



***Benemérita Universidad Autónoma de Puebla***

***Facultad de Ciencias de la Computación***

**MODEL:**

***Módulo de Ofertas y  
Demandas de Empleo en Línea***

Tesis Profesional

Que para obtener el Título de Licenciado en Ciencias de la  
Computación

**Presenta:**

**ELUD DE JESUS CÁRCAMO**

**Asesor:**

Dra. Concepción Pérez de Celis Herrero

**Puebla, Pue. Septiembre de 2008**

## Índice

<b>INTRODUCCIÓN.....</b>	<b>4</b>
<b>CAPITULO 1. METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....</b>	<b>6</b>
1.1 CICLO DE VIDA DE LOS PROYECTOS DE DESARROLLO DE SOFTWARE .....	6
1.2 FASES O ETAPAS DEL PROCESO DE DESARROLLO UNIFICADO DE SOFTWARE .....	7
1.3 MODELOS ITERATIVO E INCREMENTAL .....	8
1.4 BENEFICIOS DEL ENFOQUE ITERATIVO.....	9
1.5 DIAGRAMA DE CASOS DE USO .....	10
1.6 CASOS DE USO.....	11
1.7 MODELO DE ANÁLISIS .....	13
1.8 MODELO DE DISEÑO.....	14
1.9 EL LENGUAJE UNIFICADO DE MODELADO .....	15
1.10 DIAGRAMA DE SECUENCIAS .....	16
1.11 BASES DE DATOS .....	17
1.12 COMPONENTES DE UN SISTEMA DE BASE DE DATOS .....	19
1.13 SISTEMA GESTOR DE BASES DE DATOS.....	19
1.13.1 Modelos de Datos.....	19
1.14 DISEÑO CONCEPTUAL DE BASES DE DATOS .....	20
1.14.1 El modelo entidad – relación (M E-R).....	20
1.15 COMPONENTES DEL MODELO .....	20
1.16 CLASIFICACIÓN DE LOS ATRIBUTOS. ....	21
1.17 TIPOS DE CLAVE.....	22
1.8 NORMALIZACIÓN.....	23
<b>CAPITULO 2. ANÁLISIS DEL SISTEMA .....</b>	<b>24</b>
2.1 PLANTEAMIENTO DEL PROBLEMA: MODELO DEL NEGOCIO .....	24
2.2 OBJETIVOS .....	25
2.3 DEFINICIÓN DE REQUERIMIENTOS .....	26
2.4 ANÁLISIS DE FACTIBILIDAD .....	28
2.5 CASOS DE USO.....	28
2.6 DIAGRAMA DE SECUENCIA.....	33
<b>CAPITULO 3. ANÁLISIS Y DISEÑO DE LA BASE DE DATOS ASOCIADA AL SISTEMA.....</b>	<b>37</b>
3.1 DISEÑO CONCEPTUAL .....	37
3.2 DIAGRAMA E-R.....	38
3.3 DIAGRAMA ENTIDAD – RELACIÓN EN UML .....	39
3.4 DISEÑO LÓGICO .....	41
3.5 NORMALIZACIÓN: .....	43
3.6 DICCIONARIO DE DATOS DEL SISTEMA.....	44
<b>CAPITULO 4. IMPLEMENTACIÓN DEL SISTEMA (MODEL).....</b>	<b>50</b>
4.1 LENGUAJE DE PROGRAMACIÓN.....	50
4.2 MANEJADOR DE BASE DE DATOS .....	50
4.3 HERRAMIENTAS UTILIZADAS .....	50

4.4 CÓDIGO .....	51
4.5 PANTALLAS DEL SISTEMA.....	52
<b>CONCLUSIONES .....</b>	<b>55</b>
<b>TRABAJO A FUTURO .....</b>	<b>55</b>
<b>BIBLIOGRAFÍA.....</b>	<b>56</b>
<b>PAGINAS VISITADAS .....</b>	<b>56</b>

### Índice de figuras

Figura 1.1 El proceso fases, iteraciones y actividades. ....	9
Figura 1.2 Diagrama de Casos de Uso del sistema MODEL (Módulo de Ofertas y Demanda de Empleo en Línea). ....	12
Figura 1.3 Caso de Uso Alta Vacante en el Modelo de Análisis. ....	13
Figura 1.4 Diagrama de colaboración para el caso de uso Alta Vacante.....	14
Figura 1.5 Las clases de diseño refinan las clases de análisis. ....	14
Figura 1.6 Diagrama de Secuencias, del caso de uso “Alta Vacante”. ....	17
Figura 1.7 Ejemplo de una entidad (Vacante). ....	20
Figura 1.8 Ejemplo de un Atributo de la Entidad Vacante. ....	21
Figura 1.9 Atributo Derivado.....	22
Figura 2.1 Caso de uso Administrar Vacantes.....	30
Figura 2.2 Caso de uso administrar bolsa de trabajo.....	30
Figura 2.3 Caso de uso Buscar Empleo. ....	31
Figura 2.4 Caso de uso Registro.....	31
Figura 2.5 Caso de uso Modificar Perfil.....	32
Figura 2.6 Caso de uso Verificación. ....	33
Figura 2.7 Diagrama de secuencia “Registro de un Desempleado”. ....	34
Figura 2.8 Diagrama de Secuencia Buscar Personal. ....	35
Figura 2.9 Diagrama de secuencia de búsqueda de empleo.....	36
Figura 3.1 Diagrama E – R .....	38
Figura 3.2 Diagrama Entidad – Relación en UML. ....	40
Figura 4.1 Pantalla inicial del sistema Model.....	52
Figura 4.2 Pantalla de login para empleadores.....	53
Figura 4.3 Alta de una vacante.....	53
Figura 4.4 Modificación de Perfil de un Empleador.....	54
Figura 4.5 Buscar Personal .....	54

## Introducción

El índice de desempleo en México del año 2007 fue del 3.9 % y en Febrero del 2008 fue reducido al 3.81 % de la Población Económicamente Activa según datos de la Organización para la Cooperación y el Desarrollo Económico (OCDE). El número de mexicanos desempleados en el año 2008 permanecerá en un millón 800 mil personas, indicó la OCDE. Este es un gran número de personas que aún no tienen empleo, por lo que se deben de promover cada día mas ofertas de trabajo para que México salga del lugar 32 de 111 países con mejor calidad de vida.

