de**  **software** | 1,24 | 1,10 | 1,00 | **0,91** | 0,83 |  |
| **Restricciones de tiempo de**  **desarrollo** | 1,22 | 1,08 | **1,00** | 1,04 | 1,10 |  |

***TABLA 23.*** *FACTOR DE AJUSTE – COCOMO*

***FUENTE.*** *ELABORACION PROPIA*

Para hallar la variable FAE se debe multiplicar todos los factores

𝑭𝑨𝑬 = 𝟏, 𝟏𝟓 ∗ 𝟎, 𝟗𝟒 ∗ 𝟎, 𝟖𝟓 ∗ 𝟏, 𝟎𝟎 ∗ 𝟏, 𝟎𝟎 ∗ 𝟏, 𝟎𝟎 ∗ 𝟏, 𝟎𝟎 ∗ 𝟎, 𝟖𝟔 ∗ 𝟎, 𝟗𝟏 ∗ 𝟏, 𝟎𝟎 ∗ 𝟎, 𝟗𝟎

∗ 𝟏, 𝟎𝟎 ∗ 𝟏, 𝟎𝟎 ∗ 𝟎, 𝟗𝟏 ∗ 𝟏, 𝟎𝟎

𝑭𝑨𝑬 = 𝟎, 𝟓𝟖𝟖𝟗

###### CALCULO DE LA VARIABLE KLDC

𝑲𝑳𝑫𝑪 = (𝑷𝑭 ∗ 𝑳𝒊𝒏𝒆𝒂𝒔 𝒅𝒆 𝒄𝒐𝒅𝒊𝒈𝒐 𝒑𝒐𝒓 𝒄𝒂𝒅𝒂 𝑷𝑭)/𝟏𝟎𝟎𝟎

#### 𝑲𝑳𝑫𝑪 = 𝟒, 𝟔𝟒𝟎

Hallando el esfuerzo se tiene:

𝑬 = 𝟑, 𝟐 ∗ (𝟒, 𝟔𝟒𝟎)𝟏,𝟎𝟓 ∗ 𝟎, 𝟓𝟖𝟖𝟗

𝑬 = 𝟗, 𝟒𝟒 𝒑𝒆𝒓𝒔𝒐𝒏𝒂𝒔/𝒎𝒆𝒔

Hallando el tiempo de desarrollo:

𝑻 = 𝒄 ∗ 𝑬𝒅

𝑻 = 𝟐, 𝟓 ∗ 𝟗, 𝟒𝟒𝟎,𝟑𝟖

𝑻 = 𝟓, 𝟖𝟕

Por lo tanto el software se desarrollara en 6 meses.

Para saber el personal promedio del proyecto se realiza la siguiente ecuación:

𝑬

𝑷 =

𝑻

𝑷 =

𝟗, 𝟒𝟒

𝟓, 𝟖𝟕

𝑷 = 𝟏, 𝟔𝟏

Según los resultados se necesitará un equipo de 2 personas trabajando por 6 meses. Por lo tanto el personal promedio del proyecto son 2 personas.

Hallando la productividad:

𝑷𝑹 = 𝑳𝑫𝑪/𝑬

𝟒𝟔𝟒𝟎

𝑷𝑹 =

𝟗. 𝟒𝟒

𝑷𝑹 = 𝟒𝟗𝟏, 𝟓𝟐 𝑳𝒊𝒏𝒆𝒂𝒔 𝒅𝒆 𝒄𝒐𝒅𝒊𝒈𝒐 𝒑𝒐𝒓 𝒑𝒆𝒓𝒔𝒐𝒏𝒂/𝒎𝒆𝒔

Según los resultados obtenidos será necesario contar con un equipo de 2 personas trabajando alrededor de 6 meses. Asimismo los resultados se enmarcan dentro de los plazos establecidos por la empresa.

