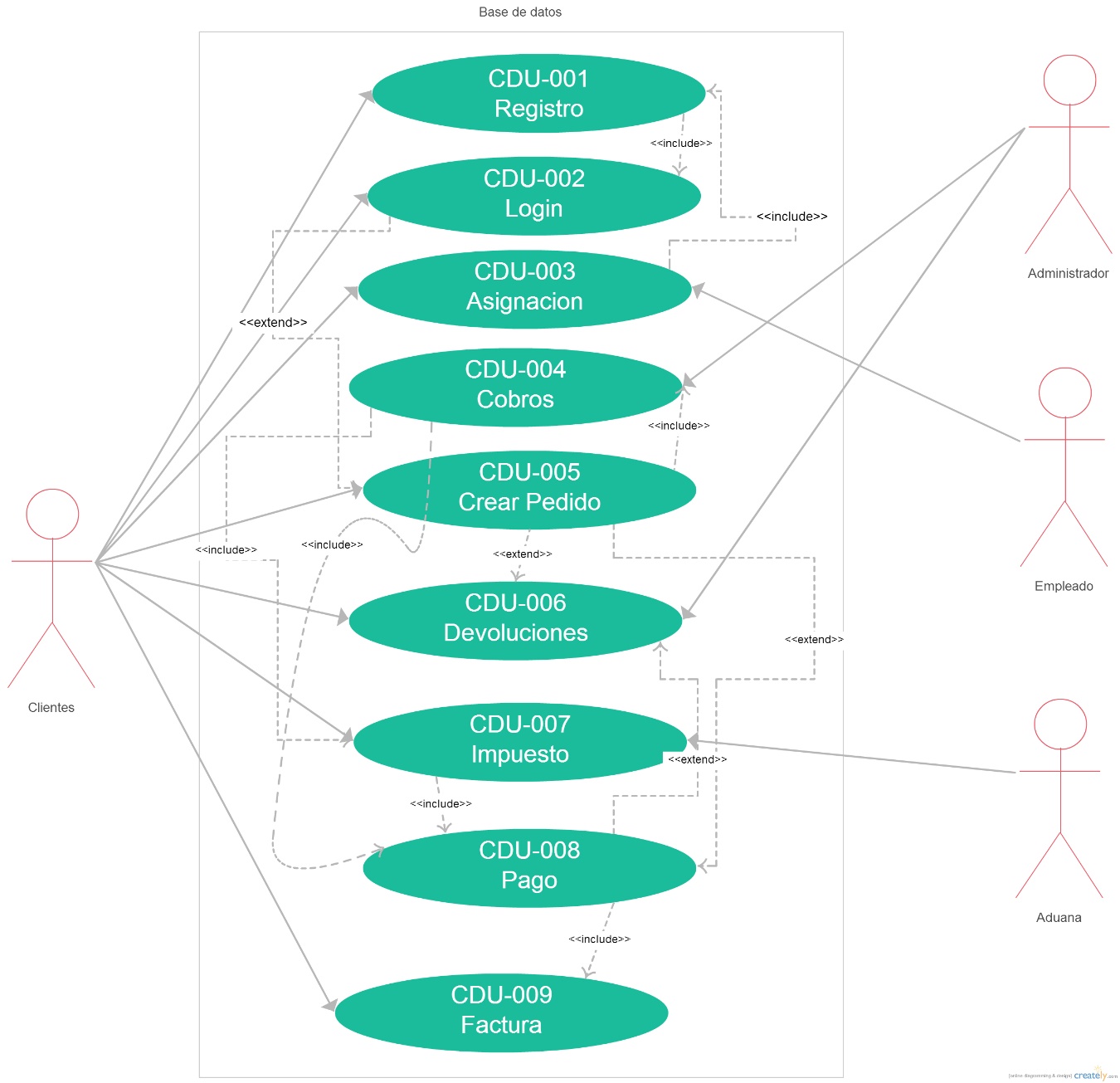
Universidad de San Carlos de Guatemala Facultad De Ingeniería Introducción a la Programación y Computación 2

Fase 3

Cesar Javier Solares Orozco

201313819

* **Caso de uso: Alto nivel, diagrama y Esenciales expandido.**
* **Diagrama:**



* **Caso de alto nivel:**

|  |  |
| --- | --- |
| Caso de Uso | CDU-001 Registro |
| Actores | Clientes |
| Tipo | Primario (Esencial) |
| Descripción | El cliente registra todos sus datos personales, los cuales se almacenaran en la base de datos. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-002 Login |
| Actores | Clientes |
| Tipo | Primario (Esencial) |
| Descripción | El Cliente una vez registrado y autorizado por el empleado podrá iniciar sesión en sistema para realizar las distintas funciones que ofrece el sistema. Administrador y empleados también podrán utilizar el login. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-003 Asignación |
| Actores | Clientes y Empleado |
| Tipo | Primario (Esencial) |
| Descripción | Una vez registrado el usuario se le asignara una casilla internacional con sus datos respectivos para almacenar sus paquetes. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-004 Cobros |
| Actores | Administrador |
| Tipo | Primario (Esencial) |
| Descripción | El Administrador administra el porcentaje de comisión para la empresa, así como los diferentes impuestos por producto. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-005 Crear Pedido |
| Actores | Clientes |
| Tipo | Primario (No Esencial) |
| Descripción | Los clientes pueden realizar pedidos de los productos disponibles, se le presentaran también los impuestos a pagar. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-006 Devoluciones |
| Actores | Clientes y Administrador |
| Tipo | Secundario |
| Descripción | Los clientes pueden devolver el producto siempre y cuando se haya hecho el pago de importación. El administrador es quien verifica el porqué de la devolución. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-008 Pago |
| Actores | Clientes |
| Tipo | Primario (Esencial) |
| Descripción | El pago será el total a pagar por el pedido ya creado anteriormente por el cliente, es como un tipo de cálculo total. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-007 Impuestos |
| Actores | Clientes y Aduana |
| Tipo | Primario (Esencial) |
| Descripción | El cliente al realizar un pedido se le realizara un cobro con el cual la aduana tiene que proporcionar la información de cuánto será el impuesto por artículo pedido. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-009 Factura |
| Actores | Clientes |
| Tipo | Primario (Esencial) |
| Descripción | Sera la confirmación del pago que ya almacena un pedido antes, con la factura tendrá derecho a reclamar algo que no le guste. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-001 Registro |
| Actores | Clientes |
| Propósito | El cliente ingresa en el modulo de registro e ingresa sus datos. |
| Resumen | El usuario registrará sus datos personales y serán almacenados en la base de datos. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R2, #R3 |
| Curso Normal | 1. El cliente ingresa en el modulo de registro 2. Ingresa sus datos personales 3. Envía su petición de registro 4. Espera su autorización 5. Registro completo |
| Cursos alternos | Línea 2: El cliente no tiene algún dato obligatorio. Línea 3: El cliente no es aceptado por algún empleado. |

* **Caso de Uso esenciales expandidos.**

|  |  |
| --- | --- |
| Caso de Uso | CDU-002 Login |
| Actores | Clientes |
| Propósito | Que el cliente registrado inicie sesion |
| Resumen | El usuario entrara al sistema con un usuario único y contraseña. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R1, #R5 |
| Curso Normal | 1. El cliente ingresa en el modulo de Log In 2. Ingresa su usuario único y su Password 3. Se verifica en el sistema los datos 4. Inicia sesión. |
| Cursos alternos | Línea 2: El cliente no está registrado o autorizado. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-003 Asignación |
| Actores | Sistema |
| Propósito | El empleado autoriza el registro del cliente y se le asigna una casilla internacional a este. |
| Resumen | El usuario con registro autorizado tiene una casilla internacional donde podrá recibir los paquetes |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R1 #R2 |
| Curso Normal | 1. El sistema recibe la autorización pedida del cliente 2. El sistema asigna una casilla internacional al cliente. |
| Cursos alternos | El cliente no esta autorizado.  El cliente no esta registrado. |
|  |  |

|  |  |
| --- | --- |
| Caso de Uso | CDU-004 Cobros |
| Actores | Administrador |
| Propósito | El administrador esta encargado de administrar la comisión por producto y los impuestos que se aplican al mismo. |
| Resumen | El administrador asiga la comisión por producto y los impuestos que cobran por producto. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R7 |
| Curso Normal | 1. El Administrar pedirá los impuestos a la aduana. 2. El Administrador estará encargado de mantener al día los artículos. 3. El Administrador cargará todo tipo de cobros al pedido hecho por el usuario. |
| Cursos alternos | Línea 1: La aduana no le proporcione los impuestos. |
|  |  |