Estos datos son un motivo para colaborar con los diferentes medios para reducir el índice de desempleo en nuestro país y ayudará a las miles de personas que aún no encuentran un empleo, por diferentes razones. Este sistema manejará a ambas partes, me refiero a las empresas y a los desempleados, ya que contendrá los dos módulos integrados.

Por lo cual se propone un sistema que ayude a las personas a buscar trabajo desde cualquier lugar donde se cuente con una computadora y acceso a Internet, permitiendo consultar la información publicada por las empresas, como lo son las vacantes de empleo con su respectiva información de: puesto, sueldo, horario, lugar de trabajo, requisitos mínimos, fecha de presentación, etc., el sistema será desarrollado usando la tecnología Web y bases de datos para manejar toda la información necesaria sobre las empresas y personas desempleadas, haciendo de este sistema un proyecto interesante por utilizar tecnologías Web aplicadas a un problema social en particular.

Esto dará como resultado que toda persona interesada en unirse al sector laboral pueda consultar las diferentes vacantes de unas o varias empresas, teniendo así la oportunidad de conseguir un trabajo. Cabe mencionar que este sistema contará con opciones tanto para las personas desempleadas como para las empresas, ya que el objetivo general es lograr la comunicación entre las empresas y las personas que aún no tienen un trabajo.

El principal objetivo de este sistema es poder realizar el contacto entre una persona desempleada y un empleador por medio de los datos proporcionados en el momento de hacer su registro en el sistema. Para la realización de este proyecto se utilizará la metodología : proceso unificado.

A continuación hacemos una breve descripción de los cuatro capítulos de los que consta esta tesis:

En el primer capítulo se discute la importancia de la ingeniería de software y del uso de metodologías para la creación de cualquier sistema, así como la secuencia que se debe de seguir para elaborar software de calidad, mencionando el diagrama de Casos de Uso del sistema MODEL, el diagrama entidad - relación para identificar correctamente las tablas que participan en el sistema y las formas normales para llegar a la normalización de una base de datos.

El segundo capítulo tratará sobre el análisis del sistema Módulo de Ofertas y Demandas de Empleo en Línea (MODEL), se hace el planteamiento del problema, se definen los objetivos del sistema y la definición de los requerimientos, posteriormente se listan los casos de uso que existen en el sistema.

El tercer capítulo es el diseño del sistema y en el se muestra el diagrama entidad – relación de la base de datos y se toman las tablas que forman el sistema para aplicarles la normalización quedando estas en 4FN, cumpliendo las reglas de la normalización, al final de este capítulo se muestra el diccionario de datos del sistema.

El cuarto capítulo es la Implementación del sistema, donde se presentan tecnologías que se utilizaron para desarrollar este proyecto, utilizando el manejador de base de datos MySQL para el manejo de la información y PHP para la creación de la interfaz entre el usuario y las bases de datos.

Por último se muestran las conclusiones de este trabajo de tesis y trabajo a futuro.

## CAPITULO 1. METODOLOGÍAS DE DESARROLLO DE SOFTWARE.

La Ingeniería de Software (SE del inglés Software Engineering) “es un enfoque sistemático del desarrollo, operación, mantenimiento y retiro del software”, que en palabras más simples, se considera que “la Ingeniería de Software es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software”, es decir, “permite elaborar consistentemente productos correctos, utilizables y costo-efectivos”.<sup>[1]</sup>

Por lo consiguiente para desarrollar software de calidad que pueda competir en el mercado es necesario utilizar las metodologías que proporciona la ingeniería de software. El Proceso Unificado de Desarrollo (Rational Unified Process RUP) que junto con el Lenguaje Unificado de Modelado (UML) en la actualidad constituyen la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos <sup>[2]</sup>, razón por la cual es la metodología que se utilizará en el presente trabajo.

Estas metodologías brindan excelentes técnicas de desarrollo que permiten hacer que un sistema funcione de la mejor forma y que cumpla con los estándares requeridos para su producción y utilización, de esta manera se logrará que este producto pueda ser correcto, confiable, funcional, seguro y portable.

### 1.1 Ciclo de vida de los proyectos de desarrollo de software.

Independientemente del área de aplicación, del tamaño y la complejidad del proyecto, el proceso de desarrollo de software contiene siempre una serie de fases genéricas, existentes en todos los paradigmas. <sup>[3]</sup>

**Análisis del sistema:** define el papel de cada elemento de un sistema informático, estableciendo cuál es el papel del software dentro de ese sistema.

**Análisis de requisitos del software:** el análisis del sistema proporciona el ámbito del software, su relación con el resto de los componentes del sistema, pero antes de empezar a desarrollar es necesario hacer una definición más detallada de la función del software.

**Planificación del proyecto software:** durante esta etapa se lleva a cabo el análisis de riesgos, se definen los recursos necesarios para desarrollar el software y se establecen las estimaciones de tiempo y costes.

## 1.2 Fases o Etapas del proceso de desarrollo unificado de software.

La ingeniería de software requiere llevar a acabo muchas tareas, pero las más sobresalientes son:

- Requerimientos del Sistema
- Análisis del Sistema
- Diseño del Sistema
- Codificación
- Pruebas

En cada una de estas etapas se realiza una tarea específica las cuales describiremos a continuación.

**Requerimientos del Sistema:** El objetivo de esta fase es ayudar al equipo de proyecto a decidir cuáles son los verdaderos objetivos del proyecto, y se deben especificar detalladamente cada una de las funciones que se requieren en el sistema, se hace una descripción del producto final y responde a las siguientes preguntas:

¿Cuáles son las principales funciones del sistema para los usuarios más importantes?

¿Cómo podría ser la mejor arquitectura del sistema?

¿Cuál es el plan del proyecto y cuánto costará desarrollar el producto?

En esta fase se identifican y priorizan los **riesgos más importantes**.

**Análisis del Sistema:** Define el papel de cada elemento de un sistema informático, estableciendo cuál es el papel del software dentro de ese sistema.

**Diseño del sistema:** Traduce los requisitos a un conjunto de representaciones (gráficas, en forma de tabla o basadas en algún lenguaje

apropiado) que describen como van a estructurarse los datos, cuál va a ser la arquitectura general del sistema, así cómo su comportamiento dinámico, y cómo van a ser las interfaces.

