

UNIVERSIDAD PEDAGOGICA DE EL SALVADOR

FACULTAD DE CIENCIAS ECONÓMICAS



**PROYECTO DE GRADUACION**

**TECNICO EN SISTEMAS DE COMPUTACION**

**TEMA:**

**“CREACION DE BASE DE DATOS PARA IMPLEMENTAR UN SISTEMA  
DE CONTROL DE PASANTIAS DE LA FACULTAD DE CIENCIAS  
ECONOMICAS DE LA UNIVERSIDAD PEDAGOGICA DE EL SALVADOR  
2011”**

**PRESENTAN:**

**KAREN JULISSA ALAS CRUZ**

**JOSE ISAAC GARCIA JOVEL**

## INDICE

<b>1.1 DATOS DE IDENTIFICACION</b>	1
<b>1.2 OBJETIVOS</b>	1
<b>1.3 DEFINICION DEL PROBLEMA</b>	1
<b>1.3 INTRODUCCION</b>	1
<b>1.4 JUSTIFICACION</b>	2
<b>2.0 FUNDAMENTACION TEORICA</b>	3
<b>2.1. Sistemas</b>	3
<b>2.1.1 Teoría de sistemas</b>	3
<b>2.1.2. Concepto de sistemas</b>	5
<b>2.1.3. Características de los sistemas</b>	6
<b>2.1.4. Sistema de información</b>	7
<b>2.1.4.1. Actividades que realiza un Sistema de Información</b>	8
<b>2.1.4.2. Tipos y Usos de los Sistemas de Información</b>	9
<b>2.1.4.3. Sistemas Transaccionales.</b>	9
<b>2.1.4.4. Sistemas de Apoyo de las Decisiones.</b>	10
<b>2.1.4.5. Sistemas Estratégicos.</b>	10
<b>2.1.4.6. Evolución de los Sistemas de Información</b>	11
<b>2.1.5. Clasificación de los sistemas</b>	15
<b>2.2. Proceso Unificado</b>	17
<b>2.2.1. Características del Proceso Unificado</b>	18
<b>2.2.2. Las 4 “P” del Proceso Unificado</b>	20
<b>2.2.3. Fases del Proceso Unificado</b>	21
<b>2.2.4. Disciplinas del Proceso Unificado</b>	22
<b>2.3 Base de datos</b>	23
<b>2.3.1 Tipos de bases de datos</b>	24
<b>2.3.2 Modelos de bases de datos</b>	25
<b>2.3.3 Gestión de base de datos</b>	28
<b>3 MARCO EMPIRICO</b>	30
<b>3.1.1 METODOLOGIA DE LA INVESTIGACION</b>	30
<b>3.1.1.1 TIPO DE ESTUDIO</b>	30
<b>3.1.1.1.1 Documental o Bibliográfica</b>	30
<b>3.1.1.1.2 De Campo</b>	30
<b>3.1.2 PROCEDIMIENTOS PARA LA OBTENCION DE DATOS</b>	31
<b>3.1.2.1 Instrumento</b>	31
<b>3.1METODOLOGIA DESCRIPTIVA</b>	30
<b>4.0 MARCO OPERATIVO</b>	32
<b>4.1 Análisis del Diseño</b>	32
<b>4.1.1 Decisión de Requerimientos</b>	32
<b>4.1.2 Diagramas de casos de uso</b>	37
<b>4.1.3 ESPECIFICACION DE CASOS DE USO</b>	40
<b>4.1.4 Estructura de base de datos</b>	50
<b>Bibliografía</b>	51

## **1.1 DATOS DE IDENTIFICACION**

## **1.2 OBJETIVOS**

### **Objetivo Gral.**

Crear la B.D que cumpla con los campos necesarios para almacenar a todos los aspirantes a realizar pasantías y así obtener un mejor control de estos.

### **Objetivos específicos:**

\* Almacenar a todos los aspirantes a pasantías de acuerdo a su perfil de preparación.

\* Facilitar al aspirante a realizar pasantías a poder dar a conocer su currículo para ser tomado en cuenta en la asignación de estas.

## **1.3 INTRODUCCION**

En el presente documento se explica brevemente en qué consiste el Control de pasantías realizado por la Universidad Pedagógica de El Salvador, y como este, puede ser innovado con la tecnología haciendo uso de un Sistema informático para el control de dichos proyectos, muestra bibliográficamente conceptos y conocimientos básicos para comprender la importancia de dicho sistema. Muestra porque son importantes y de mucha utilidad los sistemas informáticos, sus principales componentes y características y como se desarrolla y lleva a cabo el presente sistema, las técnicas de investigación utilizadas, y la obtención de datos.

## **1.4 DEFINICION DEL PROBLEMA**

Un método de inserción laboral y adquisición de experiencia en el ámbito académico estudiado es el programa de pasantías que la Universidad Pedagógica ha desarrollado desde hace varios años, aunque el programa es exitoso la forma en que se llevan los registros y datos necesarios son un tanto rústicos, aunque llevan un almacenamiento de

los datos, lo cual incluye los curriculum de los alumnos interesados en realizar dichas prácticas, no hay manera de poder consultar datos específicos basándose en parámetros delimitados para así lograr la selección más acertada y que más se apegue al perfil requerido de la empresa solicitante, debido a que no es una base de datos formal. Cada curriculum es un archivo de Word, y son almacenados en una carpeta, para revisarlos hay que abrir cada documento.

Por otra parte el pasante para poder aplicar a una oferta debe enviar su curriculum por medio de correo electrónico, al encargado de su carrera, el cual será quien evalúe si cumple o no con los requerimientos.

Las ofertas de pasantías no son publicadas en un lugar específico, más bien cada encargado de carrera debe hacerlas públicas a los estudiantes por diferentes medios como: enviar mails, invitar a todos los alumnos de una clase o comentarlo con otros catedráticos para que ellos sugieran determinados estudiantes que para su criterio pueden llenar los requisitos.

## **1.5 JUSTIFICACION**

Esta investigación se realiza para mostrar una opción de la forma que se lleva el control en el Proyecto de pasantías de la Universidad Pedagógica de El Salvador para que pueda facilitar su asignación desarrollo y seguimiento, haciendo uso de las tecnologías actuales, y que llegue a ser una ayuda que beneficie tanto a los organizadores y encargados de dichos proyectos de pasantías como a los alumnos interesados en formar parte de dichos proyectos, con la presente investigación se busca que la información proporcionada tanto por administradores como por pasantes sea integra y completa, que la facilidad y accesibilidad de promover y aplicar para realizar una pasantía sea mucho mayor a la que se realiza actualmente, que las consultas y búsqueda de información sean más concretas y así exista una mejor selección y mayores oportunidades de acertar con la elección de ambas partes: pasantes-pasantías, entre otras.

## **2. FUNDAMENTACION TEORICA**

### **2.1. Sistemas**

#### **2.1.1. Teoría de sistemas**

La teoría de sistemas (TS) es un ramo específico de la teoría general de sistemas (TGS).

La TGS surgió con los trabajos del alemán Ludwig von Bertalanffy, publicados entre 1950 y 1968. La TGS no busca solucionar problemas o intentar soluciones prácticas, pero sí producir teorías y formulaciones conceptuales que pueden crear condiciones de aplicación en la realidad empírica.

Los supuestos básicos de la TGS son:

1. Existe una nítida tendencia hacia la integración de diversas ciencias naturales y sociales.
2. Esa integración parece orientarse rumbo a una teoría de sistemas.
3. Dicha teoría de sistemas puede ser una manera más amplia de estudiar los campos no físicos del conocimiento científico, especialmente en ciencias sociales.
4. Con esa teoría de los sistemas, al desarrollar principios unificadores que atraviesan verticalmente los universos particulares de las diversas ciencias involucradas, nos aproximamos al objetivo de la unidad de la ciencia.
5. Esto puede generar una integración muy necesaria en la educación científica.

La TGS afirma que las propiedades de los sistemas, no pueden ser descritos en términos de sus elementos separados; su comprensión se presenta cuando se estudian globalmente.

La TGS se fundamenta en tres premisas básicas:

1. Los sistemas existen dentro de sistemas: cada sistema existe dentro de otro más grande.
2. Los sistemas son abiertos: es consecuencia del anterior. Cada sistema que se examine, excepto el menor o mayor, recibe y descarga algo en los otros sistemas, generalmente en los contiguos. Los sistemas abiertos se caracterizan por un

proceso de cambio infinito con su entorno, que son los otros sistemas. Cuando el intercambio cesa, el sistema se desintegra, esto es, pierde sus fuentes de energía.

3. Las funciones de un sistema dependen de su estructura: para los sistemas biológicos y mecánicos esta afirmación es intuitiva. Los tejidos musculares por ejemplo, se contraen porque están constituidos por una estructura celular que permite contracciones.

El interés de la TGS, son las características y parámetros que establece para todos los sistemas. Aplicada a la administración la TS, la empresa se ve como una estructura que se reproduce y se visualiza a través de un sistema de toma de decisiones, tanto individual como colectivamente.

Desde un punto de vista histórico, se verifica que:

- ✓ La teoría de la administración científica usó el concepto de sistema hombre-máquina, pero se limitó al nivel de trabajo fabril.
- ✓ La teoría de las relaciones humanas amplió el enfoque hombre-máquina a las relaciones entre las personas dentro de la organización. Provocó una profunda revisión de criterios y técnicas gerenciales.
- ✓ La teoría estructuralista concibe la empresa como un sistema social, reconociendo que hay tanto un sistema formal como uno informal dentro de un sistema total integrado.
- ✓ La teoría del comportamiento trajo la teoría de la decisión, donde la empresa se ve como un sistema de decisiones, ya que todos los participantes de la empresa toman decisiones dentro de una maraña de relaciones de intercambio, que caracterizan al comportamiento organizacional.
- ✓ Después de la segunda guerra mundial, a través de la teoría matemática se aplicó la investigación operacional, para la resolución de problemas grandes y complejos con muchas variables.
- ✓ La teoría de colas fue profundizada y se formularon modelos para situaciones típicas de prestación de servicios, en los que es necesario programar la cantidad óptima de servidores para una esperada afluencia de clientes.

Las teorías tradicionales han visto la organización humana como un sistema cerrado. Eso ha llevado a no tener en cuenta el ambiente, provocando poco desarrollo y comprensión de la retroalimentación (feedback), básica para sobrevivir.

El enfoque antiguo fue débil, ya que

- 1) trató con pocas de las variables significantes de la situación total y,
- 2) muchas veces se ha sustentado con variables impropias.

El concepto de sistemas no es una tecnología en sí, pero es la resultante de ella. El análisis de las organizaciones vivas revela "lo general en lo particular" y muestra, las propiedades generales de las especies que son capaces de adaptarse y sobrevivir en un ambiente típico. Los sistemas vivos sean individuos o organizaciones, son analizados como "sistemas abiertos", que mantienen un continuo intercambio de materia/energía/información con el ambiente.

La TS permite re conceptualizar los fenómenos dentro de un enfoque global, para integrar asuntos que son, en la mayoría de las veces de naturaleza completamente diferente.