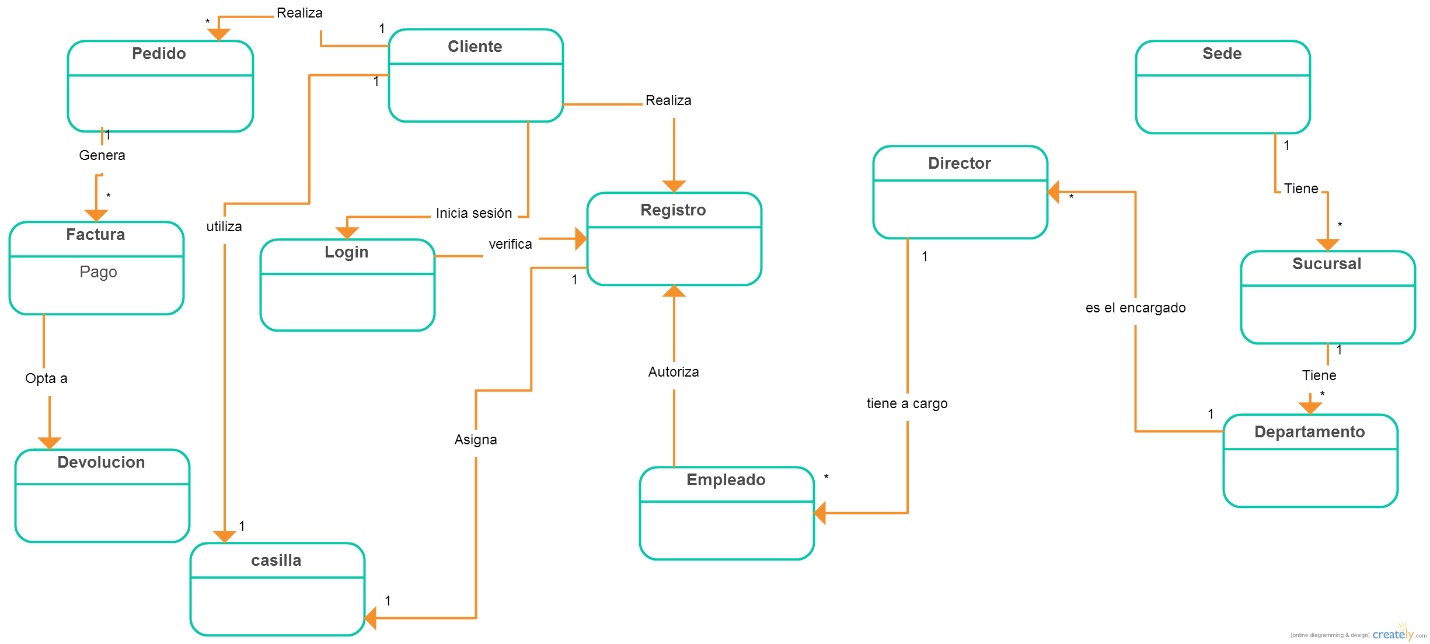
|  |  |
| --- | --- |
| Caso de Uso | CDU-005 Crear Pedido |
| Actores | Cliente |
| Propósito | El cliente adjunta al carrito todos los productos que desea comprar |
| Resumen | El cliente realiza du pedido con los productos seleccionados del carrito. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R4, #R6 |
| Curso Normal | 1. El cliente ve todos los productos que desea comprar 2. Los adjunta al carrito(pedido) 3. Ve le total del pedido 4. Realiza el pedido. |
| Cursos alternos | Línea 2: Algún artículo le falte los cobros. |

|  |  |
| --- | --- |
| Caso de Uso | CDU-006 Devoluciones |
| Actores | Cliente y Administrador |
| Propósito | Que el cliente pueda hacer una devolución de su producto |
| Resumen | Si el cliente lo desea puede devolver su producto sin ningun costo. |
| Tipo | Segundario (Esencial) |
| Referencia Cruzada | #R5 |
| Curso Normal | 1. El cliente solicita una devolución 2. Realiza la petición al administrador 3. El administrador autoriza la devolución 4. Se realiza la devolución. |
| Cursos alternos | Línea 1: El administrador no autorice la devolución.  Linea 1: el cliente no pago la importacion. |

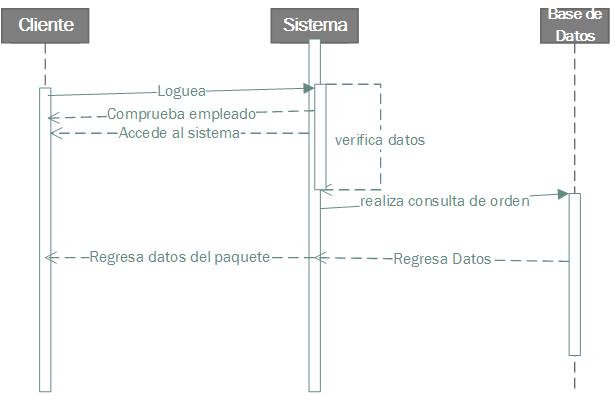
|  |  |
| --- | --- |
| Caso de Uso | CDU-007 Impuestos |
| Actores | Cliente y Administrador |
| Propósito | El cliente deberá pagar los impuestos por producto que desee comprar. |
| Resumen | El cliente tendrá claro que impuestos pagará por los artículos disponibles que estarán al día con el administrador ya autorizados en la página. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R4 |
| Curso Normal | 1. El cliente realiza su pedido 2. Se muestra el total de su pedido y los impuestos totales a pagar 3. El cliente paga los impuestos |
| Cursos alternos | Línea 1: El impuesto no esté disponible |

|  |  |
| --- | --- |
| Caso de Uso | CDU-008 Pago |
| Actores | Sistema |
| Propósito | Almacenar todos los pagos de productos e impuestos |
| Resumen | Almacenar el pago efectuado con todos los datos correspondientes del pedido |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R7, #R5 |
| Curso Normal | 1. El cliente realiza el pedido 2. El sistema muestra el total del pedido 3. El cliente efectua el pago 4. El sistema almacena ese pago |
| Cursos alternos | Línea 2: El artículo no tenga el impuesto disponible. |

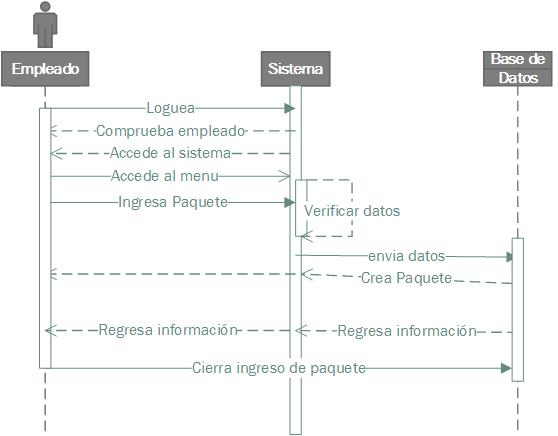
|  |  |
| --- | --- |
| Caso de Uso | CDU-009 Factura |
| Actores | Cliente |
| Propósito | Entregar un comprobante de la compra escrita al cliente |
| Resumen | Almacenará artículo y el costo final, el costo final del pedido ya echo anteriormente por el usuario. |
| Tipo | Primario (Esencial) |
| Referencia Cruzada | #R8 |
| Curso Normal | 1. El cliente efectúa su pago. 2. El sistema genera una factura. |
| Cursos alternos | Línea 1: No exista pago |

**-Modelo Conceptual** 

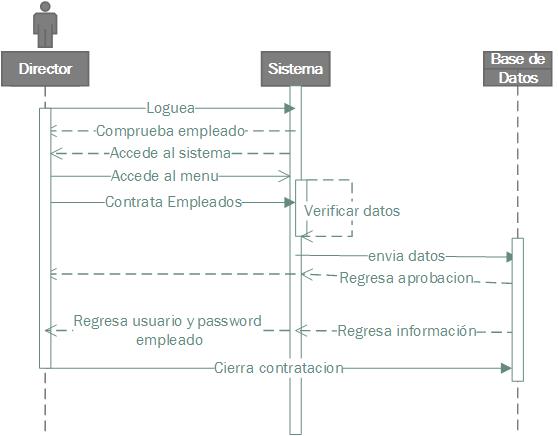
* **Diagrama de Secuencias**
  + **Cliente**