En esta fase, se decide que tecnología se utilizará en el sistema, aprovechando y adaptando sus ventajas.

**Codificación:** La implementación del diseño es una fase extensa y requiere de un equipo de desarrolladores. La coordinación de este equipo es importante si se desea finalizar el proyecto en el tiempo previsto.

El diseño se debe de traducir en una forma legible por la máquina, es en esta fase donde la codificación se lleva a cabo.

En esta fase, el diseño se traduce a un lenguaje de programación, dando como resultado un programa ejecutable. La buena calidad de los programas desarrollados depende en gran medida de la calidad del diseño. Una vez codificados los programas debe revisarse su estilo y claridad, y se comprueba que haya una correspondencia con la estructura de los mismos definida en la fase de diseño.

**Pruebas:** Una vez que tenemos implementado el software es preciso probarlo, para detectar errores de codificación, de diseño o de especificación. Las pruebas son necesarias para encontrar el mayor número posible de errores antes de entregar el programa al cliente.

### 1.3 Modelos Iterativo e incremental

Es un modelo en el cual se divide el esfuerzo de desarrollo de un proyecto de software en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos a crecimientos en el producto.

Las iteraciones deben estar controladas. Esto significa que deben seleccionarse y ejecutarse de una forma planificada. Los desarrolladores basan la selección de lo que implementarán en cada iteración en dos cosas: el conjunto de casos de uso que amplían la funcionalidad, y en los riesgos más importantes que deben disminuirse.

## 1.4 Beneficios del enfoque iterativo

- Algunos de los beneficios del enfoque iterativo se listan a continuación:
- ✓ La iteración controlada reduce el riesgo de pérdida en el valor del producto completo, es decir, solo se perderá el esfuerzo de una iteración.
  - ✓ El cliente se involucra, teniendo la oportunidad de probar y dar su opinión en el avance del proyecto.
  - ✓ El progreso se puede medir en periodos cortos de tiempo.
  - ✓ Las pruebas del software y la integración son constantes.
  - ✓ Además si existieran errores producidos en un incremento se pueden solucionar en el próximo incremento.

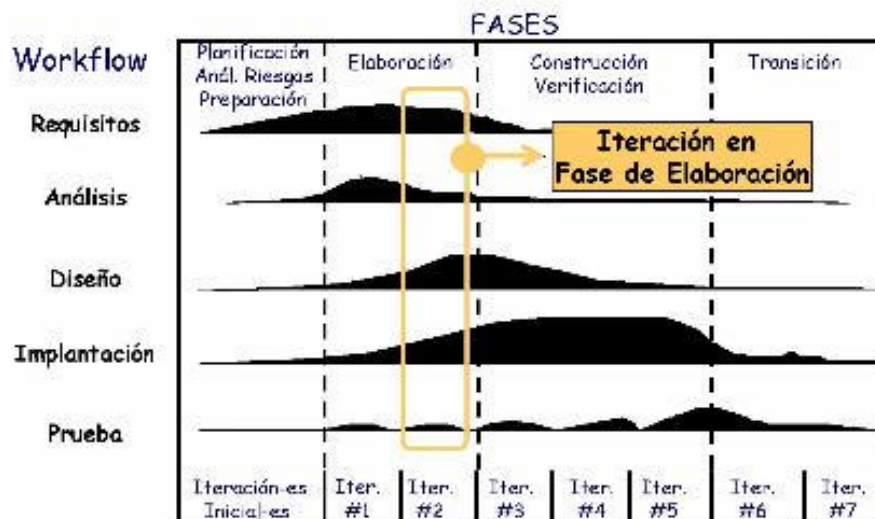


Figura 1.1 El proceso fases, iteraciones y actividades.  
(Adaptado de Jacobson, 1999).

**Mantenimiento:** se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software y a cambios debidos a las mejoras producidas por los requisitos cambiantes del cliente.

Durante la fase de mantenimiento se encuentran cuatro tipos de cambios:

**Corrección.** El mantenimiento correctivo cambia el software para corregir los defectos.

**Adaptación.** El mantenimiento adaptativo produce modificación en el software para acomodarlo a los cambios de su entorno externo.

**Mejora.** El mantenimiento perfectivo lleva al software más allá de sus requisitos funcionales originales.

**Prevención.** En esencia, el mantenimiento preventivo (reingeniería de software) hace cambios en programas de computadora a fin de que se puedan corregir, adaptar y mejorar más fácilmente.

## 1.5 Diagrama de casos de uso

Un Diagrama de Casos de Uso (Use Case Diagram) describe parte o el total del modelo de casos de uso y actores incluyendo sus interacciones, todo sistema tiene como mínimo un diagrama Main Use Case, que es una representación gráfica del entorno del sistema (actores) y su funcionalidad principal (casos de uso).

Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones).<sup>[4]</sup>

En la Figura 1.2, se muestra un ejemplo de Diagrama de Casos de Uso del sistema MODEL\*, dónde se pueden ver las interacciones de los actores, (en este caso el Empleador (Empresa), Desempleado (La persona que busca trabajo) y el Administrador) con el sistema.

---

\* Módulo de Ofertas y Demandas de Empleo en Línea, sistema del cual es objeto esta tesis y se detalla en el capítulo 2.

## 1.6 Casos de Uso

Un caso de uso es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización de software. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico. Normalmente, en los casos de usos se evita el empleo de palabras técnicas, prefiriendo en su lugar un lenguaje más cercano al usuario final. En ocasiones, se utiliza a usuarios sin experiencia junto a los analistas para el desarrollo de casos de uso.

Para cada caso de uso deben especificarse sus caminos o secuencias de acciones posibles.<sup>[5]</sup>

En la figura 1.2, se observa que en este sistema existen 3 actores los cuales son:

- **Administrador:** Es la persona encargada del mantenimiento del sistema, la cual tiene todos los privilegios.
- **Empleador:** Es el actor más importante, ya que el sistema funcionará en base a la información que este actor agregue, es decir el debe ingresar al menos una vacante, para que el desempleado pueda realizar por lo menos una búsqueda.
- **Desempleado:** Es el actor beneficiado por el sistema, puede ser cualquier persona y podrá consultar las vacantes publicadas.