### **2.1.2. Concepto de sistemas**

- Un conjunto de elementos
- Dinámicamente relacionados
- Formando una actividad
- Para alcanzar un objetivo
- Operando sobre datos/energía/materia
- Para proveer información/energía/materia

Un sistema es un objeto compuesto, cuyos componentes se relacionan con al menos algún otro componente; puede ser material o conceptual. Todos los sistemas tienen composición, estructura y entorno, pero sólo los sistemas materiales tienen mecanismo, y sólo algunos sistemas materiales tienen figura (forma). Según el sistemismo, todos los objetos son sistemas o componentes de algún sistema. Por ejemplo, un núcleo atómico es un sistema material físico compuesto de protones y neutrones relacionados por la interacción nuclear fuerte; una molécula es un sistema material químico compuesto de átomos relacionados por enlaces químicos; una célula es un sistema material biológico compuesto de orgánulos relacionados por enlaces químicos no-covalentes y rutas metabólicas; una corteza cerebral es un sistema material psicológico (mental) compuesto de neuronas relacionadas por potenciales de acción y neurotransmisores; un ejército es un sistema material social y parcialmente artificial compuesto de personas y artefactos relacionados por el mando, el abastecimiento, la comunicación y la guerra; el anillo de los números enteros es un sistema conceptual algebraico compuesto de números positivos,

negativos y el cero relacionados por la suma y la multiplicación; y una teoría científica es un sistema conceptual lógico compuesto de hipótesis, definiciones y teoremas relacionados por la correferencia y la deducción (implicación).

### **2.1.3. Características de los sistemas**

Sistema es un todo organizado y complejo; un conjunto o combinación de cosas o partes que forman un todo complejo o unitario. Es un conjunto de objetos unidos por alguna forma de interacción o interdependencia. Los límites o fronteras entre el sistema y su ambiente admiten cierta arbitrariedad.

Sistema es un conjunto de unidades recíprocamente relacionadas. De ahí se deducen dos conceptos: propósito (u objetivo) y globalismo (o totalidad).

- Propósito u objetivo: todo sistema tiene uno o algunos propósitos. Los elementos (u objetos), como también las relaciones, definen una distribución que trata siempre de alcanzar un objetivo.
- Globalismo o totalidad: un cambio en una de las unidades del sistema, con probabilidad producirá cambios en las otras. El efecto total se presenta como un ajuste a todo el sistema. Hay una relación de causa/efecto. De estos cambios y ajustes, se derivan dos fenómenos: entropía y homeostasia.
- Entropía: es la tendencia de los sistemas a desgastarse, a desintegrarse, para el relajamiento de los estándares y un aumento de la aleatoriedad. La entropía aumenta con el correr del tiempo. Si aumenta la información, disminuye la entropía, pues la información es la base de la configuración y del orden. De aquí nace la negentropía, o sea, la información como medio o instrumento de ordenación del sistema.
- Homeostasia: es el equilibrio dinámico entre las partes del sistema. Los sistemas tienen una tendencia a adaptarse con el fin de alcanzar un equilibrio interno frente a los cambios externos del entorno.

Una organización podrá ser entendida como un sistema o subsistema o un supersistema, dependiendo del enfoque. El sistema total es aquel representado por todos los componentes y relaciones necesarios para la realización de un objetivo, dado un cierto número de restricciones. Los sistemas pueden operar, tanto en serie como en paralelo.

#### 2.1.4. Sistema de información

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:

- El equipo computacional, formado por el hardware necesario para que el sistema de información pueda operar.
- El recurso humano que interactúa con el Sistema de Información, el cual está formado por las personas que utilizan el sistema.
- Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información.

**Entrada de Información:** Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos. Esto último se denomina interfaces automáticas. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, las unidades de disco, los códigos de barras, los escáner, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.

**Almacenamiento de información:** El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros, los discos compactos (CD-ROM), las unidades enclóster, entre otros.

**Procesamiento de Información:** Es la capacidad del Sistema de Información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere, por ejemplo, una

proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

**Salida de Información:** La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, discos, cintas magnéticas, USB, la voz, los graficadores y los plotters, entre otros. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro Sistema de Información o módulo. En este caso, también existe una interface automática de salida. Por ejemplo, el Sistema de Control de Clientes tiene una interface automática de salida con el Sistema de Contabilidad, ya que genera las pólizas contables de los movimientos procesales de los clientes.

A continuación se muestran las diferentes actividades que puede realizar un Sistema de Información de Control de Clientes:

#### **2.1.4.1. Actividades que realiza un Sistema de Información:**

##### **Entradas:**

- ✓ Datos generales del cliente: nombre, dirección, tipo de cliente, etc.
- ✓ Políticas de créditos: límite de crédito, plazo de pago, etc.
- ✓ Facturas (interface automático).
- ✓ Pagos, depuraciones, etc.

##### **Proceso:**

- ✓ Cálculo de antigüedad de saldos.
- ✓ Cálculo de intereses moratorios.
- ✓ Cálculo del saldo de un cliente.

##### **Almacenamiento:**

- ✓ Movimientos del mes (pagos, depuraciones).
- ✓ Catálogo de clientes.
- ✓ Facturas.

##### **Salidas:**

- ✓ Reporte de pagos.
- ✓ Estados de cuenta.

- ✓ Pólizas contables (interface automática)
- ✓ Consultas de saldos en pantalla de una terminal.

#### **2.1.4.2. Tipos y Usos de los Sistemas de Información**

Durante los próximos años, los Sistemas de Información cumplirán tres objetivos básicos dentro de las organizaciones:

1. Automatización de procesos operativos.
2. Proporcionar información que sirva de apoyo al proceso de toma de decisiones.
3. Lograr ventajas competitivas.

Los Sistemas de Información que logran la automatización de procesos operativos dentro de una organización, son llamados frecuentemente Sistemas Transaccionales, ya que su función primordial consiste en procesar transacciones tales como pagos, cobros, pólizas, entradas, salidas, etc. Por otra parte, los Sistemas de Información que apoyan el proceso de toma de decisiones son los Sistemas de Soporte a la Toma de Decisiones, Sistemas para la Toma de Decisión de Grupo, Sistemas Expertos de Soporte a la Toma de Decisiones y Sistema de Información para Ejecutivos. El tercer tipo de sistema, de acuerdo con su uso u objetivos que cumplen, es el de los Sistemas Estratégicos, los cuales se desarrollan en las organizaciones con el fin de lograr ventajas competitivas, a través del uso de la tecnología de información.

A continuación se mencionan las principales características de estos tipos de Sistemas de Información.

#### **2.1.4.3. Sistemas Transaccionales.**

Sus principales características son:

- A través de éstos suelen lograrse ahorros significativos de mano de obra, debido a que automatizan tareas operativas de la organización.
- Con frecuencia son el primer tipo de Sistemas de Información que se implanta en las organizaciones. Se empieza apoyando las tareas a nivel operativo de la organización.
- Son intensivos en entrada y salida de información; sus cálculos y procesos suelen ser simples y poco sofisticados.
- Tienen la propiedad de ser recolectores de información, es decir, a través de estos sistemas se cargan las grandes bases de información para su explotación posterior.

- Son fáciles de justificar ante la dirección general, ya que sus beneficios son visibles y palpables.

#### **2.1.4.4. Sistemas de Apoyo de las Decisiones.**

Las principales características de estos son:

- Suelen introducirse después de haber implantado los Sistemas Transaccionales más relevantes de la empresa, ya que estos últimos constituyen su plataforma de información.
- La información que generan sirve de apoyo a los mandos intermedios y a la alta administración en el proceso de toma de decisiones.
- Suelen ser intensivos en cálculos y escasos en entradas y salidas de información. Así, por ejemplo, un modelo de planeación financiera requiere poca información de entrada, genera poca información como resultado, pero puede realizar muchos cálculos durante su proceso.
- No suelen ahorrar mano de obra. Debido a ello, la justificación económica para el desarrollo de estos sistemas es difícil, ya que no se conocen los ingresos del proyecto de inversión.
- Suelen ser Sistemas de Información interactivos y amigables, con altos estándares de diseño gráfico y visual, ya que están dirigidos al usuario final.
- Apoyan la toma de decisiones que, por su misma naturaleza son repetitivos y de decisiones no estructuradas que no suelen repetirse. Por ejemplo, un Sistema de Compra de Materiales que indique cuándo debe hacerse un pedido al proveedor o un Sistema de Simulación de Negocios que apoye la decisión de introducir un nuevo producto al mercado.
- Estos sistemas pueden ser desarrollados directamente por el usuario final sin la participación operativa de los analistas y programadores del área de informática.

Este tipo de sistemas puede incluir la programación de la producción, compra de materiales, flujo de fondos, proyecciones financieras, modelos de simulación de negocios, modelos de inventarios, etc.

#### **2.1.4.5. Sistemas Estratégicos.**

Sus principales características son:

- Su función primordial no es apoyar la automatización de procesos operativos ni proporcionar información para apoyar la toma de decisiones.

- Suelen desarrollarse in house, es decir, dentro de la organización, por lo tanto no pueden adaptarse fácilmente a paquetes disponibles en el mercado.
- Típicamente su forma de desarrollo es a base de incrementos y a través de su evolución dentro de la organización. Se inicia con un proceso o función en particular y a partir de ahí se van agregando nuevas funciones o procesos.
- Su función es lograr ventajas que los competidores no posean, tales como ventajas en costos y servicios diferenciados con clientes y proveedores. En este contexto, los Sistemas Estratégicos son creadores de barreras de entrada al negocio. Por ejemplo, el uso de cajeros automáticos en los bancos en un Sistema Estratégico, ya que brinda ventaja sobre un banco que no posee tal servicio. Si un banco nuevo decide abrir su puerta al público, tendrá que dar este servicio para tener un nivel similar al de sus competidores.
- Apoyan el proceso de innovación de productos y proceso dentro de la empresa debido a que buscan ventajas respecto a los competidores y una forma de hacerlo en innovando o creando productos y procesos.

Un ejemplo de estos Sistemas de Información dentro de la empresa puede ser un sistema MRP (Manufacturing Resource Planning) enfocado a reducir sustancialmente el desperdicio en el proceso productivo, o bien, un Centro de Información que proporcione todo tipo de información; como situación de créditos, embarques, tiempos de entrega, etc. En este contexto los ejemplos anteriores constituyen un Sistema de Información Estratégico si y sólo si, apoyan o dan forma a la estructura competitiva de la empresa.

Por último, es importante aclarar que algunos autores consideran un cuarto tipo de sistemas de información denominado Sistemas Personales de Información, el cual está enfocado a incrementar la productividad de sus usuarios.

#### **2.1.4.6. Evolución de los Sistemas de Información**

De la sección anterior se desprende la evolución que tienen los Sistemas de Información en las organizaciones. Con frecuencia se implantan en forma inicial los Sistemas Transaccionales y, posteriormente, se introducen los Sistemas de Apoyo a las Decisiones. Por último, se desarrollan los Sistemas Estratégicos que dan forma a la estructura competitiva de la empresa.