###### ANALISIS DE COSTOS

###### COSTO DEL SOFTWARE

El costo del sistema se define mediante la siguiente formula:

𝑪𝒔𝒊𝒔 = 𝑷 ∗ 𝑻 ∗ 𝑺𝒑

Donde:

Csis = Costo del sistema P = Personal

T = Tiempo

Sp= Sueldo Promedio (en Bolivia) Reemplazando datos:

𝑪𝒔𝒊𝒔 = 𝟏, 𝟔𝟏 ∗ 𝟓, 𝟖𝟕 ∗ 𝟏𝟗𝟎𝟎

𝑪𝒔𝒊𝒔 = 𝟏𝟐. 𝟐𝟔𝟖, 𝟑𝟎 𝑩𝒔 (𝑩𝒐𝒍𝒊𝒗𝒊𝒂𝒏𝒐𝒔)

A partir del costo del sistema se obtiene el costo total de desarrollo de software, que se detalla a continuación:

|  |  |
| --- | --- |
| **COSTO DEL SOFTWARE** | **PRECIO** |
| **Sistema** | 12.268,3 Bs |
| **Licencia del Lenguaje de Programación (PHP)** | 0,00 Bs |
| **Licencia del Sistema Gestor de Base de Datos**  **(PostgreSQL)** | 0,00 Bs |
| **Alquiler de Hosting (Anual)** | 200,00 Bs. |
| **Costo Total del Software en Bolivianos** | **12.468,30 Bs** |
| **Costo Total del Software en Dólares**  *(cambio del dólar actual 6,97Bs)* | **1.781,19 $us** |

***TABLA 24****. COSTO TOTAL DEL SOFTWARE*

***FUENTE****. ELABORACION PROPIA*

De acuerdo a los datos obtenidos podemos decir que el costo total del software (CSW) es:

𝑪𝑺𝑾 = 𝟏𝟐. 𝟒𝟔𝟖, 𝟑𝟎 𝑩𝒔.

###### COSTO DEL HARDWARE

Debido a que el presente proyecto es un sistema web, no se necesita de un computador con altas prestaciones, sin embargo, se necesitará el equipo la administración de las ventas y los productos, y al mismo tiempo se debe cumplir con los requerimientos de los navegadores web, para tener una mejor experiencia en el manejo del sistema web, por tal motivo se hace referencia a un computador con procesador Intel core i3:

|  |  |  |
| --- | --- | --- |
| **REQUISITOS** | **DESCRIPCIÓN** | **COSTO** |
| **MICROPROCESADOR** | Intel Core i3 6ta generación | 118 $us |
| **TARJETA MADRE** | ASUS H81M | 55 $us |
| **MEMORIA** | 4 GB DDRIII | 23 $us |
| **VIDEO** | 2 GB GT610 | 42 $us |

|  |  |  |
| --- | --- | --- |
| **DISCO DURO** | 500 GB Sata | 45 $us |
| **QUEMADOR DE DVD** | LG 24X | 16 $us |
| **CASE COMBO** | Delux | 53 $us |
| **MONITOR** | 19" Samsung | 78 $us |
| **IMPRESORA** | Epson L210 | 279 $us |
| **COSTO TOTAL HARDWARE EN DÓLARES** | | **709,00 $us** |
| **COSTO TOTAL HARDWARE EN BOLIVIANOS** | | **4.963,00 Bs** |

***TABLA 25.*** *COSTO TOTAL DEL HARDWARE*

***FUENTE****. ELABORACION PROPIA*

Por lo tanto, podemos decir que el costo total del hardware (CHW) es:

𝑪𝑯𝑾 = 𝟒. 𝟗𝟔𝟑, 𝟎𝟎 𝑩𝒔.

###### COSTO DE LA INVESTIGACION

El costo de la investigación se refiere al material invertido durante toda la investigación, como ser: papelería, impresiones, fotocopias, etc. Se detalla a continuación:

|  |  |
| --- | --- |
| **MATERIAL** | **COSTO** |
| **PAPELERIA** | 170 Bs |
| **PASAJES** | 150 Bs |
| **IMPRESIONES** | 250 Bs. |
| **FOTOCOPIAS** | 80 Bs. |
| **COSTO TOTAL MATERIAL EN**  **BOLIVIANOS** | **650,00 Bs.** |
| **COSTO TOTAL DEL MATERIAL EN DOLARES** | **92, 86 $us.** |

***TABLA 26.*** *COSTO TOTAL DE LA INVESTIGACION*

***FUENTE.*** *ELABORACION PROPIA*

Por lo tanto, el costo total de la investigación (CINV) es:

𝑪𝑰𝑵𝑽 = 𝟔𝟓𝟎𝑩𝒔

###### COSTO TOTAL DEL PROYECTO

De acuerdo a los valores obtenidos anteriormente, se determinará el costo total del proyecto, para ello tenemos la siguiente ecuación matemática.