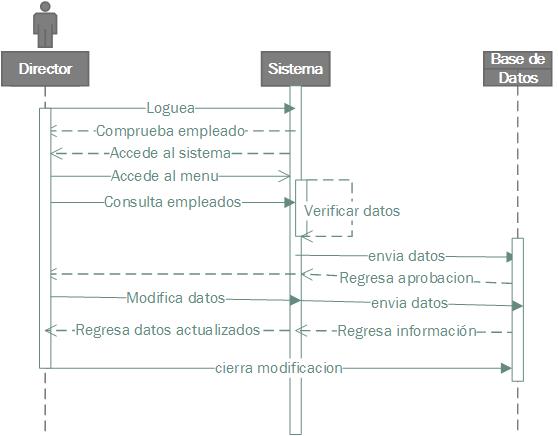
****

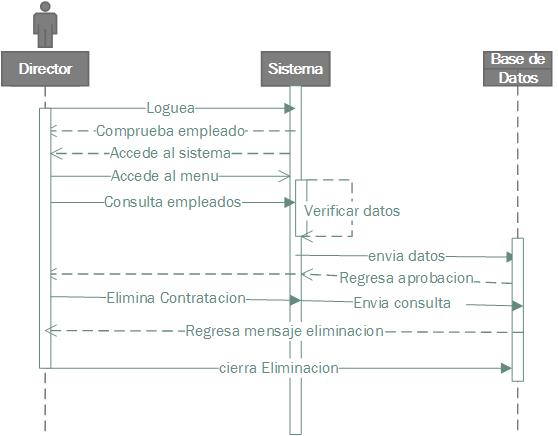
* + **Empleado**

****

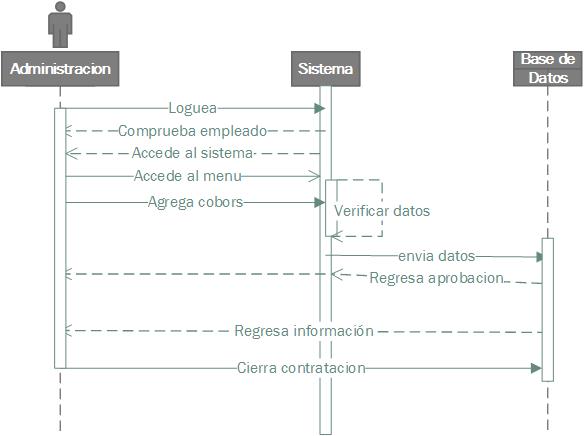
* + **Director**

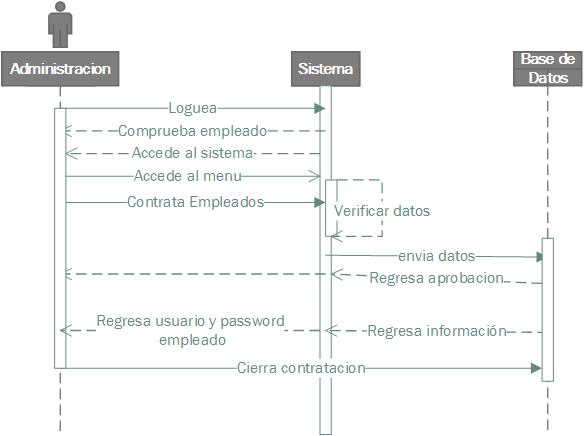


****

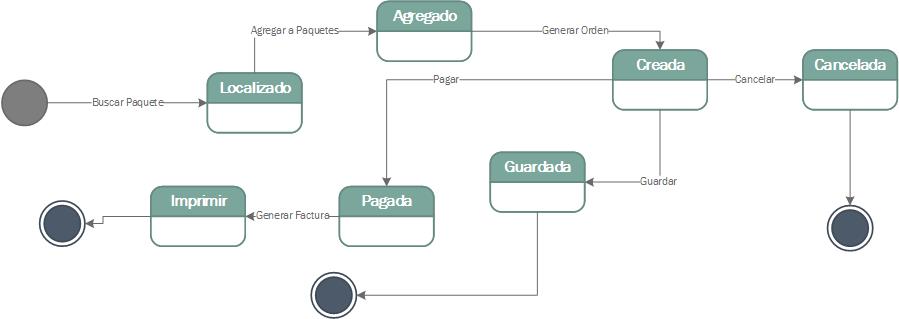
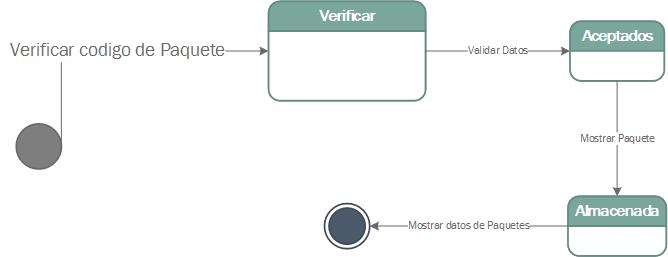
****