La función de la Informática en las organizaciones evoluciona a través de ciertas etapas de crecimiento, las cuales se explican a continuación:

- Comienza con la adquisición de la primera computadora y normalmente se justifica por el ahorro de mano de obra y el exceso de papeles.
- Las aplicaciones típicas que se implantan son los Sistemas Transaccionales tales como nóminas o contabilidad.

- El pequeño Departamento de Sistemas depende en la mayoría de los casos del área de contabilidad.
- El tipo de administración empleada es escaso y la función de los sistemas suele ser manejada por un administrador que no posee una preparación formal en el área de computación.
- El personal que labora en este pequeño departamento consta a lo sumo de un operador y/o un programador. Este último podrá estar bajo el régimen de honorarios, o bien, puede recibirse el soporte de algún fabricante local de programas de aplicación.
- En esta etapa es importante estar consciente de la resistencia al cambio del personal y usuario (ciberfobia) que están involucrados en los primeros sistemas que se desarrollan, ya que estos sistemas son importantes en el ahorro de mano de obra.
- Esta etapa termina con la implantación exitosa del primer Sistema de Información. Cabe recalcar que algunas organizaciones pueden vivir varias etapas de inicio en las que la resistencia al cambio por parte de los primeros usuarios involucrados aborta el intento de introducir la computadora a la empresa.

Etapa de contagio o expansión. Los aspectos sobresalientes que permiten diagnosticar rápido que una empresa se encuentra en esta etapa son:

- Se inicia con la implantación exitosa del primer Sistema de Información en la organización. Como consecuencia de lo anterior, el primer ejecutivo usuario se transforma en el paradigma o persona que se habrá que imitar.
- Las aplicaciones que con frecuencia se implantan en esta etapa son el resto de los Sistemas Transaccionales no desarrollados en la etapa de inicio, tales como facturación, inventarios, control de pedidos de clientes y proveedores, cheques, etc.
- El pequeño departamento es promovido a una categoría superior, donde depende de la Gerencia Administrativa o Contraloría. El tipo de administración empleado está orientado hacia la venta de aplicaciones a todos los usuarios de la organización; en este punto suele contratarse a un especialista de la función con preparación académica en el área de sistemas.
- Se inicia la contratación de personal especializado y nacen puestos tales como analista de sistemas, analista-programador, programador de sistemas, jefe de desarrollo, jefe de soporte técnico, etc.
- Las aplicaciones desarrolladas carecen de interfaces automáticas entre ellas, de tal forma que las salidas que produce un sistema se tienen que alimentar en forma manual a otro sistema, con la consecuente irritación de los usuarios.

- Los gastos por concepto de sistemas empiezan a crecer en forma importante, lo que marca la pauta para iniciar la racionalización en el uso de los recursos computacionales dentro de la empresa. Este problema y el inicio de su solución marcan el paso a la siguiente etapa.

Etapa de control o formalización. Para identificar a una empresa que transita por esta etapa es necesario considerar los siguientes elementos:

- Esta etapa de evolución de la Informática dentro de las empresas se inicia con la necesidad de controlar el uso de los recursos computacionales a través de las técnicas de presupuestación base cero (partiendo de que no se tienen nada) y la implantación de sistemas de cargos a usuarios (por el servicio que se presta).
- Las aplicaciones están orientadas a facilitar el control de las operaciones del negocio para hacerlas más eficaces, tales como sistemas para control de flujo de fondos, control de órdenes de compra a proveedores, control de inventarios, control y manejo de proyectos, etc.
- El departamento de sistemas de la empresa suele ubicarse en una posición gerencial, dependiendo del organigrama de la Dirección de Administración o Finanzas.
- El tipo de administración empleado dentro del área de Informática se orienta al control administrativo y a la justificación económica de las aplicaciones a desarrollar. Nace la necesidad de establecer criterios para las prioridades en el desarrollo de nuevas aplicaciones. La cartera de aplicaciones pendientes por desarrollar empieza a crecer.
- En esta etapa se inician el desarrollo y la implantación de estándares de trabajo dentro del departamento, tales como: estándares de documentación, control de proyectos, desarrollo y diseño de sistemas, auditoría de sistemas y programación.
- Se integra a la organización del departamento de sistemas, personal con habilidades administrativas y preparadas técnicamente.
- Se inicia el desarrollo de interfaces automáticas entre los diferentes sistemas.

Etapa de integración. Las características de esta etapa son las siguientes:

- La integración de los datos y de los sistemas surge como un resultado directo de la centralización del departamento de sistemas bajo una sola estructura administrativa.
- Las nuevas tecnologías relacionadas con base de datos, sistemas administradores de bases de datos y lenguajes de cuarta generación, hicieron posible la integración.

- En esta etapa surge la primera hoja electrónica de cálculo comercial y los usuarios inician haciendo sus propias aplicaciones. Esta herramienta ayudó mucho a que los usuarios hicieran su propio trabajo y no tuvieran que esperar a que sus propuestas de sistemas fueran cumplidas.
- El costo del equipo y del software disminuyó por lo cual estuvo al alcance de más usuarios.
- En forma paralela a los cambios tecnológicos, cambió el rol del usuario y del departamento de Sistemas de Información. El departamento de sistemas evolucionó hacia una estructura descentralizada, permitiendo al usuario utilizar herramientas para el desarrollo de sistemas.
- Los usuarios y el departamento de sistema iniciaron el desarrollo de nuevos sistemas, reemplazando los sistemas antiguos, en beneficio de la organización.

Etapa de administración de datos. Entre las características que destacan en esta etapa están las siguientes:

- El departamento de Sistemas de Información reconoce que la información es un recurso muy valioso que debe estar accesible para todos los usuarios.
- Para poder cumplir con lo anterior resulta necesario administrar los datos en forma apropiada, es decir, almacenarlos y mantenerlos en forma adecuada para que los usuarios puedan utilizar y compartir este recurso.
- El usuario de la información adquiere la responsabilidad de la integridad de la misma y debe manejar niveles de acceso diferentes.

Etapa de madurez. Entre los aspectos sobresalientes que indican que una empresa se encuentra en esta etapa, se incluyen los siguientes:

- Al llegar a esta etapa, la Informática dentro de la organización se encuentra definida como una función básica y se ubica en los primeros niveles del organigrama (dirección).
- Los sistemas que se desarrollan son Sistemas de Manufactura Integrados por Computadora, Sistemas Basados en el Conocimiento y Sistemas Expertos, Sistemas de Soporte a las Decisiones, Sistemas Estratégicos y, en general, aplicaciones que proporcionan información para las decisiones de alta administración y aplicaciones de carácter estratégico.
- En esta etapa se tienen las aplicaciones desarrolladas en la tecnología de base de datos y se logra la integración de redes de comunicaciones con terminales en lugares remotos, a través del uso de recursos computacionales.

### **2.1.5. Clasificación de los sistemas**

La clasificación de un sistema al igual que el análisis de los aspectos del mismo es un proceso subjetivo; depende del individuo que lo hace, del objetivo que se persigue y de las circunstancias particulares en las cuales se desarrolla. En este punto se dan lineamientos generales sobre las diferentes clases de sistemas y algunos ejemplos que corresponden a su definición, pero puede haber debate sobre los mismos si se tiene en cuenta las consideraciones expuestas antes.

De acuerdo con algunos planteamientos relacionados a sistemas, estos se clasifican así:

#### **Según su relación con el medio ambiente:**

- Sistemas abiertos: Sistema que intercambia materia, energía o información con el ambiente, ejemplos: Célula, ser humano, ciudad, perro, televisor, familia, estación de radio
- Sistemas cerrados: Sistema que no intercambia materia, energía o información con el ambiente, Ejemplos: Universo, reloj desechable, llanta de carro

#### **Según su naturaleza:**

- Sistemas concretos: Sistema físico o tangible, ejemplos: Equipo de sonido, edificio, pájaro, guitarra, elefante.
- Sistemas abstractos: Sistema simbólico o conceptual, ejemplos: Sistema hexadecimal, idioma español, lógica difusa.

#### **Según su origen:**

- Sistemas naturales: Sistema generado por la naturaleza, ejemplos: Río, bosque, molécula de agua.
- Sistemas artificiales: Sistema producto de la actividad humana; son concebidos y construidos por el hombre, ejemplos: Tren, avión, marcapasos, idioma inglés.

#### **Según sus relaciones:**

- Sistemas simples: Sistema con pocos elementos y relaciones, ejemplos: Juego de billar, péndulo,  $f(x) = x + 1$ , palanca.
- Sistemas complejos: Sistema con numerosos elementos y relaciones entre ellos, ejemplos: Cerebro, universidad, cámara fotográfica.

Esta clasificación es relativa porque depende del número de elementos y relaciones considerados. En la práctica y con base en límites psicológicos de la percepción y comprensión humanas, un sistema con más o menos siete elementos y relaciones se puede considerar simple.

**Según su cambio en el tiempo:**

- Sistemas estáticos: Sistema que no cambia en el tiempo, ejemplos: Piedra, vaso de plástico, montaña.
- Sistemas dinámicos: Sistema que cambia en el tiempo, ejemplos: Universo, átomo, la tierra, hongo.

Esta clasificación es relativa porque depende del periodo de tiempo definido para el análisis del sistema.

**Según el tipo de variables que lo definen:**

- Sistemas discretos: Sistema definido por variables discretas, ejemplos: lógica booleana, alfabeto.
- Sistemas continuos: Sistema definido por variables continuas, ejemplos: alternador, río.

**Otras clasificaciones:**

- Sistemas jerárquicos: Sistema cuyos elementos están relacionados mediante relaciones de dependencia o subordinación conformando una organización por niveles. Chiavenato (1999) los denomina sistemas piramidales, Ejemplos: Gobierno de una ciudad.
- Sistemas de control: Sistema jerárquico en el cual unos elementos son controlados por otros, Ejemplos: Lámpara
- Sistemas de control con retroalimentación: Sistema de control en el cual los elementos controlados envían información sobre su estado a los elementos controladores, Ejemplos: Termostato
- Sistemas determinísticos: Sistema con un comportamiento previsible, Ejemplos: Palanca, polea, programa de computador
- Sistemas probabilísticos: Sistema con un comportamiento no previsible, Ejemplos: Clima, mosca, sistema económico mundial.

## 2.2. Proceso Unificado

El Proceso Unificado es un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de competencia y diferentes tamaños de proyectos.