Donde:

𝑪𝑻𝑷 = 𝑪𝑺𝑾 + 𝑪𝑯𝑾 + 𝑪𝑰𝑵𝑽

CTP = Costo Total del Proyecto CSW = Costo del Software CHW = Costo del Hardware

CINV = Costo de la Investigación Reemplazando en la ecuación tenemos lo siguiente:

𝑪𝑻𝑷 = 𝟏𝟐. 𝟒𝟔𝟖, 𝟑𝟎 𝑩𝒔. +𝟒. 𝟗𝟔𝟑, 𝟎𝟎 𝑩𝒔. +𝟔𝟓𝟎, 𝟎𝟎 𝑩𝒔.

𝑪𝑻𝑷 = 𝟏𝟖. 𝟎𝟖𝟏, 𝟑𝟎 𝑩𝒔. ≈ 𝟐. 𝟓𝟖𝟑, 𝟎𝟎 $𝒖𝒔 [𝑫𝒐𝒍𝒂𝒓𝒆𝒔]

Por lo tanto el Costo Total del Proyecto es de ***18.081,30 Bs***.

###### RESUMEN DEL PRESUPUESTO

En esta sección se detalla en una tabla el presupuesto total del proyecto.

|  |  |
| --- | --- |
| **COSTO DEL SOFTWARE** | |
| **DETALLE** | **COSTO** |
| **Costo total de software en bolivianos** | **12.468,3 Bs.** |
| **Costo total de software en dólares** | **1.781,19$us.** |
| **COSTO DEL HARDWARE** | |
| **Costo total de hardware en bolivianos** | **4.963,00 Bs.** |
| **Costo total de hardware en dólares** | **709,00 $us.** |
| **COSTO DE LA INVESTIGACIÓN** | |

|  |  |
| --- | --- |
| **Costo total de investigación en bolivianos** | **650,00 Bs.** |
| **Costo total de investigación en dólares** | **92,86 $us.** |
| **COSTO DE SERVICIOS BÁSICOS** | |
| **Costo total de servicios básicos en bolivianos** | **500,00 Bs.** |
| **Costo total de servicios básicos en dólares** | **71,43 $us.** |
| **COSTO TOTAL** | |
| **COSTO TOTAL EN BOLIVIANOS** | **18.581,30 Bs.** |
| **COSTO TOTAL EN DÓLARES** | **2.654,47 $us.** |

***TABLA 27.*** *RESUMEN DEL PRESUPUESTO TOTAL DEL PROYECTO*

***FUENTE****. ELABORACION PROPIA*

###### CALCULO DEL VAN Y EL TIR

###### VALOR ACTUAL NETO (VAN)

Es un procedimiento que permite calcular el valor actual en cantidad monetaria a partir de determinados flujos de caja futuros, originados por una inversión inicial, refleja la diferencia entre el valor actual de los cobros menos el valor actual de los pagos. En términos generales se puede interpretar el VAN del modo siguiente:

VAN > 0 => Que la empresa genera beneficio

VAN = 0 => No hay beneficio ni pérdidas, aunque se pierde el tiempo VAN < 0 => hay pérdidas en la empresa, además de perder el tiempo.

Para el cálculo de este valor se tiene la siguiente formula:

𝑛

#### VAN = −𝐼𝑜 + ∑ 𝐹𝑡

(1 + 𝑟)𝑡

𝑡=1

Donde:

n= Cantidad de meses que se proyectó para el proyecto

F= Flujo de caja (mensual) r= Tasa de retorno

Io= Inversión inicial

T = tiempo (expresado en meses)

Reemplazando los datos en la formula con una tasa de retorno del 7% definido por el Estado Plurinacional de Bolivia se obtiene lo siguiente:

Para el primer mes:

𝑉𝐴 = 18.500/(1 + 0,7)1 = 18.500/1,7 = 10.882,35

Para el segundo mes:

𝑉𝐴 = 18.500/(1 + 0,7)2 = 18.500/2,89 = 6.401,38

Para el tercer mes:

𝑉𝐴 = 18.500/(1 + 0,7)3 = 18.500/4,913 = 3.765,52

Para el cuarto mes:

𝑉𝐴 = 18.800/(1 + 0,7)4 = 18.800/8,3521 = 2.250,93

Para el quinto mes:

𝑉𝐴 = 19.000/(1 + 0,7)5 = 19.000/14,19857 = 1.338,16

Para el sexto mes:

𝑉𝐴 = 19.000/(1 + 0,7)6 = 19.000/24,1376 = 787,15

Para el séptimo mes:

𝑉𝐴 = 20.000/(1 + 0,7)7 = 20.000/41,0339 = 487,4

Entonces:

𝑉𝐴𝑁 = 10.882,35 + 6.401,38 + 3.765,52 + 2.250,93 + 1.338,16 + 787,15 + 487,4

= 25.912,89

𝑉𝐴𝑁 = 25.912,89 − 18.581,30 = 7.331,59

Por lo tanto tendremos:

#### 𝑽𝑨𝑵 = 𝟕. 𝟑𝟑𝟏, 𝟓𝟗 > 𝟎

El resultado del VAN positivo (mayor a cero) indica que el proyecto puede aceptarse puesto que producirá ganancias por encima de la rentabilidad exigida.

###### TASA INTERNA DE RETORNO (TIR)

La Tasa Interna de Retorno (también conocida como Tasa Interna de Rentabilidad) Es otro criterio utilizado para la toma de decisiones sobre los proyectos de inversión y financiamiento. Es la tasa de interés que, utilizada en el cálculo del Valor Actual Neto, hace que este sea igual a 0.

𝑛

#### TIR = ∑ 𝐹𝑡

(1 + 𝑟)𝑡

#### − 𝐼𝑜 = 0

𝑡=1

Donde:

n= Cantidad de meses que se proyectó para el proyecto F= Flujo de caja (mensual)

r= Tasa de retorno Io= Inversión inicial

T = tiempo (expresado en meses)

Como el cálculo del VAN con un 7% de tasa de retorno es positivo el valor del mismo se incrementará con la finalidad de obtener un valor cercano a cero.

10.882,35

6.401,38

3.765,52

2.250,93

1.338,16

787,15

(1 + 0.17)1 + (1 + 0.17)2 + (1 + 0.17)3 + (1 + 0.17)4 + (1 + 0.17)5 + (1 + 0.17)6

487,4

+ (1 + 0.17)7 − 18.581,30 = 𝟎

𝑇𝐼𝑅 = 18.609,35 − 18.581,30

𝑇𝐼𝑅 = 28,05

El resultado 28,05 es el que más se aproxima a 0, tendremos un:

#### 𝑻𝑰𝑹 = 𝟏𝟕%

###### SEGURIDAD DEL SOFTWARE

Las características de seguridad con las que cuenta el sistema son:

|  |  |  |  |
| --- | --- | --- | --- |
| **REF.** | **FUNCION** | **ATRIBUTO** | **DETALLES** |
| **1** | **Realización de backup del**  **sistema** | Seguridad | Realiza un backup semanalmente de toda la base de datos |
| **2** | **Contraseña Segura** | Seguridad | Validar el uso de contraseñas que tengan letras minúsculas, mayúsculas, números y caracteres  especiales |
| **3** | **Cifrado de contraseña** | Seguridad | Se cifra la contraseña con MD5 para  luego almacenarla dentro de la base de datos siendo los datos seguros. |
| **4** | **Sesiones y Roles** | Seguridad | El administrador asigna un nombre de usuario y contraseña a cada empleado dando un rol de acceso y los datos no sean públicos a  cualquier persona. |
| **5** | **Validación de Usuarios** | Seguridad | El administrador no puede crear varios usuarios con los mismos datos, asimismo el un Cliente no puede registrarse en el sistema N veces, solo lo podrá hacer una única  vez. |

***TABLA 28.*** *SEGURIDAD DEL SOFTWARE*

***FUENTE.*** *ELABORACION PROPIA*

## CONCLUSIONES

Se desarrolló un sistema de información web que realiza el comercio

electrónico, que promociona y efectúa la venta de productos que posee

GROUPWARE SYSTEMS, utilizando la metodología SCRUM para obtener módulos

del sistema capaces de poner en funcionamiento al término de cada sprint y no demorar demasiado en el desarrollo del mismo, reduciendo así los tiempos de entrega.

Las conclusiones son:

* Se analizó los requerimientos de la empresa determinando las necesidades del cliente con el uso de entrevistas, historias de usuario, realizando estimaciones de acuerdo al método *planning póker* de SCRUM medidas en *story points. Ver Capítulo V, acápite 5.1.2.1.*
* Se diseñó el sistema de adquisición de productos definiendo los módulos y funciones necesarias que albergará el sistema con el uso de la herramienta de modelado UML, al mismo tiempo haciendo uso de dos modelos como ser: el modelo funcional y navegacional para tener una mejor comprensión del funcionamiento del sistema. *Ver Capítulo V, acápite 5.1.2.13.*
* Se desarrolló el sistema web mediante el uso de herramientas Open Source como ser el Lenguaje de Programación PHP, el Sistema Gestor de Base de datos PostgreSQL, para la adquisición online de productos y registrar las compras y transacciones efectuadas. *Consultar con Manual de Usuario (Anexos)*
* Se implementó el sistema de información web dentro de la empresa automatizando el proceso de adquisición de productos, facilitando al cliente

y a la empresa en controlar y registrar las transacciones que se realiza por parte de ambos.