* + **Administrador**



****

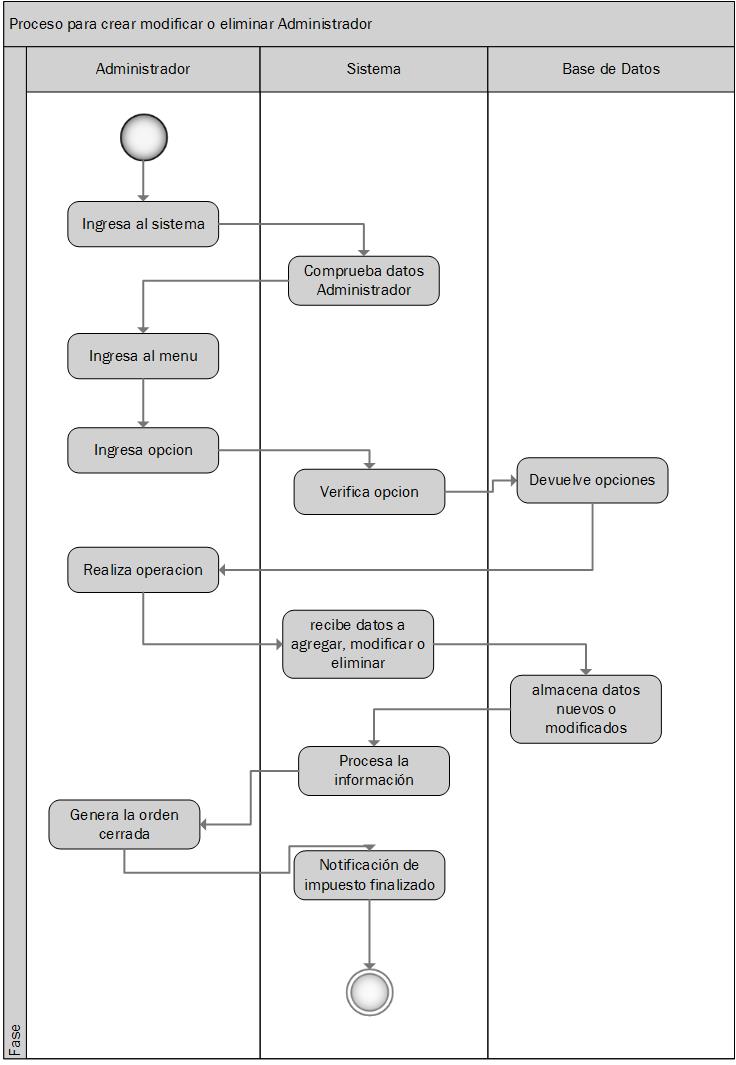
* **Diagrama de estados**

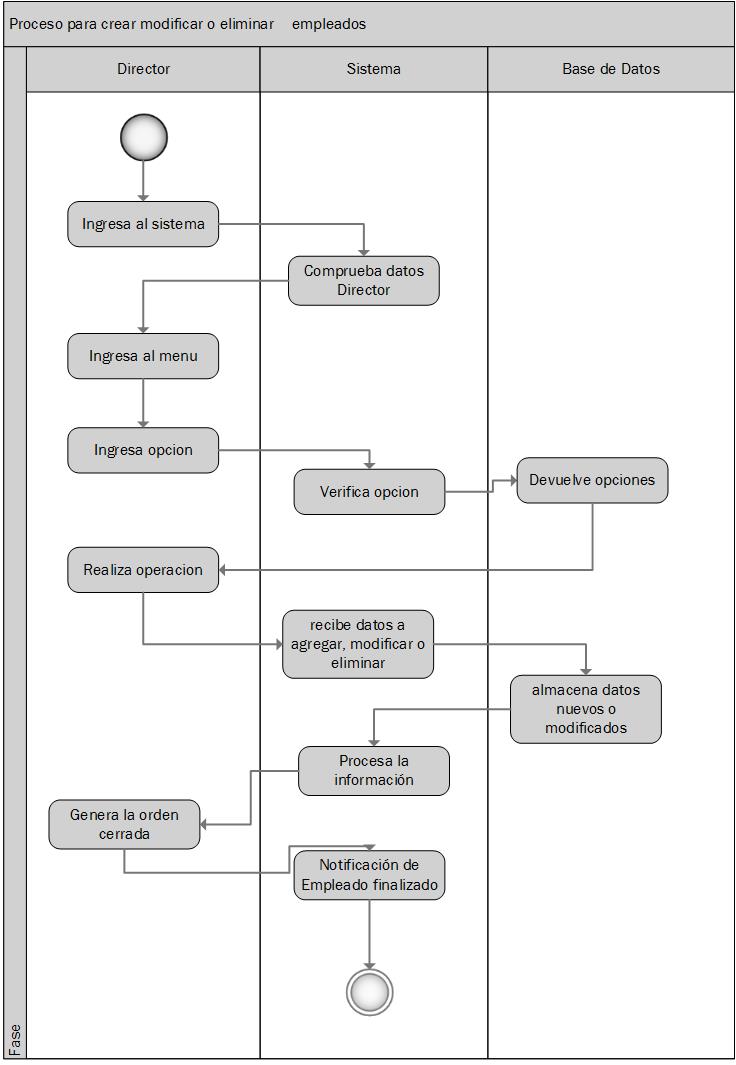
****

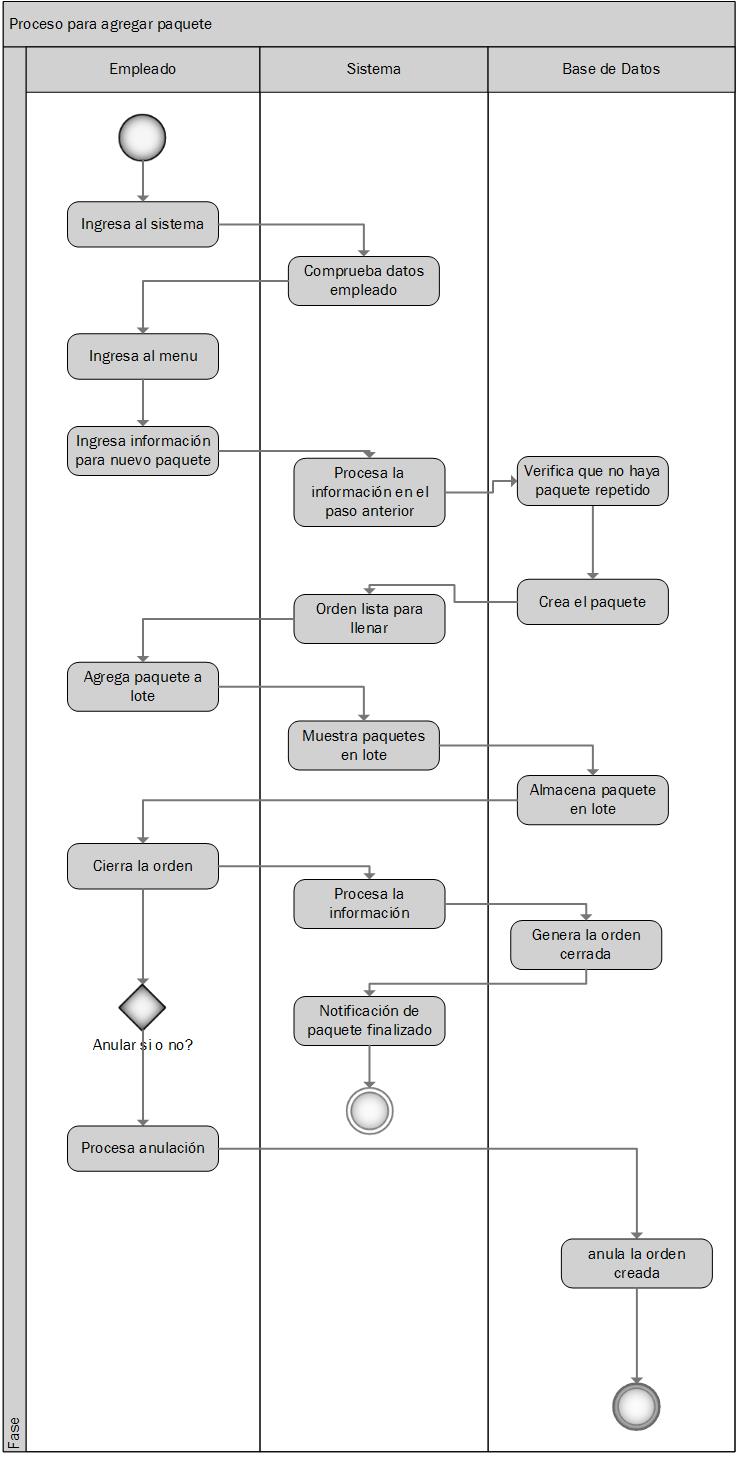
**-Glosario**

|  |  |  |  |
| --- | --- | --- | --- |
| Termino | Categoría | Descripción | Notación |
| Actores | Concepto/ Caso de uso | Entidad externa al sistema que de alguna forma participa en la historia del caso de uso. |  |
| Almacenamiento | Arquitectura de software | Es donde se almacena la información que se desea ingresar a la aplicación. |  |
| Arquitectura de Software | Concepto | Una arquitectura de software se selecciona y diseña con base en objetivos (requerimientos) y restricciones y es el diseño más alto nivel de la estructura de un sistema. |  |
| Base de Datos | Concepto | Es un conjunto de datos que persisten que son utilizados por una aplicación de software. |  |
| Capa de presentación | Concepto/ Arquitectura Software | Es la que ve el usuario, también conocida como interfaz gráfica, presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso. Esta capa se comunica únicamente con la capa de negocio. |  |
| Capa de negocio | Concepto/ Arquitectura software | Es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. |  |
| Capa de datos | Concepto/ Arquitectura de Software | Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos. |  |
| Casos de Uso | Concepto | Documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso. |  |
| Diagrama Entidad – Relación: | concepto | Es la manera de expresar gráficamente un modelo de datos. Está compuesto de entidades y las relaciones entre las mismas. |  |
| Diseño | Arquitectura de Software | Nivel conceptual y la parte de diagramas que permiten la creación de la aplicación. |  |
| Entidad | Concepto/diagrama entidad-relación | Representa cualquier objeto distinguible en un modelo de negocios que se debe representar en la base de datos. | NOMBRE |
| Gliffy | Concepto | Es un sitio web que nos da las herramientas necesarias para diseñar diversos tipos de diagramas profesionales. |  |
| Include | Concepto/Caso de uso | Relación directa entre 2 CDU  Los CDU incluidos son obligatorios y siempre son requeridos para ejecutar correctamente el CDU en el que se incluye.  <<include>> |  |
| Llave Foránea (FK) | Concepto/Diagrama entidad-relación | Es el conjunto de uno o más atributos que permiten relacionar la entidad a la que pertenece con la entidad que se relaciona la misma. | (#) |
| Llave Primaria (PK) | Concepto/ Diagrama entidad-relación | Es el identificador único para cada tupla de una entidad. Puede ser uno o más atributos que hacen única la tupla. Son obligatorios. | # |
| Llaves únicas | Concepto/Diagrama entidad-relación | Restricción que impide la repetición del valor dentro de la misma columna, pero no identifica a la tupla. Pueden  Haber valores nulos. |  |
| Modelo conceptual | Concepto | permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que se relacionan esos elementos entre sí |  |
| Relaciones de entidad | Concepto/ diagrama entidad-relación | Una relación es aquella que permite describir una relación entre los datos. Las relaciones entre entidades. |  |
| Relaciones De uno a uno | Concepto/ Diagrama entidad-relación | Esta relación ocurre cuando una instancia en la entidad A está relacionada únicamente a una instancia en la entidad B. |  |
| Relaciones De uno a muchos | Concepto/  Diagrama entidad-relación | Esta relación ocurre cuando una instancia en la entidad A está relacionada con varias instancias en la entidad B. |  |
| SQL | Concepto | Es un lenguaje declarativo, con el cual se puede  acceder a la base de datos, para manipular  Información. |  |
| SQL Server | Concepto | Es un sistema de manejo de bases de datos del modelo relacional, desarrollado por la empresa Microsoft. |  |