Los casos de uso importantes que se obtienen son:

- **Administrar Bolsa de Trabajo:** Solo lo puede realizar el Administrador del sistema, y contiene opciones para agregar, eliminar, modificar o consultar a los usuarios registrados.

- Administrar Vacantes: Contiene opciones de Altas, Bajas, Modificaciones y Consultas de los empleos, y este caso de uso lo realiza solo el actor Empleador.
- Buscar empleo: Es el caso de uso que ejecuta el actor Desempleado, y podrá realizar búsquedas de vacantes.

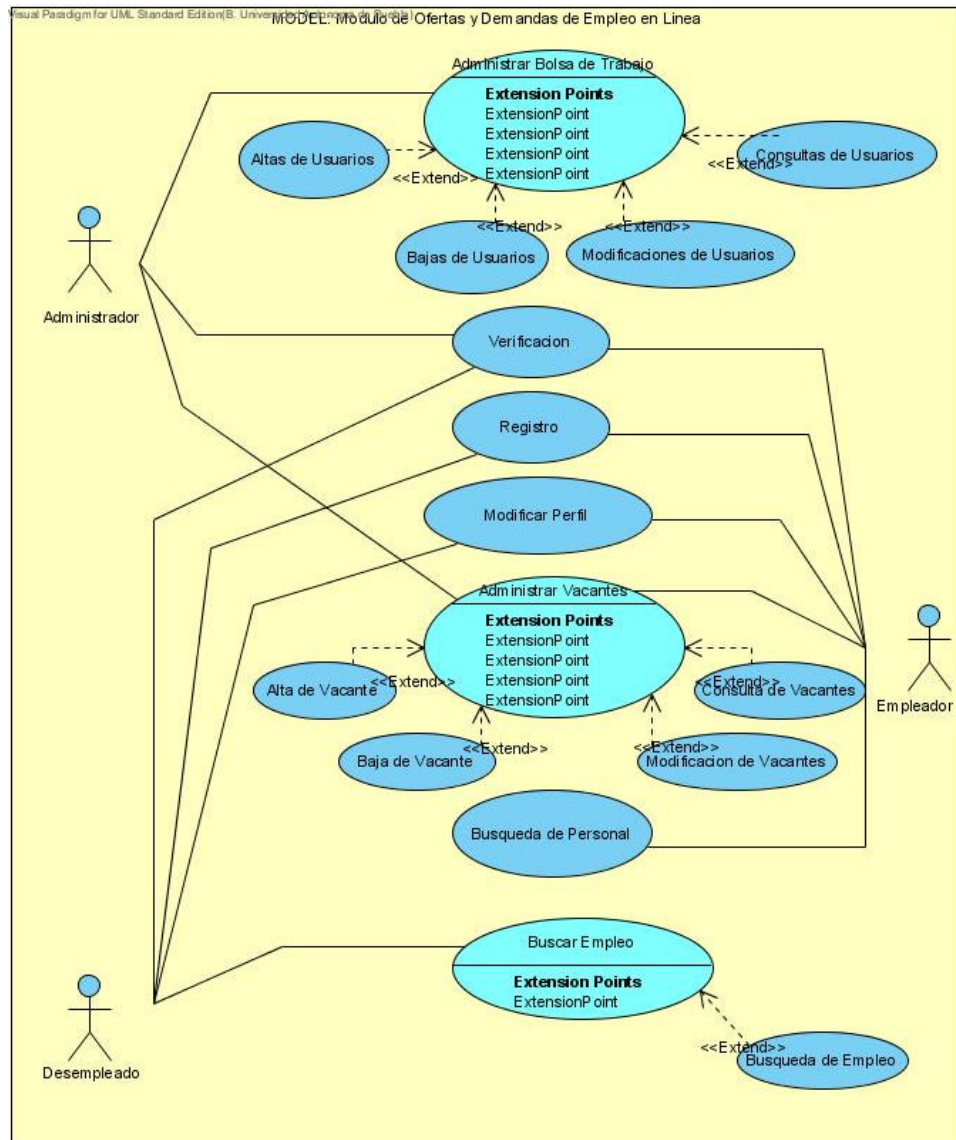


Figura 1.2 Diagrama de Casos de Uso del sistema MODEL (Módulo de Ofertas y Demanda de Empleo en Línea).

## 1.7 Modelo de Análisis

En este modelo se describen un conjunto de Clases de Análisis que se utilizan para realizar una descripción abstracta de la realización de los casos de uso del modelo de casos de uso. Las clases de análisis luego evolucionan hacia otras clases más detalladas en el Modelo del Diseño, ver figura 1.3.