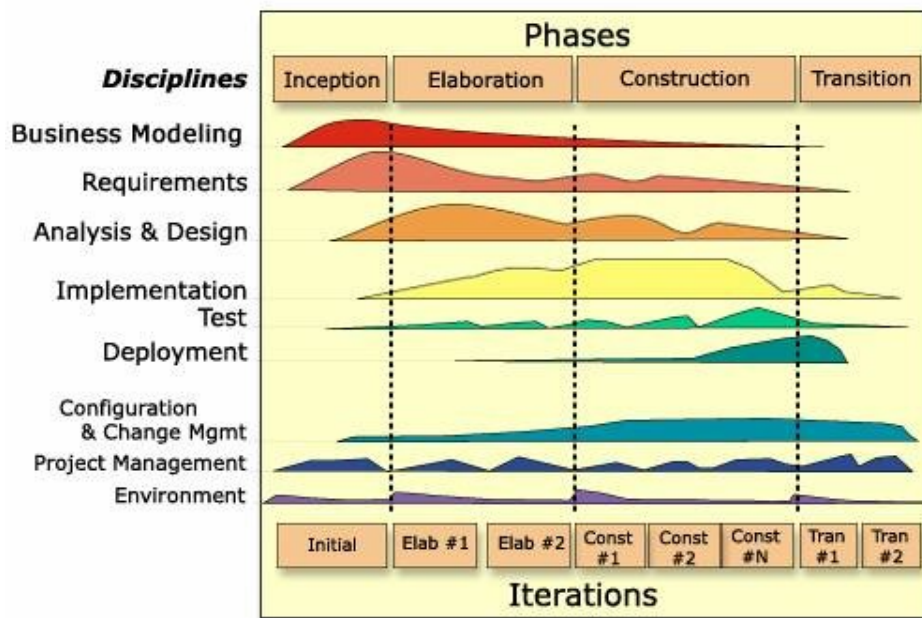
Provee un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible.

El Proceso Unificado tiene dos dimensiones (**Figura 1**):

- Un eje horizontal que representa el tiempo y muestra los aspectos del ciclo de vida del proceso a lo largo de su desenvolvimiento
- Un eje vertical que representa las disciplinas, las cuales agrupan actividades de una manera lógica de acuerdo a su naturaleza.

La primera dimensión representa el aspecto dinámico del proceso conforme se va desarrollando, se expresa en términos de fases, iteraciones e hitos (milestones).

La segunda dimensión representa el aspecto estático del proceso: cómo es descrito en términos de componentes del proceso, disciplinas, actividades, flujos de trabajo, artefactos y roles.



## **Figura 1**

El Proceso Unificado se basa en componentes (component-based), lo que significa que el sistema en construcción está hecho de componentes de software interconectados por medio de interfaces bien definidas (well-defined interfaces).

El Proceso Unificado usa el Lenguaje de Modelado Unificado (UML) en la preparación de todos los planos del sistema. De hecho, UML es una parte integral del Proceso Unificado, fueron desarrollados a la par.

Los aspectos distintivos del Proceso Unificado están capturados en tres conceptos clave: dirigido por casos de uso (use-case driven), centrado en la arquitectura (architecture-centric), iterativo e incremental. Esto es lo que hace único al Proceso Unificado.

### **2.2.1. Características del Proceso Unificado**

#### ***El Proceso Unificado es dirigido por casos de uso***

Un sistema de software se crea para servir a sus usuarios. Por lo tanto, para construir un sistema exitoso se debe conocer qué es lo que quieren y necesitan los usuarios prospectos.

El término usuario se refiere no solamente a los usuarios humanos, sino a otros sistemas. En este contexto, el término usuario representa algo o alguien que interactúa con el sistema por desarrollar.

Un *caso de uso* es una pieza en la funcionalidad del sistema que le da al usuario un resultado de valor. Los casos de uso capturan los requerimientos funcionales. Todos los casos de uso juntos constituyen el *modelo de casos de uso* el cual describe la funcionalidad completa del sistema. los casos de uso no son solamente una herramienta para especificar los requerimientos del sistema, también dirigen su diseño, implementación y pruebas, esto es, dirigen el proceso de desarrollo.

Aún y cuando los casos de uso dirigen el proceso, no son elegidos de manera aislada. Son desarrollados a la par con la arquitectura del sistema, esto es, los casos de uso dirigen la arquitectura del sistema y la arquitectura del sistema influencia la elección de los casos de uso. Por lo tanto, al arquitectura del sistema y los casos de uso maduran conforme avanza el ciclo de vida.

#### ***El Proceso Unificado está centrado en la arquitectura***

El papel del arquitecto de sistemas es similar en naturaleza al papel que el arquitecto desempeña en la construcción de edificios. El edificio se mira desde diferentes

puntos de vista: estructura, servicios, plomería, electricidad, etc. Esto le permite al constructor ver una radiografía completa antes de empezar a construir. Similarmente, la arquitectura en un sistema de software es descrita como diferentes vistas del sistema que está siendo construido.

El concepto de arquitectura de software involucra los aspectos estáticos y dinámicos más significativos del sistema. La arquitectura surge de las necesidades de la empresa, tal y como las interpretan los usuarios y otros stakeholders, y tal y como están reflejadas en los casos de uso. Sin embargo, también está influenciada por muchos otros factores, tales como la plataforma de software en la que se ejecutará, la disponibilidad de componentes reutilizables, consideraciones de instalación, sistemas legados, requerimientos no funcionales (ej. desempeño, confiabilidad). La arquitectura es la vista del diseño completo con las características más importantes hechas más visibles y dejando los detalles de lado. Ya que lo importante depende en parte del criterio, el cual a su vez viene con la experiencia, el valor de la arquitectura depende del personal asignado a esta tarea. Sin embargo, el proceso ayuda al arquitecto a enfocarse en las metas correctas, tales como claridad (understandability), flexibilidad en los cambios futuros (resilience) y reúso.

¿Cómo se relacionan los casos de uso con la arquitectura? Cada producto tiene función y forma. Uno sólo de los dos no es suficiente. Estas dos fuerzas deben estar balanceadas para obtener un producto exitoso. En este caso función corresponde a los casos de uso y forma a la arquitectura. Existe la necesidad de intercalar entre casos de uso y arquitectura. Es un problema del “huevo y la gallina”. Por una parte, los casos de uso deben, cuando son realizados, acomodarse en la arquitectura. Por otra parte, la arquitectura debe proveer espacio para la realización de todos los casos de uso, hoy y en el futuro. En la realidad, ambos arquitectura y casos de uso deben evolucionar en paralelo.

### ***El Proceso Unificado es Iterativo e Incremental***

Desarrollar un producto de software comercial es una tarea enorme que puede continuar por varios meses o años. Es práctico dividir el trabajo en pequeños pedazos o mini-proyectos. Cada mini-proyecto es una iteración que finaliza en un incremento. Las iteraciones se refieren a pasos en el flujo de trabajo, los incrementos se refieren a crecimiento en el producto. Para ser más efectivo, las iteraciones deben estar controladas, esto es, deben ser seleccionadas y llevadas a cabo de una manera planeada.

Los desarrolladores basan su selección de qué van a implementar en una iteración en dos factores. Primero, la iteración trata con un grupo de casos de uso que en conjunto extienden la usabilidad del producto. Segundo, la iteración trata con los riesgos más

importantes. Las iteraciones sucesivas construyen los artefactos del desarrollo a partir del estado en el que fueron dejados en la iteración anterior.

En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, crean el diseño usando la arquitectura como guía, implementan el diseño en componentes y verifican que los componentes satisfacen los casos de uso. Si una iteración cumple sus metas – y usualmente lo hace – el desarrollo continúa con la siguiente iteración. Cuando la iteración no cumple con sus metas, los desarrolladores deben revisar sus decisiones previas y probar un nuevo enfoque.

### **2.2.2. Las 4 “P” del Proceso Unificado**

#### **PROYECTO.**

- Elemento organizativo a través del cual se gestiona el desarrollo de software.
- El resultado de un proyecto es una versión de un producto.
- Elemento organizativo de gestión
- El proyecto construye el producto
- Secuencia de cambio: El sistema evoluciona
- Serie de iteraciones: Cada iteración implementa un conjunto de casos de uso o atenúa algunos riesgos. Mini-proyecto
- Patrón organizativo: Tipos de trabajadores y artefactos a conseguir.

#### **PROCESO.**

- Un proceso de ingeniería de software es una definición del conjunto de actividades necesarias para transformar los requisitos de usuario en un producto.
- Un proceso es una plantilla para crear proyectos
- Conjunto de actividades para crear el producto
- Es una plantilla para crear proyectos
- Se define en términos de flujos de trabajo
- (conjunto de actividades)
- Se identifican trabajadores y artefactos
- Adaptación o especialización del proceso
- Se utilizan diagramas de actividad de UML para describir los flujos de trabajo

#### **PRODUCTO.**

- Artefactos que se crean durante la vida del proyecto, como los modelos, código fuente, ejecutables, y documentación

- El resultado de llevar a cabo un proceso software dentro de un proyecto concreto
- Artefactos que se crean durante la vida del proyecto
- Modelos, códigos, ejecutables, documentación, diagramas UML, bocetos de la interfaz de usuario, prototipos, componentes, planes de prueba Ingeniería y gestión
- Colección de modelos
- Modelo de casos de uso, análisis, diseño, despliegue, implementación y prueba

## **PERSONAS.**

- Los principales autores de un proyecto de software son los arquitectos, desarrolladores, ingenieros de prueba y el personal de gestión que les da soporte, además de los usuarios, clientes, y otros interesados
- Las personas son realmente seres humanos, a diferencia del término abstracto “trabajadores”
- Personas
- Arquitectos, desarrolladores, ingenieros de prueba, personal de gestión, usuarios, clientes.
- El proceso de desarrollo afecta a las personas (viabilidad, gestión del riesgo, estructura de los equipos, planificación, comprensión, cumplimiento)
- Formación, entrenamiento y experiencia
- De recurso a trabajador (puestos que asumen las personas)
- Cada trabajador tiene un conjunto de responsabilidades y lleva a cabo un conjunto de actividades

### **2.2.3. Fases del Proceso Unificado**

Para el PU la vida de un sistema se encuentra dividida en **ciclos**. Cada ciclo termina con un lanzamiento de diferentes modelos del producto.

Cada ciclo consiste de cuatro **fases**:

- Concepción
- Elaboración
- Construcción
- Transición.

Cada fase se encuentra subdividida en **iteraciones**. Al final de cada fase se produce un **milestone**” o punto de revisión. Una iteración es un mini proyecto que concluye con la entrega de algún documento o implementación interna. Dentro de cada iteración se realizan actividades de captura de requerimientos, análisis, diseño, implementación y prueba.

- **Iniciación:** Establecer la visión, el alcance y el plan inicial del proyecto.
  - Objetivos del ciclo de vida
  - Establecer ámbito del sistema
  - Reducir peores riesgos
  - Preparar el análisis del negocio
  
- **Elaboración:** Diseñar, implementar y probar una arquitectura correcta, y completar el plan del proyecto.
  - Arquitectura del ciclo de vida
  - Obtener línea base de la arquitectura
  - Capturar mayoría de requisitos
  - Reducir siguientes riesgos
  
- **Construcción:** Diseñar, implementar y probar una arquitectura correcta, y completar el plan del proyecto.
  - Funcionalidad operativa inicial
  - Desarrollo del sistema entero
  
- **Transición:** Proporcionar el sistema a sus usuarios finales.
  - Versión del producto
  - Producto preparado para su entrega al usuario
  - Se enseña a los usuarios a utilizarlo

#### **2.2.4. Disciplinas del Proceso Unificado**

##### **Requisitos:**

- El objetivo es describir que es lo que tiene que hacer el sistema y poner a los desarrolladores y al cliente de acuerdo en esta descripción.