* Se Realizó pruebas del sistema web con el uso de las pruebas de caja negra, verificando la aceptación por parte del cliente y la empresa, solucionando algunas fallas que el software pudo tener. *Ver Capítulo V, acápite 5.3.*
* Se capacitó al personal de la empresa con el fin de que el usuario tenga una mejor experiencia y haga un mejor uso del sistema, realizando demostraciones del sistema, en diferentes escenarios que se pueden presentar.

## RECOMENDACIONES

Para un futuro que garantice la continuidad del sistema se tiene las siguientes recomendaciones:

*En primer lugar agregar un sistema contable para manejar de mejor manera las cuentas de la empresa.

*Se recomienda ampliar el “Sistema de Información web, Comercio Electrónico para la Adquisición de Productos Online” con un módulo de facturación, y un módulo de gestión de inventarios, siempre y cuando las empresas que trabajen con GROUPWARE SYSTEMS acepten el tipo de facturación electrónica.

*Por otra parte se recomienda realizar un mantenimiento correctivo y preventivo al sistema, de acuerdo a las necesidades de la empresa.

## BIBLIOGRAFÍA

Pérez Medina, J. L., & Sánchez, I. C. (2012). HACIA LA EXTENSION DEL

MÉTODO GRAY WATCH. *Publicaciones en Ciencias y Tecnología*, 19.

Afonso, M. A., & Segnini, J. E. (2009). *DESARROLLO DE UN SISTEMA*

*AUTOMATIZADO BAJO ENTORNO WEB.* Barcelona, España: Universidad de Oriente Núcleo Anzoátegui.

Aguilera López, P. (2010). *SEGURIDAD INFORMATICA.* España: Editex.

Alvarez, M. A. (2010). *MANUAL DE CODEIGNITER.* España: DesarrolloWeb.com. Barranco de Areba, J. (2001). *METODOLOGÍA DEL ANÁLISIS ESTRUCTURADO*

*DE SISTEMAS.* Madrid, España: Universidad Pontifica Comillas ICA-

ICADE.

Berenguel Gómez, J. L. (2016). *DESARROLLO DE APLICACIONES WEB EN EL*

*ENTORNO SERVIDOR.* Madrid, España: Paraninfo.

Booch, G., Rambaugh, J., & Jacobson, I. (2004). *THE UNIFIED MODELING LANGUAGE REFERENCE MANUAL.* United States: Addison Wesley.

Caceres, A. E. (2014). *ANALISIS Y DISEÑO DE SISTEMAS DE INFORMACION.*

Capuñay Uceda, O. (2013). *DESARROLLO WEB CON PHP : Aprende PHP paso a paso.* Perú: Ocapunay.

Clements, P. (1996). A SURVEY OF ARCHITECTURE DESCRIPTION

LANGUAGES. *Proceedings of the International Workshop on Software Specification and Design*, 16 .

Cohen, D., & Asin, E. (2000). *SISTEMA DE INFORMACION PARA LOS*

*NEGOCIOS, Un enfoque para la toma de decisiones* (Tercera ed.). Mexico: McGRAW-HILL/INTERAMERICANA EDITORES.

Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2012). *SCRUM PRIMER: Una introducción básica a la teoría y práctica de Scrum.* United States: Scrum Primer.

Denzer, P. (2002). *MANUAL NO OFICIAL - PostgreSQL.* España: U.T.F.S.M. Díaz, A. M. (2013). *ROPUESTA DE UN MÉTODO PARA EL DISEÑO DE VISTAS*

*ARQUITECTURALES DE UN SISTEMA BAJO UN ENFOQUE DE*

*CALIDAD.* Barquisimeto, Venezuela: Universidad Centroccidental “Lisandro Alvarado”.

Espinoza, D. (2007). *MULTIMEDIA RESPETEN LOS DERECHOS DEL NIÑO.*

Ciudad de La Habana, Cuba: Universidad de la Ciencias Informaticas.