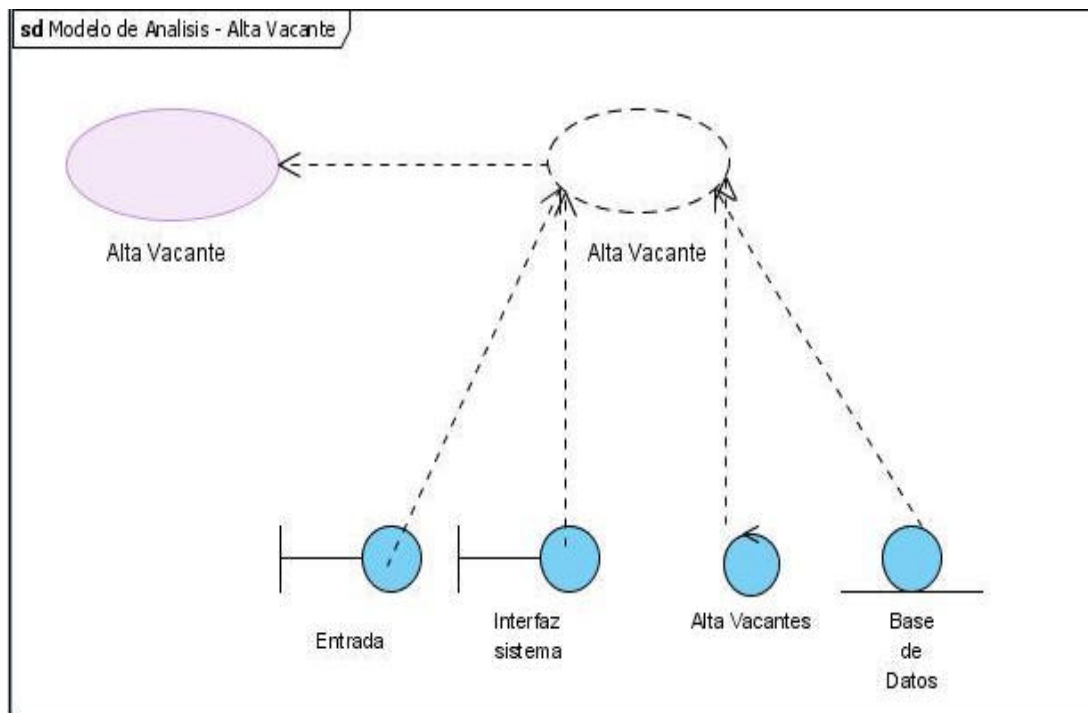


Figura 1.3 Caso de Uso Alta Vacante en el Modelo de Análisis.

Durante el Análisis también se pueden utilizar diagramas de colaboración para describir la realización de un caso de uso, veamos el ejemplo de la figura 1.4.

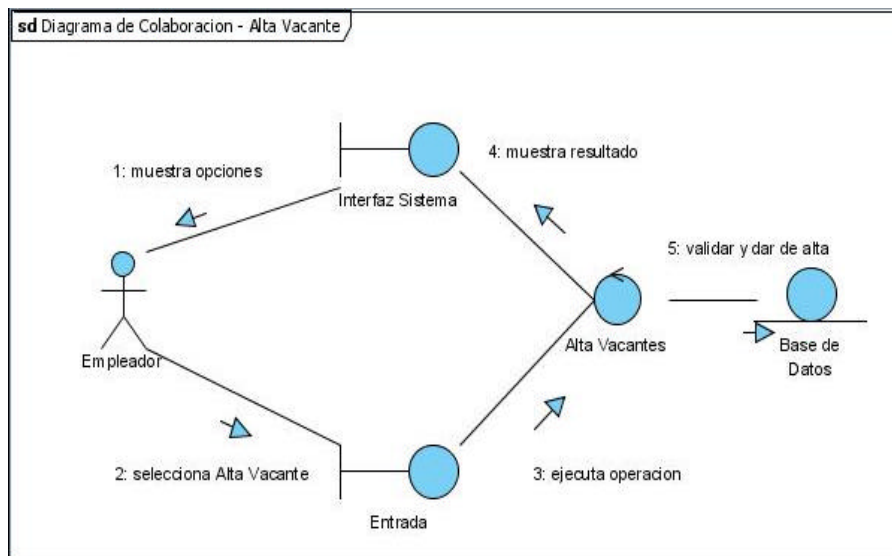


Figura 1.4 Diagrama de colaboración para el caso de uso Alta Vacante.

### 1.8 Modelo de diseño

Este modelo se crea tomando el modelo de análisis como entrada principal. Es similar al modelo de análisis ya que incluye clasificadores, relaciones y realizaciones de casos de uso, y existe una relación entre los artefactos del diseño y los del análisis, ver figura 1.5.

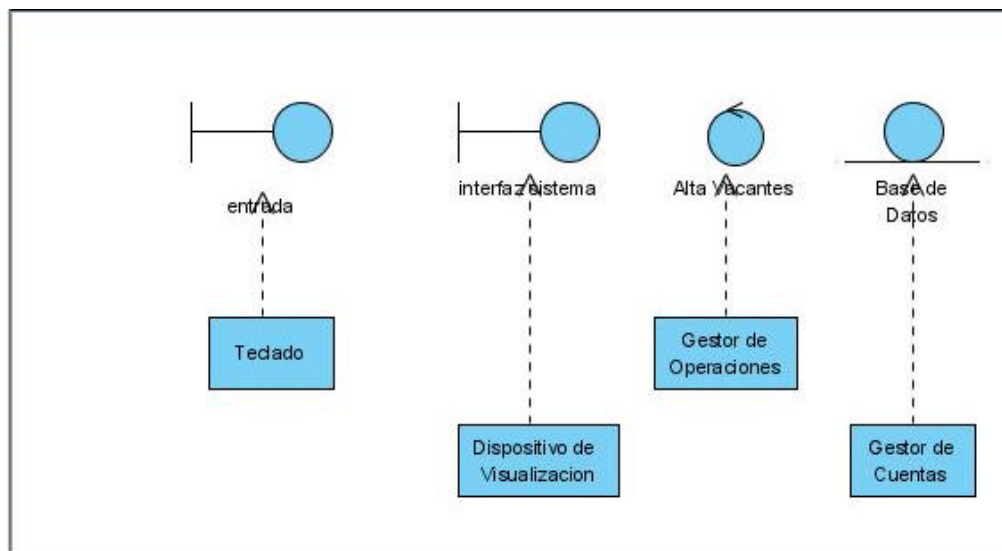


Figura 1.5 Las clases de diseño refinan las clases de análisis.

## 1.9 El Lenguaje Unificado de Modelado

Es un lenguaje para **visualizar, especificar, construir y documentar** los artefactos de un sistema de software. UML puede usarse en las diferentes etapas del ciclo de vida del desarrollo y en diferentes tecnologías de implementación, y es independiente del proceso de desarrollo de software.

Una vista es un subconjunto de construcciones de modelado que se enfocan en un aspecto en particular del sistema. Las vistas pueden dividirse en tres áreas **clasificación estructural, comportamiento dinámico, y gestión del modelo**.

Las vistas de **clasificación estructural** incluyen:

- ✓ Vista estática
  - **Diagrama de clases:** muestra las clases, interfaces, colaboraciones y sus relaciones. Son los más comunes y dan una vista estática del proyecto.
- ✓ Vista de casos de uso
  - **Diagrama de casos de uso:** Muestran los casos de uso, actores y sus relaciones. Son muy importantes para modelar y organizar el comportamiento del sistema.
- ✓ Vista de implementación
  - **Diagrama de componentes:** Muestran la organización de los componentes del sistema. Un componente se corresponde con una o varias clases, interfaces o colaboraciones.
  - **Diagrama de despliegue:** Muestra los nodos y sus relaciones. Un nodo es un conjunto de componentes. Se utiliza para reducir la complejidad de los diagramas de clases y componentes de un gran sistema. Sirve como resumen e índice.