##### **Análisis y Diseño:**

- Describe como el software será realizado en la fase de implementación.

- Se plasma en un modelo de diseño que consiste en una serie de clases (agrupadas en paquetes y subsistemas) con interfaces bien definidos.
- También contiene descripciones de cómo los objetos colaboran para realizar las acciones incluidas en los casos de uso.

#### **Implementación:**

- Se implementan las clases y objetos en términos de componentes (ficheros fuentes, binario, ejecutables, etc.)

#### **Prueba:**

- Se comprueba que el funcionamiento es correcto analizando diversos aspectos: los objetos como unidades, la integración entre objetos, la implementación de todos los requisitos, etc.

#### **Despliegue:**

- Se crea la versión externa del producto, se empaqueta, se distribuye y se instala en el lugar de trabajo. También se da asistencia y ayuda a los usuarios.

#### **Gestión de proyectos:**

- Encargada de definir los planes del proyecto global, los planes de fase y los planes de iteración.

#### **Entorno:**

- Se centra en las actividades necesarias para configurar el proceso de un proyecto.
- El objetivo es proveer a la organización de desarrollo software de un entorno de trabajo (que incluye procedimientos y herramientas) que soporten al equipo de desarrollo.

## **2.3 Base de datos**

Una **base de datos** o **banco de datos** es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la **informática** y la **electrónica**, la mayoría de las bases de datos tienen formato electrónico, que ofrece un amplio rango de soluciones al problema de almacenar datos.

En informática existen los **sistemas gestores de bases de datos** (SGBD), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de los sistemas gestores de bases de datos se estudian en informática.

Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

Aunque las bases de datos pueden contener muchos tipos de datos, algunos de ellos se encuentran protegidos por las leyes de varios países.

### **2.3.1 Tipos de bases de datos**

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al criterio elegido para su clasificación:

#### ➤ **Según la variabilidad de los datos almacenados**

##### **Bases de datos estáticas**

Éstas son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar **proyecciones** y tomar **decisiones**.

##### **Bases de datos dinámicas**

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un videoclub, etc.

#### ➤ **Según el contenido**

##### **Bases de datos bibliográficas**

Solo contienen un surrogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque si no estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias—ver más abajo). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

##### **Bases de datos de texto completo**

Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

## Directorios

Un ejemplo son las guías telefónicas en formato electrónico.

**Banco de imágenes, audio, video, multimedia, etc.**

## Bases de datos o “bibliotecas” de información Biológica

Son bases de datos que almacenan diferentes tipos de información proveniente de las ciencias de la vida o médicas. Se pueden considerar en varios subtipos:

- Aquellas que almacenan secuencias de nucleótidos o proteínas.
- Las bases de datos de rutas metabólicas
- Bases de datos de estructura, comprende los registros de datos experimentales sobre estructuras 3D de biomoléculas
- Bases de datos clínicas
- Bases de datos bibliográficas (biológicas)

### 2.3.2 Modelos de bases de datos

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una “descripción” de algo conocido como *contenedor de datos* (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de *base de datos*; por lo general se refieren a **algoritmos**, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

#### ➤ Bases de datos Jerárquicas

Éstas son bases de datos que, como su nombre indica, almacenan su información en una estructura jerárquica. En este modelo los datos se organizan en una forma similar a un árbol (visto al revés), en donde un *nodo padre* de información puede tener varios *hijos*. El nodo que no tiene padres es llamado *raíz*, y a los nodos que no tienen hijos se los conoce como *hojas*.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

➤ Bases de datos de red

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de *nodo*: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

➤ Bases de datos relacional

Éste es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de “relaciones”. Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados “tuplas”. Pese a que ésta es la teoría de las bases de datos relacionales creadas por Edgar Frank Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante “consultas” que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años '80 (1980-1989) la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión.

➤ Bases de datos orientada a objetos

Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los *objetos* completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

- **Encapsulación** – Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.
- **Herencia** – Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.
- **Polimorfismo** – Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o *signatura*) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o *parámetros*). La implementación (o *método*) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

Se está trabajando en **SQL3**, que es el estándar de SQL92 ampliado, que soportará los nuevos conceptos orientados a objetos y mantendría compatibilidad con SQL92.

➤ Bases de datos documentales

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. Tesauro es un sistema de índices optimizado para este tipo de bases de datos.

➤ Base de datos deductivas

Un sistema de **base de datos deductivas**, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. También las bases de datos deductivas son llamadas **base de datos lógica**, a raíz de que se basan en lógica matemática.

### 2.3.3 Gestión de base de datos

Los **Sistemas de gestión de base de datos** son un tipo de software muy específico, dedicado a servir de interfaz entre la **base de datos**, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. En los textos que tratan este tema, o temas relacionados, se mencionan los términos SGBD y DBMS, siendo ambos equivalentes, y acrónimos, respectivamente, de Sistema Gestor de Bases de Datos y *DataBase Management System*, su expresión inglesa.

El propósito general de los sistemas de gestión de base de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos.

Existen distintos objetivos que deben cumplir los SGBD:

- **Abstracción de la información.** Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios *niveles de abstracción*.
- **Independencia.** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.
- **Redundancia mínima.** Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias.
- **Consistencia.** En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.
- **Seguridad.** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra asegurada frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.
- **Integridad.** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.

- **Respaldo y recuperación.** Los SGBD deben proporcionar una forma eficiente de realizar copias de seguridad de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.
- **Control de la concurrencia.** En la mayoría de entornos (excepto quizás el doméstico), lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es también frecuente que dichos accesos se realicen de forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.
- **Tiempo de respuesta.** Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en darnos la información solicitada y en almacenar los cambios realizados.

- **Ventajas:**

1. Facilidad de manejo de grandes volúmenes de información.
2. Gran velocidad en muy poco tiempo.
3. Independencia del tratamiento de información.
4. Seguridad de la información (acceso a usuarios autorizados), protección de información, de modificaciones, inclusiones, consulta.
5. No hay duplicidad de información, comprobación de información en el momento de introducir la misma.
6. Integridad referencial al terminar los registros.

- **Inconvenientes:**

1. El costo de actualización del hardware y software son muy elevados.
2. Costo (salario) del administrador de la base de datos es costoso.
3. El mal diseño de esta puede originar problemas a futuro.
4. Un mal adiestramiento a los usuarios puede originar problemas a futuro.
5. Si no se encuentra un manual del sistema no se podrán hacer relaciones con facilidad.
6. Generan campos vacíos en exceso.
7. El mal diseño de seguridad genera problemas en esta.

### **3 MARCO EMPIRICO**

#### **3.1METODOLOGIA DESCRIPTIVA**

El fenómeno de estudio se encuentra en la Universidad Pedagógica de El Salvador la cual surge como un centro de enseñanza superior con una mística y sello de trabajo que se caracteriza por la formación técnica, científica y humanística de recursos humanos en las diversas áreas del conocimiento del ámbito profesional; funciona bajo sistemas pedagógicos modernos cuya visión general de la educación es la de un proceso esencial para la formación integral de los diferentes profesionales. De ahí el nombre que la Universidad actualmente ostenta: **UNIVERSIDAD PEDAGÓGICA DE EL SALVADOR.**

Fundada en la ciudad de San Salvador, se firma su Acta de Constitución el 16 de marzo de 1982 y es el 26 de julio de 1982 que son aprobados sus Estatutos, según acuerdo No. 2239 del Ramo de Educación, día en que la universidad celebra su aniversario.

Dedicada a la mejora continua y en su afán de formar recursos humanos de éxito en la sociedad crea el proyecto de Pasantías, para facilitar las prácticas académicas e inserción laboral de sus alumnos

#### **3.1.1 METODOLOGIA DE LA INVESTIGACION**

##### **3.1.1.1 TIPO DE ESTUDIO**

###### **3.1.1.1.1 Documental o Bibliográfica:**

La metodología de recolección de datos fue empleada por libros de apoyo, sitios web y del apoyo de la universidad. Además nos apoyamos en sitios web para la obtención de conocimientos en el desarrollo del sistema.

###### **3.1.1.1.2 De Campo**

La investigación se realizo en las instalaciones de la Universidad Pedagógica de El Salvador, la cual nos brindo la información necesaria para el desarrollo de este sistema.

### **3.1.2 PROCEDIMIENTOS PARA LA OBTENCION DE DATOS**

Para obtener la información de cómo se realizaba actualmente el Control del proyecto de pasantías hablamos con los encargados de dicho proyecto, Lic. Antonio Deleón y Lic. Roxana de Abrego, entre otras cosas nos comentaron que aunque se lleva un control de los proyectos de pasantías no existe un programa para llevarlo adecuadamente, lo cual hace que la labor de asignación de pasantías sea más trabajosa.

Se necesita que el Sistema pueda hacer consultas para delimitar los perfiles de acuerdo a las exigencias y requisitos de las empresas solicitantes. Aunque actualmente el programa es un éxito completo, se puede actualizar haciendo uso de las herramientas informáticas.

Se observó cómo se lleva el control de la información, este es almacenado en carpetas las cuales contienen archivos en Excel, existe un archivo por hoja de vida.

Ya existe un formato de hoja de vida que se le envía a los alumnos para que lo completen y lo reenvíen por correo electrónico.

La institución cuenta con los recursos necesarios para llevar a cabo dicho proyecto, ya que tiene un departamento de informática completo y moderno.

#### **3.1.2.1 Instrumento**

La entrevista no fue diseñada previamente mas bien se escucho la explicación de cómo funcionaba actualmente el sistema sus pro y contra y como había estado funcionando hasta la fecha.

Las preguntas fueron realizadas principalmente al Lic. Antonio Gómez Deleon

¿Cómo surge la idea de la realización de pasantías en la Universidad?

Surge con el deseo de formar profesionales capacitados para el ámbito laboral, y la realización de una pasantía en su campo de estudios es una oportunidad para su desarrollo como profesional.

Lo de las pasantías no es algo nuevo esto es algo que siempre ha existido, pero al pasar el tiempo se le ha ido dando más importancia.

¿Cómo informan al estudiante este proyecto?

En los pasillos hay pizarras informativas que invitan al estudiante a integrarse a este proyecto, además los docentes también están encargados de informar a sus alumnos cuando hay una pasantía disponible, se le envían al alumno a su correo institucional un formato de hoja de vida para que la complete y la envíe de nuevo al remitente.

¿Cómo organizan las hojas de vida que los alumnos envían?

Se organizan por carpetas, cada hoja de vida es un documento, y este se archiva en la carpeta que corresponde a su carrera, cuando hay una oportunidad de pasantía se revisan uno por uno para elegir los que más se apegan con el perfil solicitado, luego se les contacta a los alumnos para saber si les interesa y los que quieren y pueden ir se envían a la empresa solicitante. Luego de terminado el proceso se lleva un registro de las pasantías realizadas y de los alumnos que han participado.

¿Considera que la utilización de un sistema para el control de pasantías puede ser útil?