Fernandez Alarcon, V. (2006). *DESARROLLO DE SISTEMA DE INFORMACION:*

*Una metodologia basada en el modelado.* España: EDICIONES UPC.

Fernández Romero, Y., & Díaz Gonzáles, Y. (2012). PATRON MODELO VISTA CONTROLADOR. *REVISTA DIGITAL DE LAS TECNOLOGIAS DE LA INFORMACION Y LAS COMUNICACIONES: Telematica*, 57.

Finlay, P. N. (1994). *INTRODUCING DECISION SUPPORT SYSTEMS.* Oxford,

United States: Blackwell Publishers.

Fournier C., M. d. (1992). *RESEÑA DE "DISEÑO DE SISTEMAS DE INFORMACION" DE JOHN G. BURCH Y GARY GRUDNITSKI.* Mexico: UNIVERSIDAD AUTÓNOMA METROPOLITANA UNIDAD XOCHIMILCO.

Fowler, M. (1999). *REFACTORING: Improving the Design of Existing Code.*

Massachusetts, United States: Addison-Wesley.

Garcia, F. J. (2008). *INGENIERIA DE SOFTWARE:Diseño Orientado a Objetos.*

Salamanca, España: Creative Commons.

Gárcia-Cervigón Hurtado, A., & Alegre Ramos, M. (2011). *SEGURIDAD INFORMATICA.* Madrid, España: Paraninfo, S.A.

*ISO 25000*. (2015). Obtenido de <http://iso25000.com/index.php/normas-iso-> 25000/iso-25010

Kendall, E. K., & Kendall, E. J. (2005). *ANALISIS Y DISEÑO DE SISTEMAS.*

Mexico: PEARSON EDUCATION.

Larman, C. (2006). *UML Y PATRONES.* Madrid, España: Pearson Education. Laudon, C. K., & Laudon, P. J. (2012). *SISTEMAS DE INFORMACION*

*GERENCIAL.* Mexico: PEARSON EDUCATION.

Mantilla Garcia, E. (2015). *Los Servicios de Pago, su regulacion y desarrollo en la Era Digital.* Madrid, España: Universidad CEU San Pablo.

Marroquín, N. (2010). *TRAS LOS PASOS DE UN HACKER.* Quito, Ecuador: NMC Research Cía Ltda.

Medina, J. (2005). *METODOLOGÍA Y HERRAMIENTAS UML PARA EL*

*MODELADO Y ANÁLISIS.* Santander: Tesis Doctoral.

Muller, P. A. (1997). *INSTANT UML.* Birmingham, England: Wrox Press. Palacio, J. (2015). *SCRUM MANAGER I: Las Reglas de Scrum.* Zaragoza,

España: Safe Creative.

Pressman, R. S. (2010). *INGENIERIA DE SOFTWARE: Un Enfoque Práctico.*

Mexico: McGRAW-HILL Interamericana.

Rodriguez Rodriguez, J. M., & Daureo Campillo, M. J. (2003). *SISTEMAS DE INFORMACION: Aspectos Tecnicos y Legales.* España: UNIVERSIDAD DE ALMERIA.

Rumbaugh, J., & Blaha, M. (1991). *OBJECT-ORIENTED MODELING AND*

*DESIGN.* United States: Prentice-Hall.

Sánchez Maza, M. Á. (2012). *JAVASCRIPT.* Málaga, España: Innovación Y Cualificación.

Schwaber, K., & Sutherland, J. (2013). *LA GUIA DEFINITIVA DE SCRUM: Las Reglas de Juego.* United States: Creative Commons.

Sommerville, I. (2005). *INGENIERIA DEL SOFTWARE.* Madrid, España: Pearson Educacion S.A.

Talledo, J. S. (2015). *IMPLANTACIÓN DE APLICACIONES WEB EN ENTORNO INTERNET, INTRANET Y EXTRANET.* España: Paraninfo.

Turban, E. (1995). *DECISION SUPPORT AND EXPERT SYSTEMS: Management*

*Support Systems.* Englewood Cliffs, United States: Prentice Hall.

Van Slype, G., Van Dyck, M., & Guillot, M. (1973). *SYSTEMES DOCUMENTAIRES ET ORDINATEUR.* Francia: Éditions d'Organisation.

Vértice, P. (2010). *E-COMMERCE: Aplicación y Desarrollo.* España: Editorial Vértice.