El **comportamiento dinámico** describe el comportamiento del sistema a través del tiempo.

Las vistas de comportamiento dinámico incluyen:

- ✓ Vista de la máquina de estados
  - **Diagrama de estados (ciclo de vida de objeto):** muestra los estados, eventos, transiciones y actividades de los diferentes objetos. Son útiles en sistemas que reaccionen a eventos.

- ✓ Vista de actividades
  - **Diagrama de actividades:** Es un caso especial del diagrama de estados. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.
- ✓ Vista de interacción
  - **Diagrama de secuencia y Diagrama de colaboración:** Muestran a los diferentes objetos y las relaciones que pueden tener entre ellos, los mensajes que se envían entre ellos. Son dos diagramas diferentes, que se puede pasar de uno a otro sin pérdida de información, pero que nos dan puntos de vista diferentes del sistema. En resumen, cualquiera de los dos es un Diagrama de Interacción.

La **gestión del modelo** describe la organización de los modelos mismos en unidades jerárquicas.

- ✓ Vista de gestión
  - **Diagrama de clases (paquetes, subsistemas, modelos):** muestra las clases, interfaces, colaboraciones y sus relaciones. Son los más comunes y dan una vista estática del proyecto.

Estos son los diferentes tipos de diagramas con los que cuenta UML, en la mayoría de los casos no se requieren de todos, y UML permite definir solo los necesarios.

## 1.10 Diagrama de Secuencias

El Diagrama de secuencia sirve para representar la realización de un caso de uso, en específico en el modelo de diseño, este tipo de diagrama muestra la interacción entre el “Actor” y el sistema.

A continuación se muestra el diagrama de secuencias para el caso de uso “Alta de Vacante” (Figura. 1.6), este caso de uso lo puede realizar solamente el “Actor” “Empleador”, quien es el contacto de alguna empresa en específico.

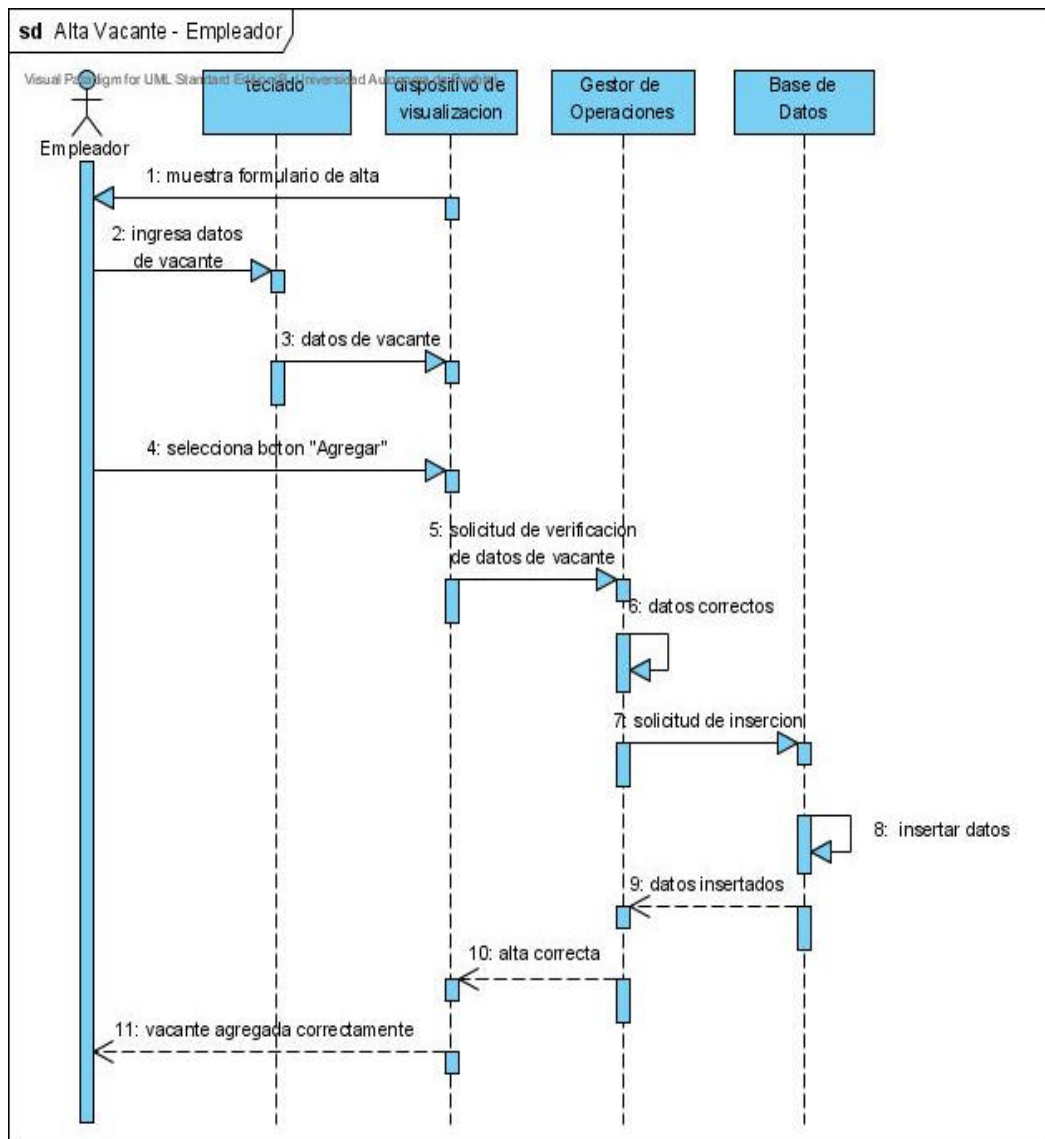


Figura 1.6 Diagrama de Secuencias, del caso de uso “Alta Vacante”.