Si, por que la información estaría ordenada y segura, además tendría más acceso a los alumnos y les permitiría actualizar sus datos y nos permitiría como encargados llevar un mejor control y selección de alumnos pasantes.

## **4.0 MARCO OPERATIVO**

### **4.1 Análisis del Diseño**

#### **4.1.1 Decisión de Requerimientos**

Los programas utilizados para el desarrollo del Sistema de Control de Pasantías fueron los siguientes:

#### **Microsoft Expression Web 4**

Microsoft Expression Web 4 fue lanzado el 7 de Junio de 2010. Añade la opción de complementos HTML y el acceso a una función SuperPreview en línea, para probar las páginas en exploradores no instalados en el sistema del usuario (como exploradores para las plataformas Mac OS X o Linux) de forma online. Microsoft Expression Web 4 también incluye un analizador de SEO, que comprueba el Web creado para optimizar y mejorar lo máximo posible su posicionamiento en buscadores.

con nombre código Quartz es un editor de paginas Web HTML desarrollado por Microsoft como una aplicación para sitios Web, considerada una versión superior de FrontPage 2003 por su semejanza en aspectos a este último. Utiliza el mecanismo WYSIWYG (Lo que ve es lo que obtiene) para modificar páginas Web, está enfocada al cumplimiento de estándares, ya que permite validar nuestro contenido contra el estándar que el usuario desee seguir o navegadores en los que se planea montar la aplicación Web. Además, es compatible con hojas de estilo CSS y con Microsoft Visual Studio. Asimismo, incluye la capacidad de procesar archivos XML mediante JavaScript.

Desde la versión 2 del programa, se permite la integración con lenguajes de servidor como ASP.NET o PHP sin necesidad de instalar un servidor. También permite la interacción con Adobe Photoshop para generar imágenes.

## **Servidor HTTP Apache**

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix(BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de Internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado").

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios Web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. (Estadísticas históricas y de uso diario proporcionadas por Netcraft3 ).

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

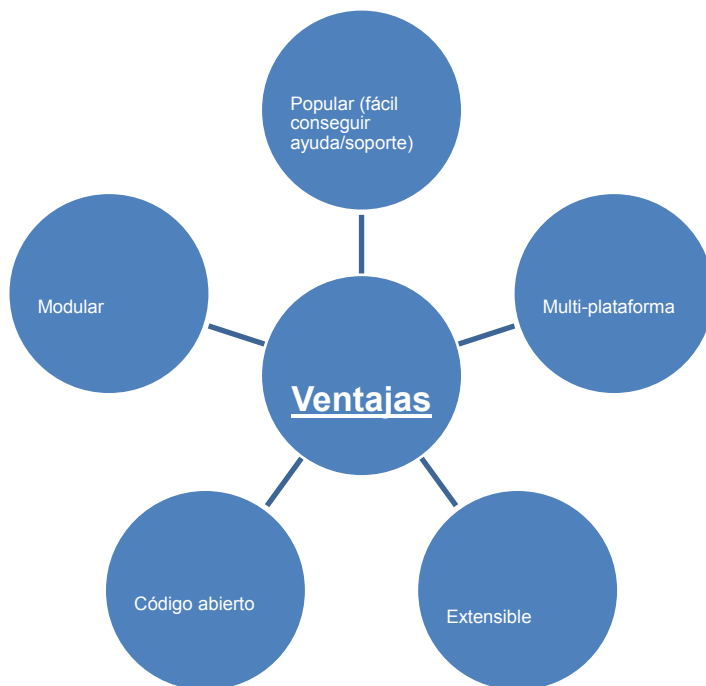
Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

Apache es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto a MySQL y los lenguajes de programación PHP/Perl/Python (y ahora también Ruby).

Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM WebSphere application server. Mac OS X integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones WebObjects. Es soportado de alguna manera por Borland en las herramientas de desarrollo Kylix y Delphi. Apache es incluido con Novell NetWare 6.5, donde es el servidor web por defecto, y en muchas distribuciones Linux.

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet. Un usuario que tiene Apache instalado en su escritorio puede colocar arbitrariamente archivos en la raíz de documentos de Apache, desde donde pueden ser compartidos.

Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de previsualizar y probar código mientras éste es desarrollado.



## PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Se usa principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

### Ventajas

Es un lenguaje multiplataforma.

Orientado al desarrollo de aplicaciones Web dinámicas con acceso a información almacenada en una base de datos.

El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.

Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.

Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).

Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.

Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

Permite aplicar técnicas de programación orientada a objetos.

Biblioteca nativa de funciones sumamente amplia e incluida.

No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.

Tiene manejo de excepciones (desde PHP5).

Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar (muchos otros lenguajes tampoco lo hacen), aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.

## MySQL

Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

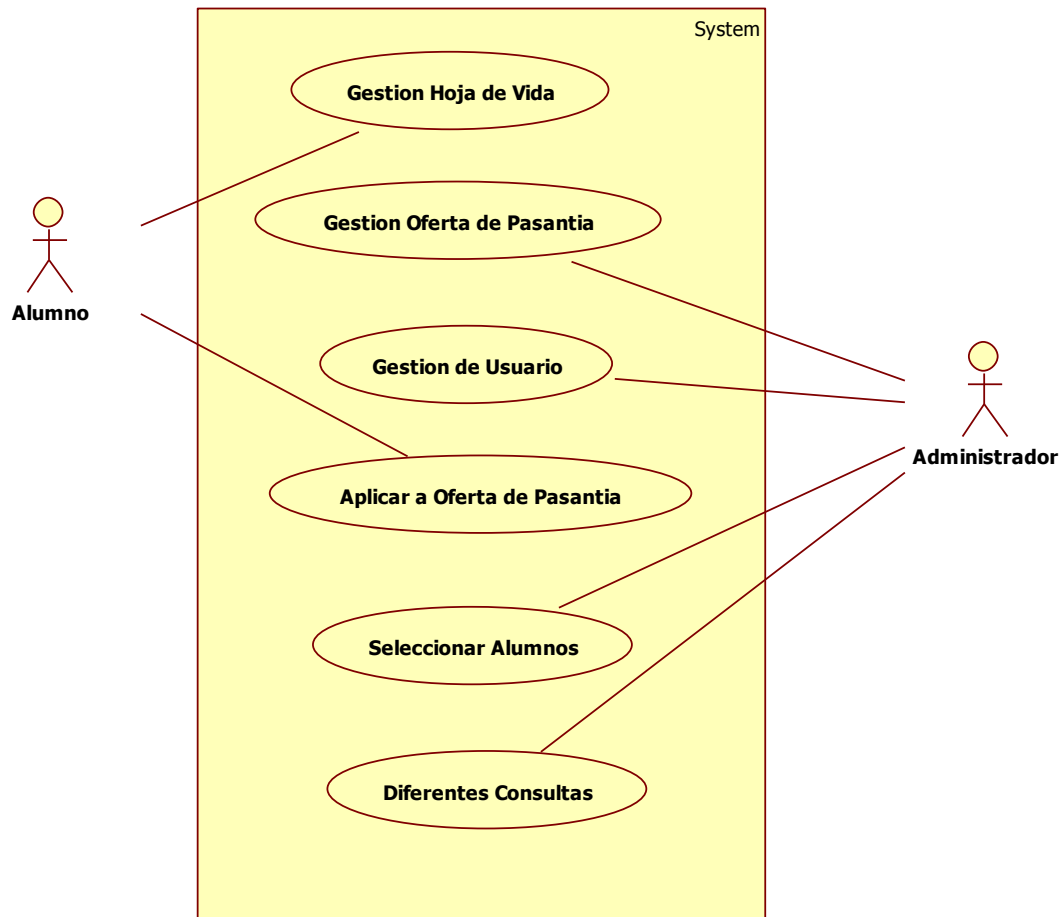
Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