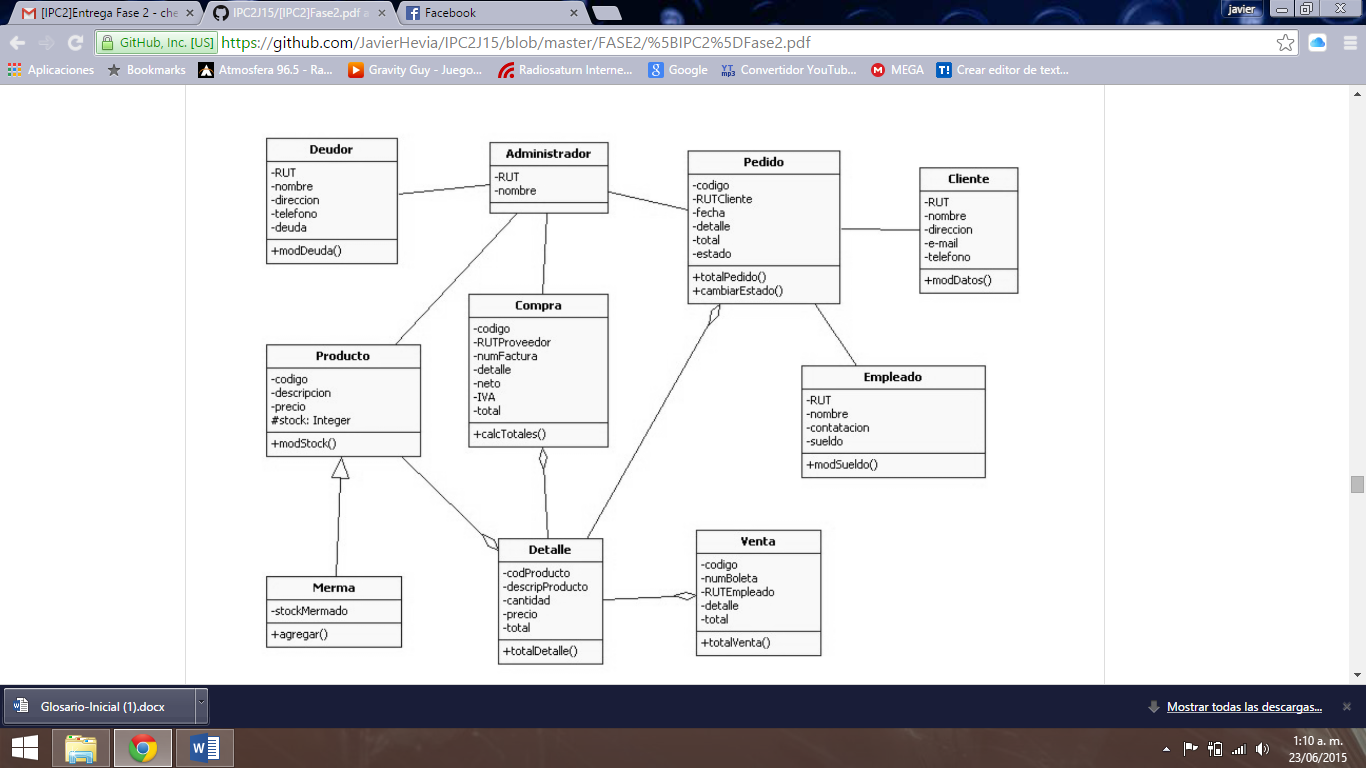
* Diagrama de actividades



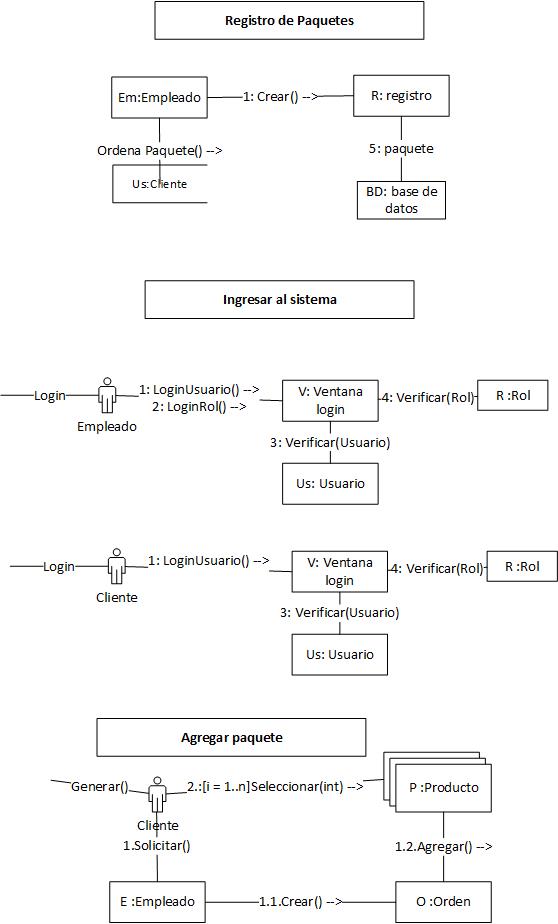




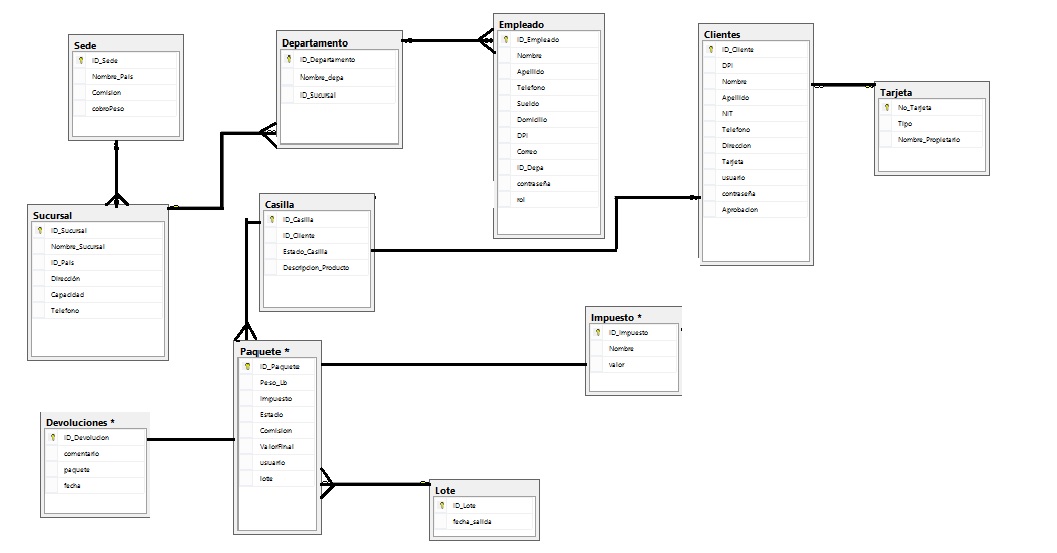
* Diagrama de Clases



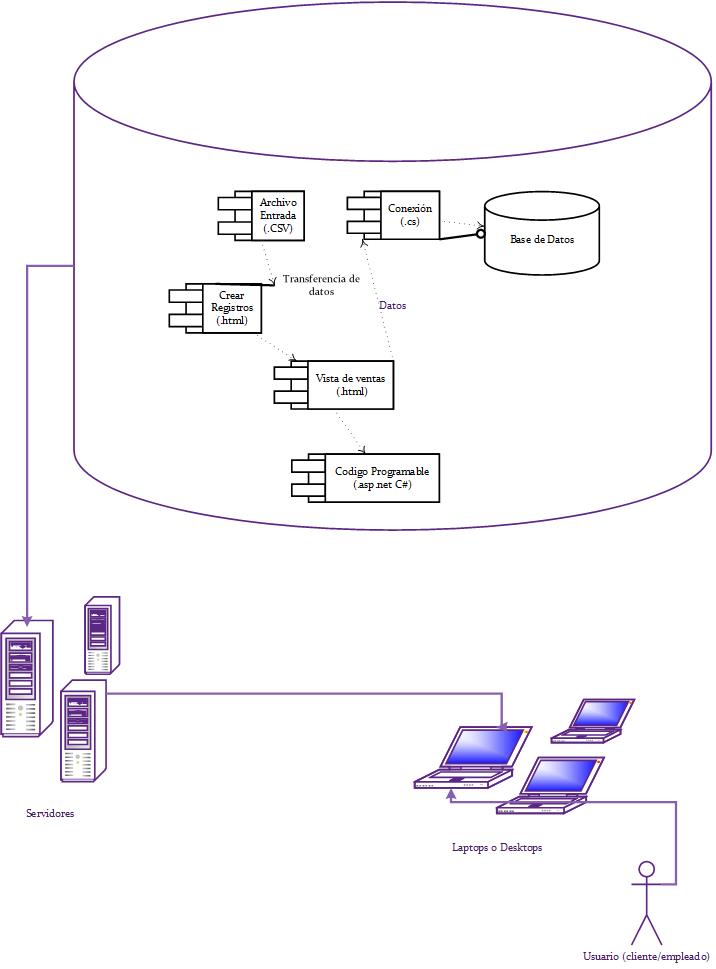
* Diagrama de Colaboracion



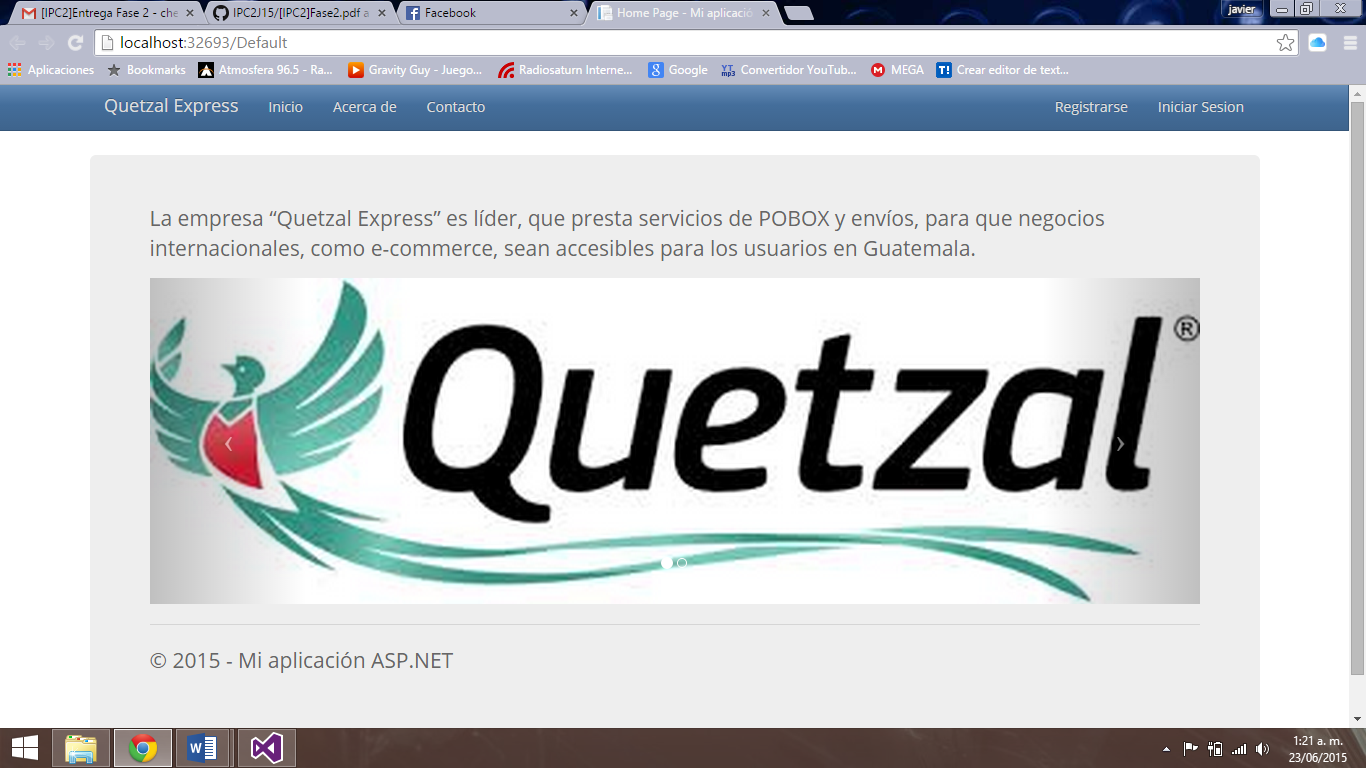
* Entidad Relacion