### 1.11 Bases de datos

Se refiere a la colección de información que existe a lo largo de un período de tiempo, a menudo de varios años. Más claramente, el término base de datos se refiere a una colección de datos gestionada por un sistema gestor de base de datos, (S. G. B. D.), o simplemente gestor de base de datos.<sup>[6]</sup>

Nosotros consideramos como bases de datos a la colección de datos junto con el S. G. B. D., idealmente el Sistema Gestor de Bases de Datos debe de poseer una serie de características indispensables para satisfacer a los usuarios.

- ✓ Debe de poseer un lenguaje de definición de datos que permita fácilmente la creación de nuevas bases de datos. Así como la modificación de su estructura.
- ✓ Debe poseer un lenguaje de manipulación de datos, que permita la inserción, eliminación, modificación y consulta de los datos de la base de datos, de la forma más eficiente y conveniente posible.
- ✓ Debe permitir el almacenamiento de enormes cantidades de datos (miles de millones de caracteres), sin que el usuario perciba una degradación en cuanto al rendimiento global del sistema.
- ✓ Debe permitir la gestión segura de los datos, con respecto a accesos no autorizados, y accidentes producidos por los dispositivos mecánicos o electrónicos que soportan los datos almacenados.
- ✓ Debe permitir el acceso simultáneo por parte de varios usuarios, impidiendo además, que dichos accesos concurrentes den lugar a datos corruptos o inconsistentes.

Estas características dan lugar a otra definición de base de datos: una organización coherente de datos permanentes, y accesibles para usuarios concurrentes.

Esta definición lleva implícito el concepto de S. G. B. D., ya que es éste quien se debe de encargar de almacenar los datos (y por tanto de hacerlos permanentes y no volátiles), de gestionar su integridad (y por tanto de que sean coherentes), y de controlar a los múltiples usuarios (permitiendo así la concurrencia).<sup>[7]</sup>

## **1.12 Componentes de un sistema de Base de Datos**

### **Datos.**

Efectivamente una base de datos no tiene sentido sino esta compuesta de datos. Lo que no está tan claro es la forma en que estos datos se deben disponer, que datos se deben almacenar, y cómo los debe entender la computadora.

La disposición de los datos depende del ámbito de aplicación concreto en que se enmarque la base de datos.

### **Metadatos.**

Es la información que el sistema guarda sobre los datos almacenados. Por ejemplo, en una biblioteca se usan fichas que especifican autores, títulos, casas editoriales y lugares para buscar libros. Así, los metadatos ayudan a ubicar datos.

## **1.13 Sistema Gestor de Bases de Datos.**

### **1.13.1 Modelos de Datos**

Es un conjunto de conceptos que sirven para describir la estructura de una base de datos, los datos, las relaciones entre los datos y las restricciones que deben cumplirse sobre los datos.

Los modelos de datos contienen también un conjunto de operaciones básicas para la realización de consultas (lecturas) y actualizaciones de datos. Además, los modelos de datos más modernos incluyen conceptos para especificar comportamiento, permitiendo especificar un conjunto de operaciones definidas por el usuario.

#### **1.13.1.1 Los Modelos Lógicos.**

Estos conceptos pueden ser entendidos por los usuarios finales, aunque no están demasiado alejados de la forma en que los datos se organizan físicamente. Los modelos lógicos ocultan algunos detalles de cómo se almacenan los datos, pero pueden implementarse de manera directa en la computadora.

### 1.13.1.2 Los Modelos Conceptuales.

Utilizan modelos como entidades, atributos y relaciones. Una *entidad* representa un objeto o concepto del mundo real como, por ejemplo, un empleado de la empresa inmobiliaria o una oficina. Un *atributo* representa alguna propiedad de interés de una entidad como, por ejemplo, el nombre o el salario del empleado. Una *relación* describe una interacción entre dos o más entidades, por ejemplo, la relación de trabajo entre un empleado y su oficina.

## 1.14 Diseño conceptual de bases de datos.

### 1.14.1 El modelo entidad – relación (M E-R)

Este esquema conceptual, representa la información de forma absolutamente independiente al Sistema Gestor de Base de Datos. Los esquemas internos de las diferentes bases de datos no captan suficientemente bien la semántica del mundo real, de ahí que primero haya que pasar por uno o dos esquemas previos más cercanos al mundo real.

## 1.15 Componentes del modelo

**Entidad:** Se trata de cualquier objeto u elemento (real o abstracto) acerca del cual se pueda almacenar información en la base de datos. Es decir cualquier elemento informativo que tenga importancia para una base de datos. Ejemplos de entidades son Luís, la vacante de “Programador”, el sueldo de \$1,200, etc., un ejemplo se muestra en la figura 1.7.

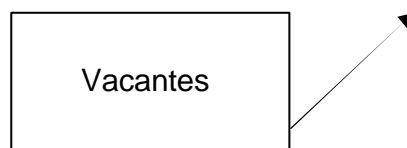


Figura 1.7 Ejemplo de una entidad (Vacante).

**Atributos:** Son aquellos que describen características o propiedades de un conjunto de entidades. Un atributo se representa de la siguiente manera, ver figura 1.8:

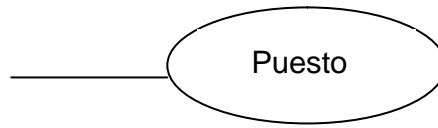


Figura 1.8 Ejemplo de un Atributo de la Entidad Vacante.

**Dominios:** Es el conjunto de valores posibles que puede tomar un atributo. El dominio de un atributo simple consta de una descripción física y una semántica. La descripción física indica el tipo de datos: número o cadenas de texto, la longitud de los datos y otras descripciones por limitaciones. La descripción semántica indica la función o propósito del atributo, lo que distingue a este atributo de otros que pudieran tener la misma descripción física.

### 1.16 Clasificación de los atributos.

**Simples y Compuestos:** Los atributos simples son aquellos que no están divididos en subpartes. Los atributos compuestos se pueden dividir en subcarpetas (es decir, en otros atributos). Los atributos compuestos se pueden dividir en subpartes (es decir, en otros atributos). Los atributos compuestos ayudan a agrupar los atributos relacionados, haciendo los modelos más claros.

**Univalorados y multivalorados:** Los atributos univalorados son aquellos que tienen un valor solo para una entidad concreta. Los atributos multivalorados tienen un conjunto de valores para una entidad específica. En ellos se pueden colocar apropiadamente límites inferiores y superiores en el número de valores en el atributo multivalorado.