## 4.1.2 Diagramas de casos de uso

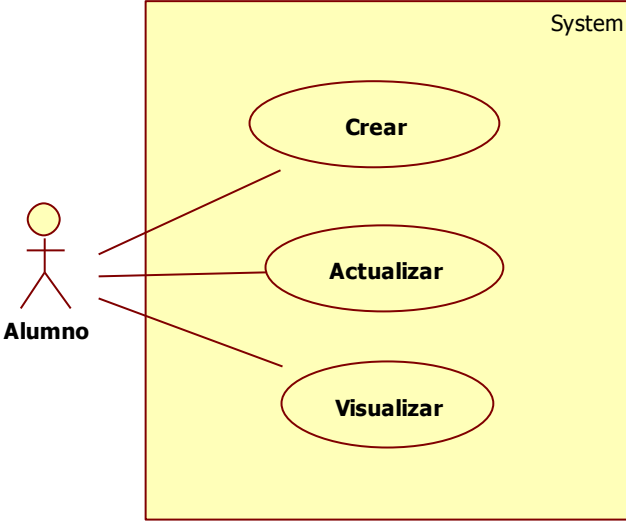
CASOS DE USO

SISTEMA DE CONTROL DE PASANTIAS

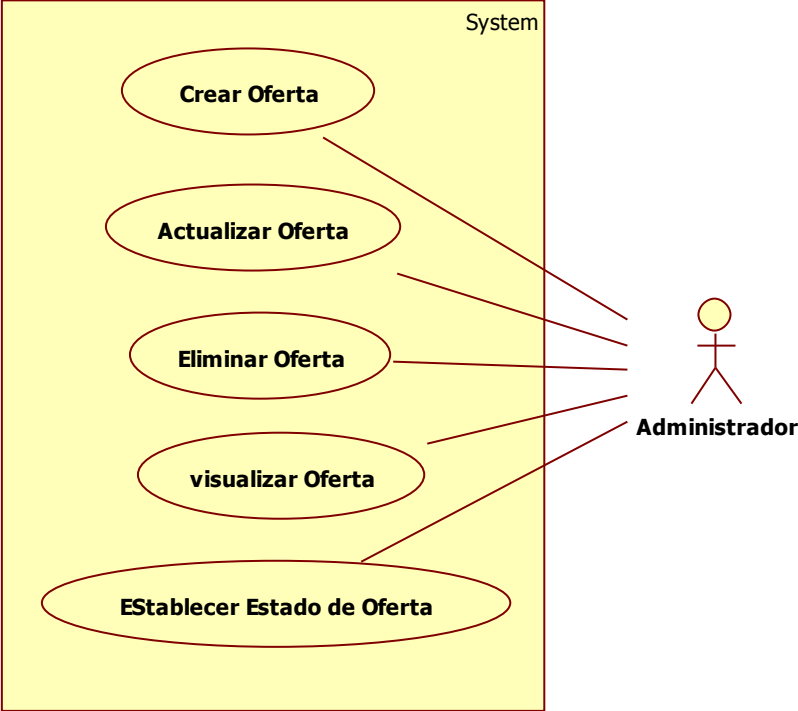
PRINCIPAL



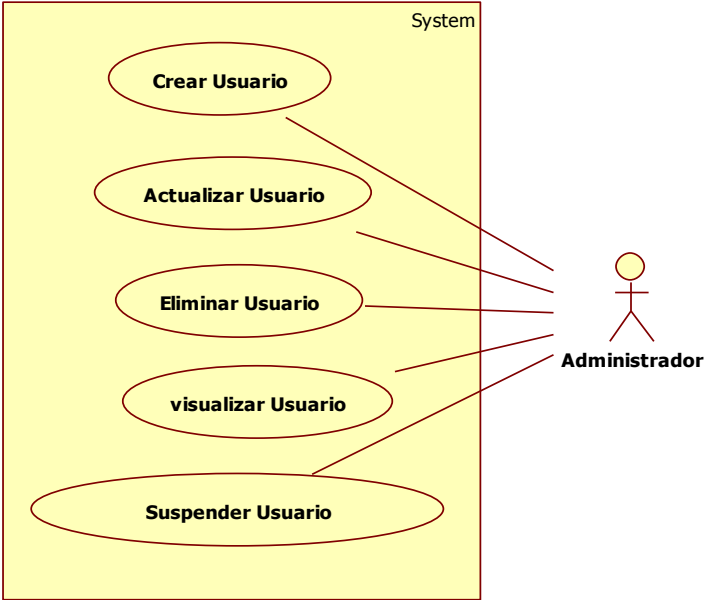
GESTION HOJA DE VIDA



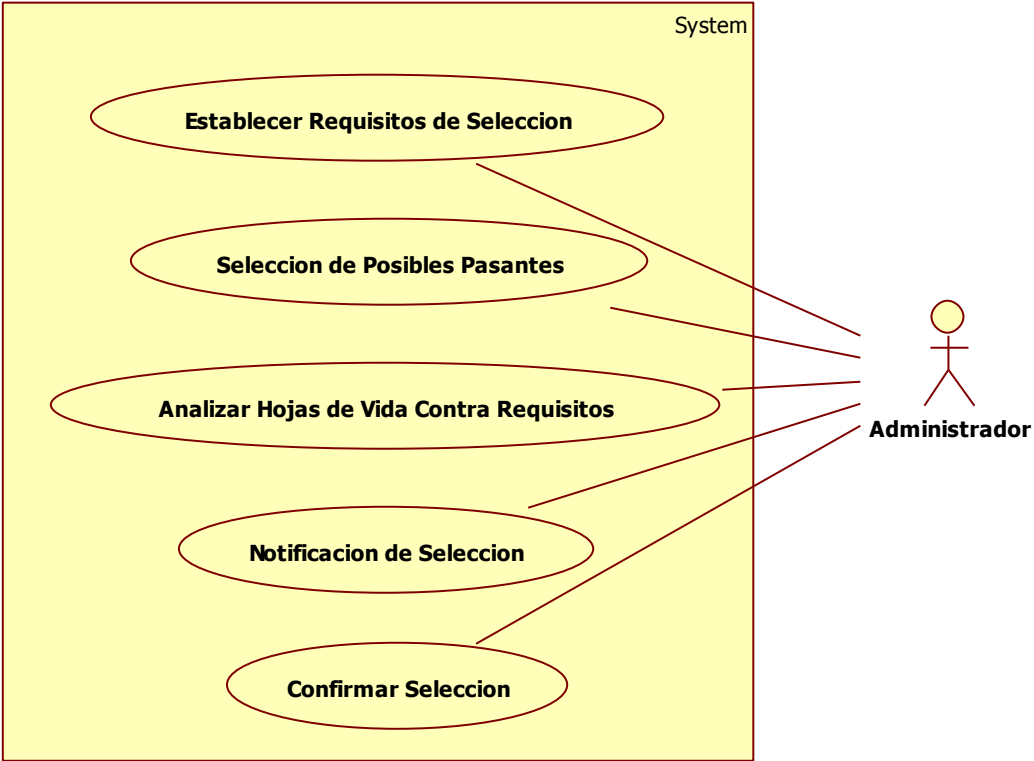
GESTION OFERTA DE PASANTIA



GESTION DE USUARIO



SELECCIONAR ALUMNOS



### 4.1.3 ESPECIFICACION DE CASOS DE USO

Nombre:	<b>Crear hoja de vida</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Alumno
Descripción:	Permite crear hoja de vida
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como usuario
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Ingresar Nombre</li> <li>2. Ingresar Apellido</li> <li>3. Seleccionar Carrera</li> <li>4. Seleccionar Fecha de Nacimiento</li> <li>5. Ingresar Numero de DUI</li> <li>6. Ingresar Numero de NIT</li> <li>7. Seleccionar el ciclo Académico que cursa</li> <li>8. Seleccionar la Cantidad de Materias Cursadas</li> <li>9. Ingresar Dirección</li> <li>10. Ingresar Correo electrónico</li> <li>11. Ingresar Números de Teléfono</li> <li>12. Marcar el nivel de conocimiento y uso de software</li> <li>13. Especificar Conocimientos de otros Software</li> <li>14. Seleccionar Nivel de Conocimiento de Ingles</li> <li>15. Especificar que otro idioma conoce</li> <li>16. Ingresar que opción de Bachiller obtuvo</li> <li>17. Especificar en qué institución realizo bachillerato</li> <li>18. Especificar en qué año se graduó de bachiller</li> <li>19. Especificar otros estudios</li> <li>20. Especificar experiencia laboral</li> <li>21. Ingresar referencias</li> </ol>
Flujo Alternativo	<ol style="list-style-type: none"> <li>15.- Especificar si conoce otro Idioma</li> <li>1. Si no conoce otro idioma puede dejar este campo en blanco</li> </ol>
Post-Condiciones:	

Nombre:	<b>Actualizar hoja de vida</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Alumno
Descripción:	Permite Actualizar la respectiva hoja de vida
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como usuario Haber ingresado ya una hoja re vida
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Abrir la hoja de vida que ya creo anteriormente</li> <li>2. Seleccionar campos que modificara</li> <li>3. Modificar campos</li> <li>4. Validar la modificación</li> <li>5. Confirmar el cambio</li> </ol>
Flujo Alternativo	<p>4. Validar la modificación</p> <ol style="list-style-type: none"> <li>1. Si los espacios modificados no fueron llenados correctamente se mostrara un mensaje de notificación sobre los campos que no pasaron la validación</li> </ol>
Post-Condiciones:	

Nombre:	<b>Visualizar hoja de vida</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Alumno
Descripción:	Permite visualizar la respectiva hoja de vida
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como usuario Haber ingresado ya su hoja de vida
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar si desea que su hoja de vida sea visible</li> </ol>
Flujo Alternativo	

Post-Condiciones:	
-------------------	--

Nombre:	<b>Crear Oferta de Pasantía</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite crear una oferta de pasantía
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como Administrador
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Ingresar nombre de la empresa solicitante</li> <li>2. Ingresar cantidad de alumnos solicitados</li> <li>3. Detallar el departamento que solicita a los pasantes</li> <li>4. Especificar la actividad a realizar</li> <li>5. Especificar las carreras que pueden aplicar</li> <li>6. Ingresar requisitos solicitados por la empresa</li> </ol>
Flujo Alternativo	
Post-Condiciones:	

Nombre:	<b>Actualizar Oferta</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite actualizar una oferta de pasantía
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una oferta de pasantía
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar oferta</li> <li>2. Modificar campos deseados</li> <li>3. Validar la modificación</li> <li>4. Confirmar el cambio</li> </ol>
Flujo Alternativo	<ol style="list-style-type: none"> <li>4. Validar la modificación <ol style="list-style-type: none"> <li>1. Si los espacios modificados no fueron llenados correctamente se mostrara un mensaje de notificación sobre los campos que no pasaron la validación</li> </ol> </li> </ol>

Post-Condiciones:	

Nombre:	<b>Eliminar Oferta</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite eliminar una oferta de pasantía
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una oferta de pasantía
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar oferta de pasantía</li> <li>2. Seleccionar eliminar pasantía</li> <li>3. Confirmar eliminación de oferta</li> </ol>
Flujo Alternativo	
Post-Condiciones:	La oferta a sido eliminada

Nombre:	<b>Visualizar Oferta</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite que una oferta de pasantía sea visible a los alumnos
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una oferta de pasantía
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar oferta de pasantía</li> <li>2. Seleccionar que se desea hacer visible la oferta</li> </ol>
Flujo Alternativo	<p>2.- Seleccionar que se desea hacer visible la oferta</p> <p>Si la oferta no se desea hacer visible solo quedara creada pero los alumnos no podrán verla</p>
Post-Condiciones:	La oferta es visible ahora

Nombre:	<b>Establecer Estado de Oferta</b>
Versión:	1.0

Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite mostrar en qué estado se encuentra un oferta de pasantía
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una oferta de pasantía
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar oferta de pasantía</li> <li>2. Establecer estado de oferta</li> </ol>
Flujo Alternativo	<p>2.- establecer estado de oferta</p> <p>Si la oferta esta habilitada el alumno podrá aplicar a la oferta</p> <p>Si la oferta está cerrada se bloqueara automáticamente la opción de aplicación</p>
Post-Condicion:	
Nombre:	<b>Crear Usuario</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite crear una cuenta de usuario
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como Administrador
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Ingresar datos generales de usuario</li> <li>2. Ingresar datos académicos básicos del alumno</li> <li>3. Asignarle una contraseña</li> </ol>
Flujo Alternativo	
Post-Condicion:	Usuario creado

Nombre:	<b>Actualizar Usuario</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite actualizar una cuenta de usuario
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador

	Haber creado ya un usuario
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar Usuario</li> <li>2. Modificar campos deseados</li> <li>3. Validar la modificación</li> <li>4. Confirmar el cambio</li> </ol>
Flujo Alternativo	<ol style="list-style-type: none"> <li>4. Validar la modificación</li> <li>2. Si los espacios modificados no fueron llenados correctamente se mostrara un mensaje de notificación sobre los campos que no pasaron la validación</li> </ol>
Post-Condiciones:	

Nombre:	<b>Eliminar Usuario</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite eliminar una cuenta de usuario
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una cuenta de usuario
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar cuenta de usuario</li> <li>2. Seleccionar eliminar cuenta de usuario</li> <li>3. Confirmar eliminación de cuenta de usuario</li> </ol>
Flujo Alternativo	
Post-Condiciones:	La cuenta ha sido eliminada

Nombre:	<b>Visualizar Usuario</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite que una cuenta de usuario pueda ser visible para aplicar
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una cuenta de usuario

Flujo Normal:	<ol style="list-style-type: none"> <li>3. Seleccionar una cuenta de usuario</li> <li>4. Seleccionar que se desea hacer visible la cuenta de usuario</li> </ol>
Flujo Alternativo	<ol style="list-style-type: none"> <li>2.- Seleccionar que se desea hacer visible la cuenta de usuario</li> </ol> <p>Si la cuenta no se desea hacer visible solo quedara creada pero no podrá el alumno aplicar a alguna opción ya que su cuenta no será visible.</p>
Post-Condiciones:	La cuenta de usuario es visible ahora