* Arquitectura Componentes



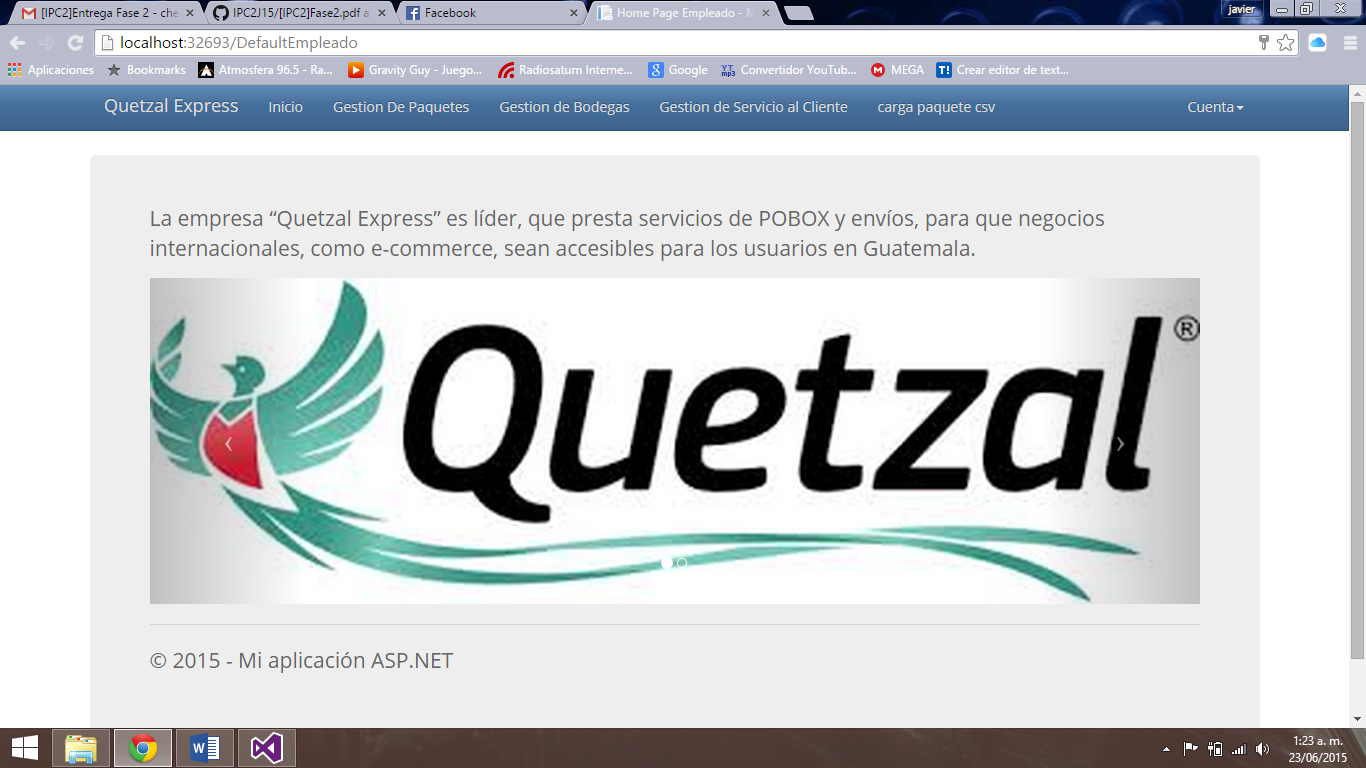
* Definicion de Diseño
  + Vista General



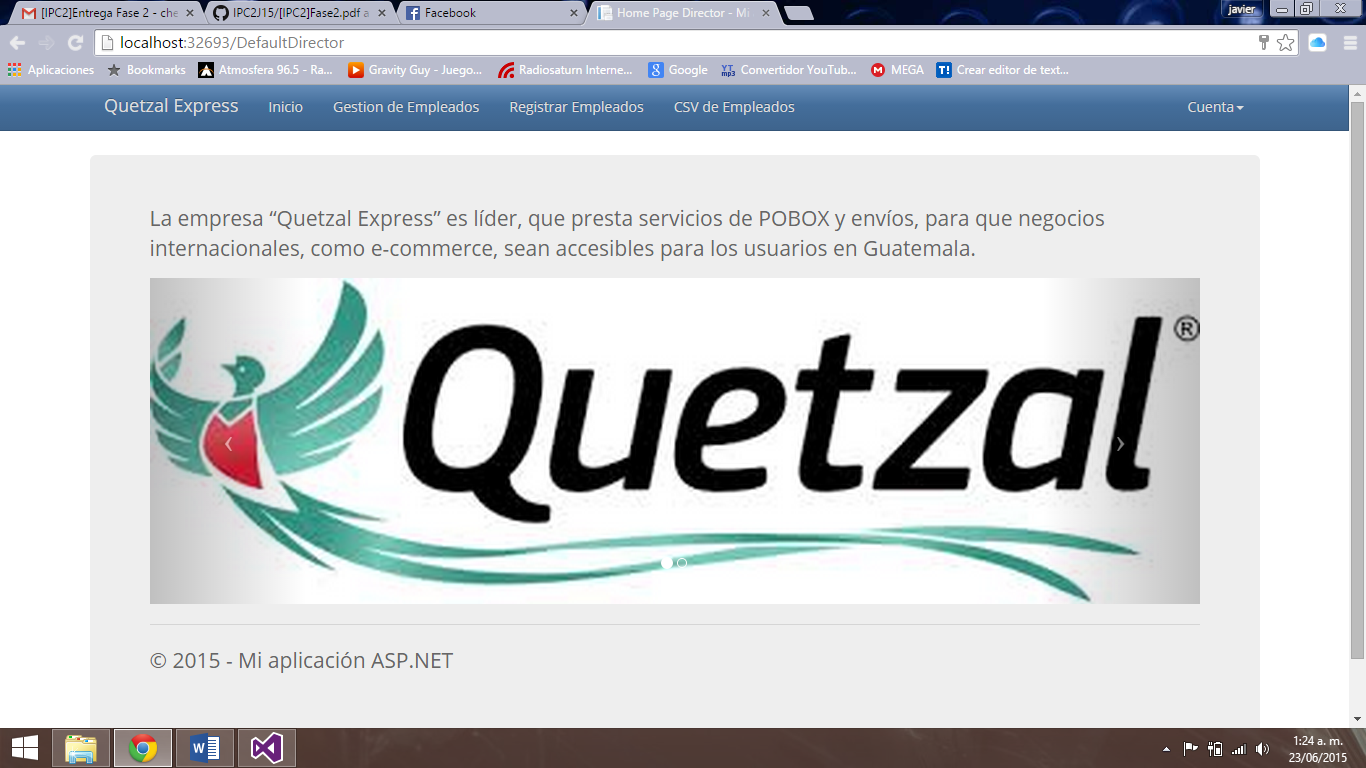
* + Vista Cliente



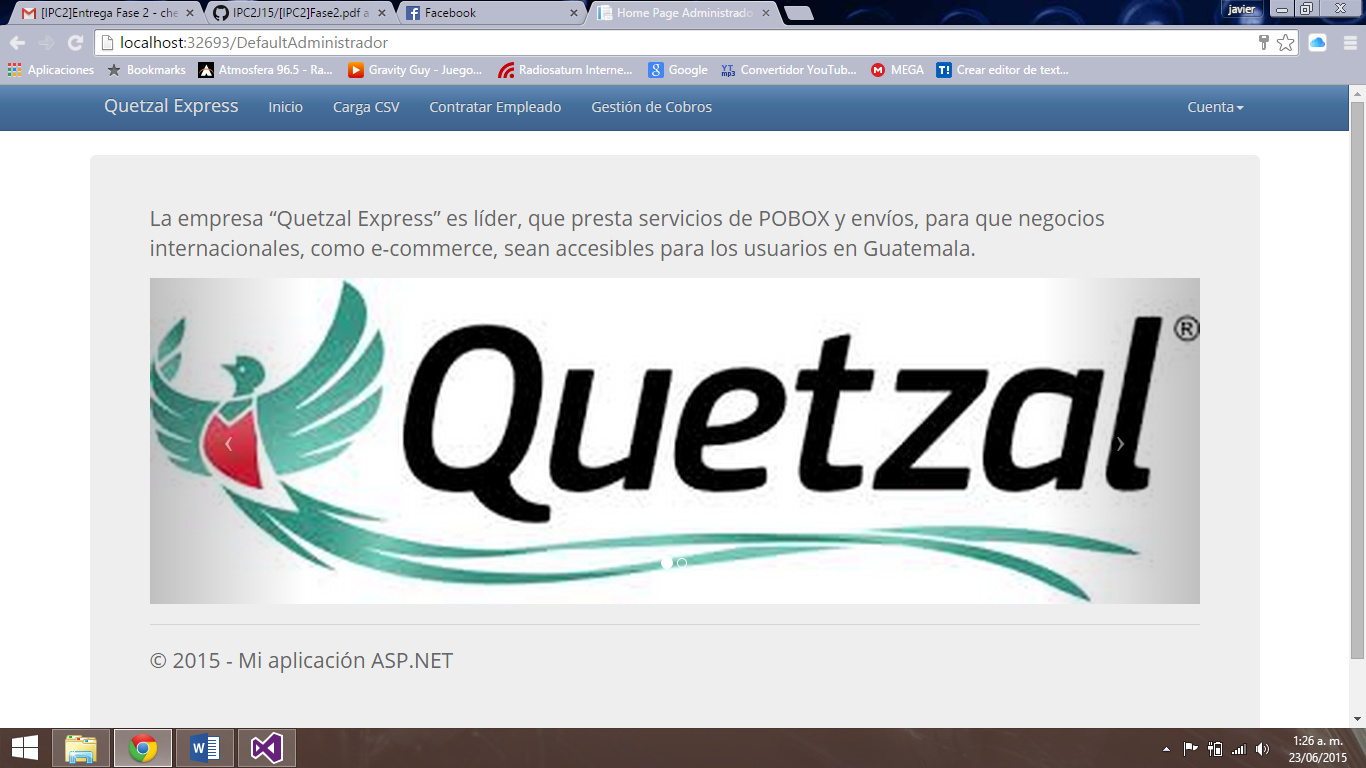
* + Vista Empleado



* + Vista Director



* + Vista Administrador



* Script base de datos

USE [master]