Nombre:	<b>Suspender Usuario</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite suspender una cuenta de usuario, básicamente eliminar su cuenta
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber creado ya una cuenta de usuario
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Seleccionar la cuenta de usuario</li> <li>2. Seleccionar la opción suspender usuario</li> <li>3. Dejar un registro del motivo por el que se suspendió la cuenta</li> <li>4. Confirmar la suspensión de usuario</li> </ol>
Flujo Alternativo	
Post-Condiciones:	

Nombre:	<b>Establecer requisitos de Selección</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite crear, Actualizar y visualizar los requisitos de selección
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador

	Existir ya una oferta de pasantía
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Elegir una oferta de pasantía</li> <li>2. Crear requisitos de selección</li> </ol>
Flujo Alternativo	
Post-Condición:	Los requisitos quedan establecidos

Nombre:	<b>Selección de Posibles Pasantes</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite la elección de los posibles estudiantes que podrían aplicar a determinada propuesta.
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Existir ya cuentas de usuarios
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Por medio de consultas elegir que estudiantes podrían ser los aspirantes</li> </ol>
Flujo Alternativo	<ol style="list-style-type: none"> <li>1. Por medio de consultas elegir que estudiantes podrían ser los aspirantes</li> </ol> <p>Estarían incluidos tanto los que han elegido aplicar a la oferta de pasantía como los que no lo han hecho aun</p>
Post-Condición:	Queda creada una lista de posibles pasantes

Nombre:	<b>Analizar hojas de vida contra requisitos</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Permite hacer una comparación entre los requisitos solicitados y los datos ingresados en las hojas de vida de los usuarios
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Existir oferta de pasantías Existir cuentas de usuarios con sus respectivas hojas de vida activas y que estén dentro de los posibles pasantes
Flujo Normal:	

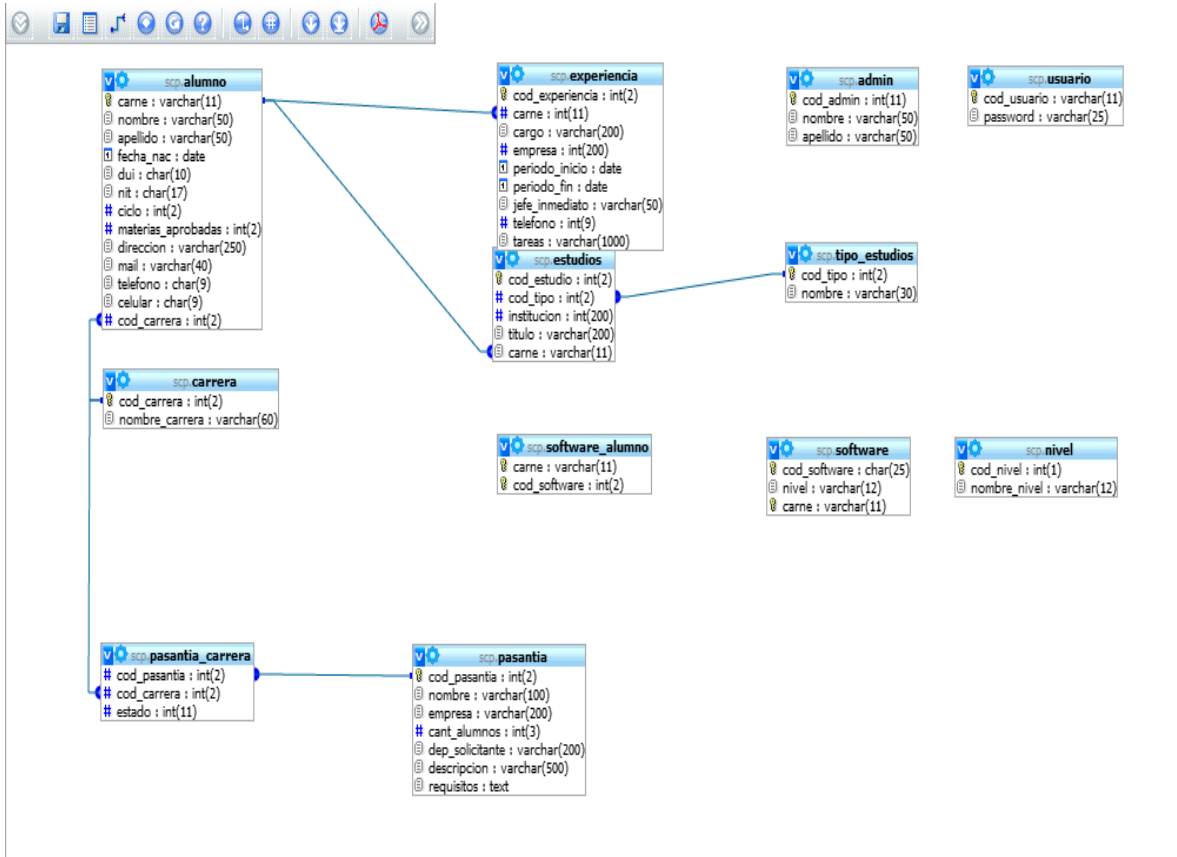
	<ol style="list-style-type: none"> <li>1. elegir un usuario</li> <li>2. elegir una oferta de pasantía con su respectivos requisitos</li> <li>3. Comparar hoja de vida contra requisitos</li> <li>4. Marcar un resultado de la comparación</li> <li>5. Ordenar por grado de aplicación</li> <li>6. Crear una lista de usuarios que aplican</li> <li>7.</li> </ol>
Flujo Alterno	
Post-Condiciones:	

Nombre:	<b>Notificación de Selección</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	Informa a los estudiantes que han sido seleccionados para aplicar a la propuesta
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Debe existir ya la lista de usuarios que aplican
Flujo Normal:	<ol style="list-style-type: none"> <li>1. Enviar un mail o notificación por medio de su cuenta de usuario que puede aplicar a la propuesta.</li> <li>2. Esperar la respuesta de usuario.</li> </ol>
Flujo Alterno	
Post-Condiciones:	Usuarios quedan notificados

Nombre:	<b>Confirmar Selección</b>
Versión:	1.0
Autor:	Karen Julissa Alas
Actores:	Administrador
Descripción:	El administrador confirma a la empresa que estudiantes enviara para la realización de las pasantías
Fecha:	01/05/11
Precondiciones:	Haberse autenticado en el sistema como administrador Haber recibido las confirmaciones de los estudiantes que desean

	aplicar
Flujo Normal:	<ol style="list-style-type: none"><li>1. Notificar a la empresa que estudiantes han sido elegidos para la realización de pasantías.</li><li>2. Poner en contacto Pasante-Empresa</li></ol>
Flujo Alternativo	
Post-Condiciones:	La oferta de pasantía ha sido realizada

## 4.1.4 Estructura de base de datos



## Bibliografía

- Material didáctico y de apoyo recibido durante el desarrollo de clases
- Basesdedatos.org
- Clasificación arbitraria de los sistemas, páginas 706 – 708, Chiavenato, 1999, clasificación de los sistemas de acuerdo con los criterios de complejidad y previsión de su funcionamiento.
- Una tipología de sistemas, páginas 130 - 143, Checkland, 1993, clasificación de los sistemas en físicos, naturales, de actividad humana, diseñados y sociales.
- Sistema de Información, Manuel Peralta, <http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>
- Chiavenato, Idalberto. Introducción a la Teoría General de la Administración. 3ra. Edición. Edit. McGraw-Hill. 1992.
- Von Bertalanffy, Ludwig. Teoría General de Sistemas. Petrópolis, Vozes. 1976
- Sitio oficial Universidad Pedagógica de El Salvador [www.pedagogica.edu.sv](http://www.pedagogica.edu.sv)
- Pagina de principal de MYSQL <http://dev.mysql.com>
- MIA. Marcela García Alonso <http://www.utim.edu.mx/~mgarcia/DOCUMENTO/ADSI2/ADSI2-1.pdf>

# ANEXOS

## MANUAL “SISTEMA DE CONTROL DE PASANTIAS”



### LOGIN

La parte del 'login' permite ingresar al sistema a los usuarios que ya se han registrado previamente, basta con ingresar su nombre de usuario y su clave.



### REGISTRARSE EN EL SISTEMA

Para registrarse en el sistema el usuario debe ingresar los siguientes datos generales:

- Carne

- Nombre del Alumno
- Apellido
- Carrera a la que pertenece (debe de elegir una de la lista)
- Nombre de usuario
- Clave y confirmar su clave.

El usuario debe introducir nombre y apellidos validos y un número de carnet vigente. En caso de no llenar todos campos requeridos no podrá registrarse.



Una vez el usuario haya creado su cuenta e ingrese al sistema, podrá aplicar a las opciones de pasantías, las cuales estarán organizadas por categorías.

En el menú izquierdo tendrá la opción de ver su soja de vida, editarla, y salir del sistema.

# Sistema de control de pasantías

Facultad de Ciencias Económicas



Alumnos

- Regresar
- Ver Hoja de Vida
- Salir

Editar Hoja de Vida

**Datos Personales** | Conocimientos | Estudios | Experiencia

Apellidos:

Nombre:

Carrera:

Fecha de Nacimiento:

Numero de DUI:

Numero de NIT:

Ciclo Academico que Cursa:

Cantidad de Materias Cursadas:

Direccion:

Correo Electronico:

Telefono:

En el menú Editar hoja de Vida, Automáticamente aparecerán los datos personales previamente ingresados para el registro, el usuario deberá completarlos ingresando datos validos.

Al terminar de completar los campos y aplicar los datos automáticamente quedaran grabados en la base de datos.

# Sistema de control de pasantías

Facultad de Ciencias Económicas



Alumnos

- Regresar
- Ver Hoja de Vida
- Salir

Editar Hoja de Vida

**Datos Personales** | **Conocimientos** | Estudios | Experiencia

Software:  Nivel:

Software	Nivel	
Word	Experto	
Excel	Intermedio	
Intenet	Avanzado	
Power Point	Intermedio	

En la pestaña Conocimientos podrá agregar el software que conoce y el nivel de conocimiento. Puede eliminar cuando desee los conocimientos agregados y agregar nuevos.

Alumnos

Regresar

Ver Hoja de Vida

Salir

Editar Hoja de Vida

Datos Personales
Conocimientos
Estudios
Experiencia

Tipo

Otro Tipo

Institucion

Titulo

	Institucion	Titulo	Acciones
und	piritu Santo	Bachiller	<input type="button" value="Eliminar"/>
und	d Pedagogica de El	Lic. en Contaduria Publica	<input type="button" value="Eliminar"/>

En la pestaña estudios podrá ingresar todos los estudios realizados, sean estos: educación básica, bachillerato, cursos libres, etc.

Y al igual que la pestaña de conocimientos podrán eliminar y agregar.

Alumnos

Regresar

Ver Hoja de Vida

Salir

Editar Hoja de Vida

Datos Personales
Conocimientos
Estudios
Experiencia

Cargo

Empresa

Jefe Inm.

desde

hasta

Telefono

Cargo	Empresa	Jefe Inmediato	Telefono	Acciones

También agregara la experiencia laboral obtenida, el tiempo que laboro para determinada empresa, jefe inmediato, numero de contacto, y acciones o tareas que realizaba en dicho trabajo.