GO

/\*\*\*\*\*\* Object: Database [Fase2J15] Script Date: 23/06/2015 1:34:44 a. m. \*\*\*\*\*\*/

CREATE DATABASE [Fase2J15]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'Fase2J15', FILENAME = N'C:\Program Files (x86)\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Fase2J15.mdf' , SIZE = 5120KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'Fase2J15\_log', FILENAME = N'C:\Program Files (x86)\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Fase2J15\_log.ldf' , SIZE = 1024KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

ALTER DATABASE [Fase2J15] SET COMPATIBILITY\_LEVEL = 110

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [Fase2J15].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [Fase2J15] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [Fase2J15] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [Fase2J15] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [Fase2J15] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [Fase2J15] SET ARITHABORT OFF

GO

ALTER DATABASE [Fase2J15] SET AUTO\_CLOSE OFF

GO

ALTER DATABASE [Fase2J15] SET AUTO\_CREATE\_STATISTICS ON

GO

ALTER DATABASE [Fase2J15] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [Fase2J15] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [Fase2J15] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [Fase2J15] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [Fase2J15] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [Fase2J15] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [Fase2J15] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [Fase2J15] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [Fase2J15] SET DISABLE\_BROKER

GO

ALTER DATABASE [Fase2J15] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [Fase2J15] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [Fase2J15] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [Fase2J15] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [Fase2J15] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [Fase2J15] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [Fase2J15] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [Fase2J15] SET RECOVERY FULL

GO

ALTER DATABASE [Fase2J15] SET MULTI\_USER

GO

ALTER DATABASE [Fase2J15] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [Fase2J15] SET DB\_CHAINING OFF

GO

ALTER DATABASE [Fase2J15] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [Fase2J15] SET TARGET\_RECOVERY\_TIME = 0 SECONDS

GO

EXEC sys.sp\_db\_vardecimal\_storage\_format N'Fase2J15', N'ON'

GO

USE [Fase2J15]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Administrador] Script Date: 23/06/2015 1:34:44 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Administrador](

[ID\_Administrador] [int] IDENTITY(1,1) NOT NULL,

[Nombre] [varchar](max) NULL,

[Apellido] [varchar](max) NULL,

[Telefono] [int] NULL,

[Sueldo] [float] NULL,

[Domicilio] [varchar](max) NULL,

[DPI] [int] NULL,

[Correo] [varchar](max) NULL,

[contraseña] [varchar](50) NOT NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Clientes] Script Date: 23/06/2015 1:34:44 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Clientes](

[ID\_Cliente] [int] IDENTITY(1,1) NOT NULL,

[DPI] [int] NOT NULL,

[Nombre] [varchar](max) NOT NULL,

[Apellido] [varchar](max) NOT NULL,

[NIT] [int] NOT NULL,

[Telefono] [int] NOT NULL,

[Direccion] [varchar](max) NOT NULL,

[Tarjeta] [int] NOT NULL,

[usuario] [varchar](max) NULL,

[contraseña] [varchar](max) NULL,

[Aprobacion] [int] NULL,

[Casilla] [int] NULL,

CONSTRAINT [PK\_Clientes] PRIMARY KEY CLUSTERED

(

[ID\_Cliente] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Departamento] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Departamento](

[ID\_Departamento] [int] IDENTITY(1,1) NOT NULL,

[Nombre\_depa] [varchar](max) NOT NULL,

[ID\_Sucursal] [int] NOT NULL,

CONSTRAINT [PK\_Departamento] PRIMARY KEY CLUSTERED

(

[ID\_Departamento] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Devoluciones] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Devoluciones](

[ID\_Devolucion] [int] IDENTITY(1,1) NOT NULL,

[ID\_Pago] [int] NOT NULL,

[comentario] [varchar](max) NOT NULL,

[fecha] [varchar](max) NOT NULL,

CONSTRAINT [PK\_Devoluciones] PRIMARY KEY CLUSTERED

(

[ID\_Devolucion] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Empleado] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Empleado](

[ID\_Empleado] [int] IDENTITY(1,1) NOT NULL,

[Nombre] [varchar](max) NULL,

[Apellido] [varchar](max) NULL,

[Telefono] [int] NULL,

[Sueldo] [float] NULL,

[Domicilio] [varchar](max) NULL,

[DPI] [int] NULL,

[Correo] [varchar](max) NULL,

[ID\_Depa] [int] NULL,

[contraseña] [varchar](50) NOT NULL,

[rol] [varchar](50) NULL,

CONSTRAINT [PK\_Empleado] PRIMARY KEY CLUSTERED

(

[ID\_Empleado] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Historial] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Historial](

[ID\_Historial] [int] NULL

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Impuesto] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Impuesto](

[ID\_Impuesto] [int] IDENTITY(1,1) NOT NULL,

[Nombre] [varchar](max) NULL,

[valor] [int] NULL,

CONSTRAINT [PK\_Impuesto] PRIMARY KEY CLUSTERED

(

[ID\_Impuesto] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Lote] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Lote](

[ID\_Lote] [int] IDENTITY(1,1) NOT NULL,

[fecha\_salida] [varchar](50) NULL,

[fecha\_envio] [varchar](50) NULL,

CONSTRAINT [PK\_Lot] PRIMARY KEY CLUSTERED

(

[ID\_Lote] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Paquete] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Paquete](

[ID\_Paquete] [int] IDENTITY(1,1) NOT NULL,

[Peso\_Lb] [float] NULL,

[Impuesto] [int] NULL,

[Estado] [varchar](50) NULL,

[Comision] [int] NULL,

[ValorFinal] [float] NULL,

[usuario] [int] NULL,

[lote] [int] NULL,

CONSTRAINT [PK\_Paquete] PRIMARY KEY CLUSTERED

(

[ID\_Paquete] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Producto] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Producto](

[ID\_Producto] [int] IDENTITY(1,1) NOT NULL,

[Peso] [float] NOT NULL,

[Valor] [float] NOT NULL,

[Costo] [float] NOT NULL,

[Impuesto] [int] NOT NULL,

[Comision] [float] NOT NULL,

[Total] [float] NOT NULL,

[Stock] [int] NOT NULL,

CONSTRAINT [PK\_Producto] PRIMARY KEY CLUSTERED

(

[ID\_Producto] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Sede] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Sede](

[ID\_Sede] [int] IDENTITY(1,1) NOT NULL,

[Nombre\_Pais] [varchar](max) NULL,

[Comision] [int] NULL,

[cobroPeso] [int] NULL,

CONSTRAINT [PK\_Sede] PRIMARY KEY CLUSTERED

(

[ID\_Sede] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Sucursal] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Sucursal](

[ID\_Sucursal] [int] IDENTITY(1,1) NOT NULL,

[Nombre\_Sucursal] [varchar](max) NULL,

[ID\_Pais] [int] NOT NULL,

[Dirección] [varchar](max) NOT NULL,

[Capacidad] [int] NOT NULL,

[Telefono] [int] NOT NULL,

CONSTRAINT [PK\_Sucursal] PRIMARY KEY CLUSTERED

(

[ID\_Sucursal] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[Tarjeta] Script Date: 23/06/2015 1:34:45 a. m. \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Tarjeta](

[No\_Tarjeta] [int] NOT NULL,

[Tipo] [varchar](max) NOT NULL,

[Nombre\_Propietario] [varchar](max) NOT NULL,

CONSTRAINT [PK\_Tarjeta] PRIMARY KEY CLUSTERED

(

[No\_Tarjeta] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

ALTER TABLE [dbo].[Administrador] ADD CONSTRAINT [DF\_Administrador\_contraseña] DEFAULT ('2015quetzal') FOR [contraseña]

GO

ALTER TABLE [dbo].[Empleado] ADD CONSTRAINT [DF\_Empleado\_contraseña] DEFAULT ('2015quetzal') FOR [contraseña]

GO

ALTER TABLE [dbo].[Clientes] WITH CHECK ADD CONSTRAINT [FK\_Clientes\_Tarjeta] FOREIGN KEY([Tarjeta])

REFERENCES [dbo].[Tarjeta] ([No\_Tarjeta])

GO

ALTER TABLE [dbo].[Clientes] CHECK CONSTRAINT [FK\_Clientes\_Tarjeta]

GO

ALTER TABLE [dbo].[Departamento] WITH CHECK ADD CONSTRAINT [FK\_Departamento\_Sucursal] FOREIGN KEY([ID\_Sucursal])

REFERENCES [dbo].[Sucursal] ([ID\_Sucursal])

GO

ALTER TABLE [dbo].[Departamento] CHECK CONSTRAINT [FK\_Departamento\_Sucursal]

GO

ALTER TABLE [dbo].[Empleado] WITH NOCHECK ADD CONSTRAINT [FK\_Empleado\_Departamento] FOREIGN KEY([ID\_Depa])

REFERENCES [dbo].[Departamento] ([ID\_Departamento])

GO

ALTER TABLE [dbo].[Empleado] CHECK CONSTRAINT [FK\_Empleado\_Departamento]

GO

ALTER TABLE [dbo].[Paquete] WITH CHECK ADD CONSTRAINT [FK\_Paquete\_Clientes] FOREIGN KEY([usuario])

REFERENCES [dbo].[Clientes] ([ID\_Cliente])

GO

ALTER TABLE [dbo].[Paquete] CHECK CONSTRAINT [FK\_Paquete\_Clientes]

GO

ALTER TABLE [dbo].[Paquete] WITH CHECK ADD CONSTRAINT [FK\_Paquete\_Lot] FOREIGN KEY([lote])

REFERENCES [dbo].[Lote] ([ID\_Lote])

GO

ALTER TABLE [dbo].[Paquete] CHECK CONSTRAINT [FK\_Paquete\_Lot]

GO

ALTER TABLE [dbo].[Producto] WITH CHECK ADD CONSTRAINT [FK\_Producto\_Impuesto] FOREIGN KEY([Impuesto])

REFERENCES [dbo].[Impuesto] ([ID\_Impuesto])

GO

ALTER TABLE [dbo].[Producto] CHECK CONSTRAINT [FK\_Producto\_Impuesto]

GO

ALTER TABLE [dbo].[Sucursal] WITH CHECK ADD CONSTRAINT [FK\_Sucursal\_Sede] FOREIGN KEY([ID\_Pais])

REFERENCES [dbo].[Sede] ([ID\_Sede])

GO

ALTER TABLE [dbo].[Sucursal] CHECK CONSTRAINT [FK\_Sucursal\_Sede]

GO

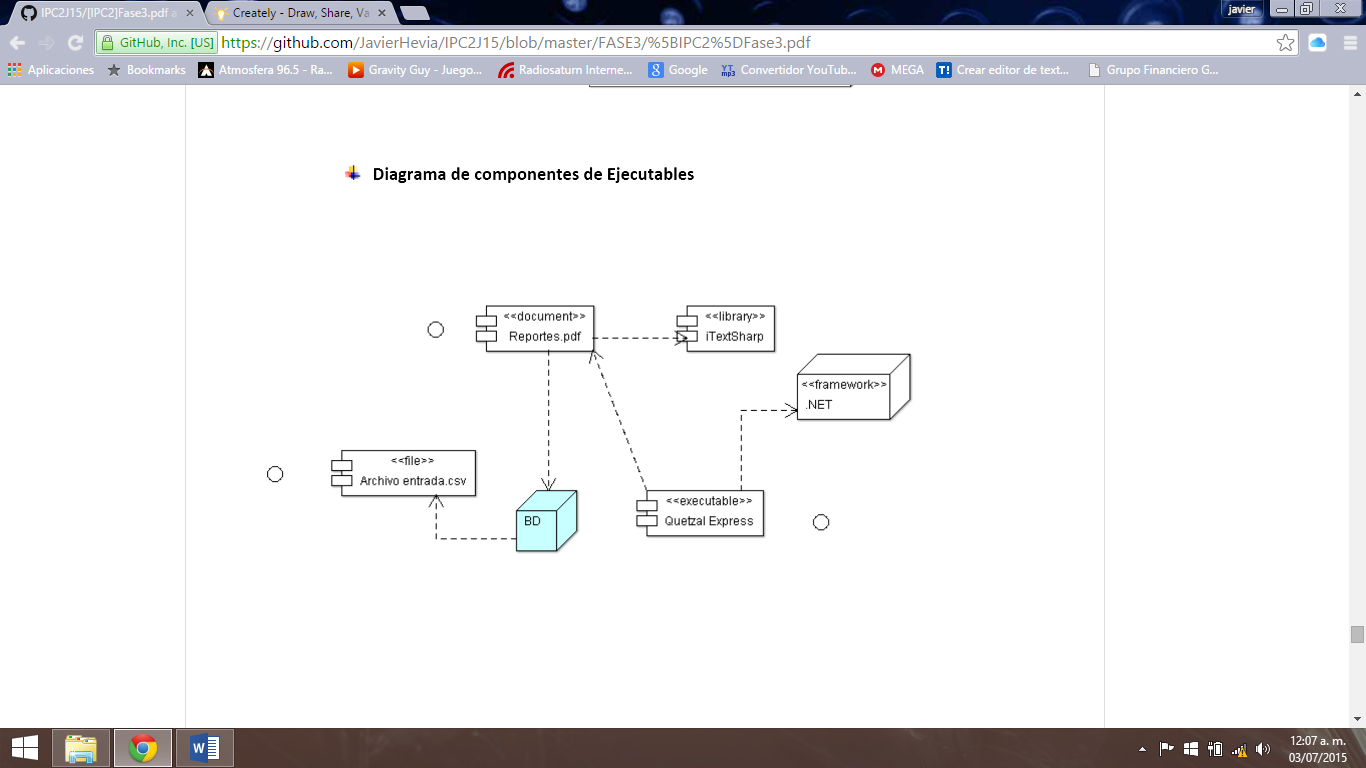
USE [master]

GO

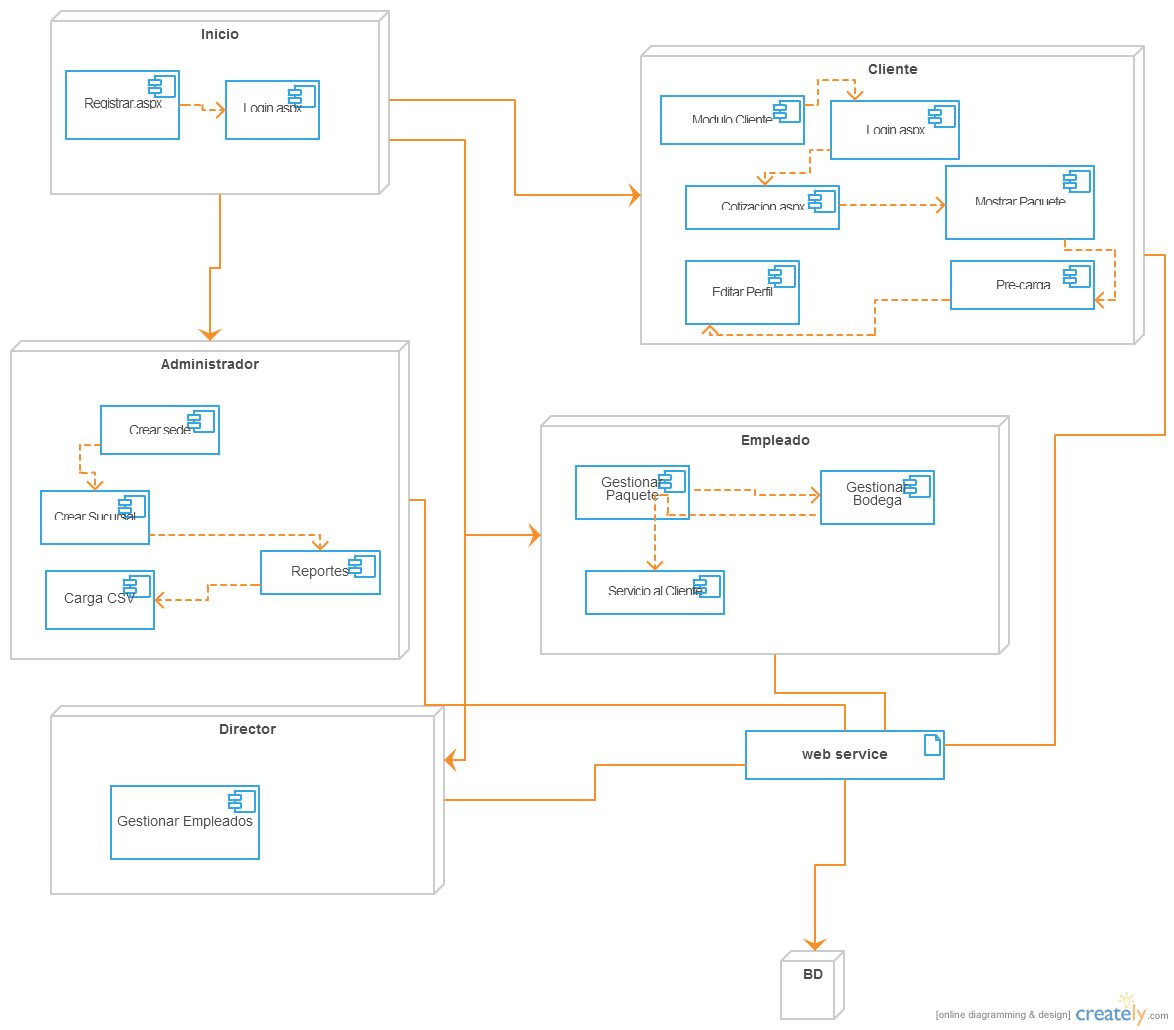
ALTER DATABASE [Fase2J15] SET READ\_WRITE

GO

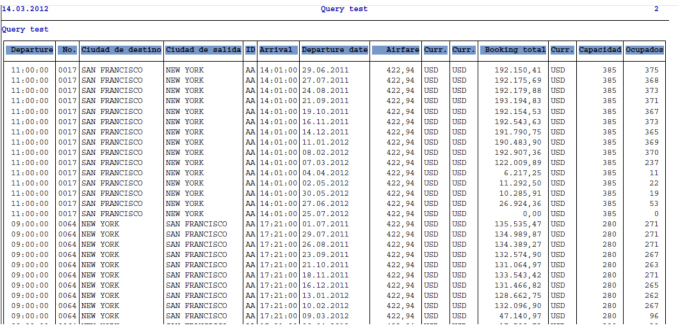
* **Diagrama de componentes Ejecutables**



**Diagrama de Componentes**

****

**Ejemplo de Reportes**

**En total serán 5 reportes**

**Arquitectura en tres capas**

El Patrón de arquitectura por capas es una de las técnicas más comúnes 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. En este esquema la capa más alta utiliza varios servicios definidos por la inferior, pero la ultima es inconsciente de la superior. Además, normalmente cada capa oculta las capas inferiores de las siguientes superiores a esta.

1. 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.
2. 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.
3. 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.