**Nulos:** un valor nulo se usa cuando una entidad no tiene un valor para un atributo. Nulo también designa que el valor de un atributo es desconocido. Un valor desconocido también puede ser bien **perdido** (el valor existe pero no tiene esa información) o **no conocido** (no se conoce si el valor existe realmente o no).

**Derivado:** El valor de este atributo se puede derivar de los valores de otros atributos o entidades.

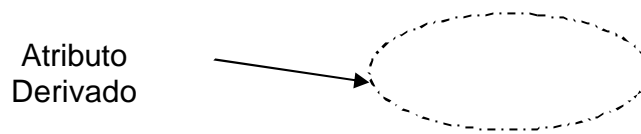


Figura 1.9 Atributo Derivado.

**Atributo descriptivo:** Si un conjunto de relaciones tiene también algunos atributos asociados a él, entonces se unen esos atributos a ese conjunto de relaciones.

**Relación:** La relación es la asociación entre 2 o mas entidades de una relación.

**Conjunto de relaciones:** Es un conjunto de relaciones del mismo tipo.

**Grado de la relación:** La cantidad de entidades en una relación es el grado de la relación.

### 1.17 Tipos de clave

**Superclave:** Es un conjunto de uno o más atributos que, tomados colectivamente, permiten identificar o reconocer de forma única una entidad en el conjunto de entidades. Este tipo de claves contiene comúnmente atributos que no son indispensables para llevar a cabo el reconocimiento del registro.

**Clave candidata:** Una superclave puede contener atributos innecesarios. Tales superclaves mínimas se llaman claves candidatas.

**Clave primaria:** Es aquella clave que el diseñador de la base de datos selecciona entre las claves candidatas encontradas, como elemento principal para identificar las entidades dentro de un conjunto de entidades.

## 1.8 Normalización

La normalización es una ayuda que proporciona un procedimiento riguroso para el diseño de bases de datos. Una base de datos mal diseñada puede funcionar inicialmente pero puede mostrar anomalías en el almacenamiento debidas al agrupamiento de los campos cuando se efectúan las operaciones de inserción, actualización o eliminación.<sup>[8]</sup>

Las relaciones normalizadas se agrupan en cuatro categorías llamadas formas normales FN siendo cada nivel una descomposición más completa de una relación que la del nivel anterior.

El proceso de normalización empieza con la combinación de todos los datos de la base en una relación, la que a su vez se descompone en dos o más relaciones más pequeñas. Se efectúan descomposiciones sucesivas de las relaciones intermedias hasta que todas las relaciones obtenidas pertenecen a la cuarta forma normal (4FN).

**Primera Forma Normal:** Una relación está en 1FN si todos los campos en cada registro contienen un solo valor tomado de sus dominios respectivos, es decir en cada campo solo debe de existir un valor.

**Segunda Forma Normal:** Una relación es segunda forma normal si es 1FN y cada atributo no-clave de la relación es total y funcionalmente dependiente de su clave principal.

**Tercera Forma Normal:** Una relación es 3FN, si es 2FN y ningún atributo no-clave en la relación es funcionalmente dependiente de algún otro atributo no-clave.

**Forma Normal de Boyce/Codd:** Una relación es BFNC si cada determinante en la relación es una clave aspirante. Si existe algún atributo que resulte ser total y funcionalmente dependiente de otro, se le llama determinante.

**Cuarta Forma Normal:** Una relación es 4FN si es BFCN y no contiene dependencias multivalores.

## CAPITULO 2. ANÁLISIS DEL SISTEMA

### 2.1 Planteamiento del Problema: Modelo del negocio

El problema de la bolsa de trabajo radica en que no se publican ofertas de empleo disponibles en otros estados, es decir, en los periódicos que circulan en el estado de Puebla, solo se puede tener acceso a la información de trabajos de ese estado, lo que provoca que la gente no se entere de otros trabajos en otros estados.

Se elaborará una base de datos, donde todas las personas que busquen un trabajo se puedan registrar, de igual manera los contactos de las empresas (Jefes de Recursos Humanos) se podrán registrar para ofertar sus vacantes, de ésta manera una empresa podrá buscar personal en línea de acuerdo a sus necesidades, y las personas (desempleados) podrán encontrar la información necesaria para contactar a la persona encargada de dicha oferta de trabajo y así, conciliar una cita.

La base de datos deberá contener toda la información relacionada con una vacante como lo es: id de la vacante, Puesto, años de experiencia necesarios, conocimientos básicos, grado mínimo de estudios, edad, sexo, sueldo, turno, tipo contrato, localidad estado, si requiere viajar, cambio de residencia y fecha para presentarse a laborar.

Así mismo contará con otra tabla llamada “empleador” que almacenará la información relacionada con el contacto de la empresa y tendrá los campos siguientes: id de empresa, nombre de la empresa, RFC de la empresa, dirección de la empresa, colonia, código postal, entidad federativa, estado, giro, nombre del contacto, teléfono, correo, usuario y contraseña.

Otra tabla importante para este sistema es la de “desempleado”, que guardará la información de las personas que buscan un trabajo y tendrá los siguientes campos: id de usuario, usuario, contraseña, nombre, fecha de nacimiento, edad, sexo, lugar de nacimiento, nacionalidad, dirección, localización, colonia, código postal, entidad federativa, estado, estado civil, estatura, compleción, número del seguro social, teléfono, correo, número de licencia, número de cartilla de servicio militar nacional y RFC.

Además de las tablas mencionadas, existe otra y es la que tendrá relación con las anteriores, la tabla de solicitudes, que tendrá los siguientes campos: id de

solicitud, fecha de solicitud hecha por el “desempleado” y fecha en la que se puede presentar a laborar.

Existen otras tablas, las cuales almacenarán información relacionada con los estudios realizados por el “desempleado”, así como los empleos anteriores en los que laboró, idiomas que habla y otros cursos a los que haya asistido.

Para el buen funcionamiento del sistema la base de datos deberá permitir realizar operaciones para el manejo de la información con respecto a las “Altas de vacantes”; donde se agregarán nuevas ofertas de trabajo, “Bajas de Vacantes”; para eliminar una oferta de trabajo que haya sido ocupada, “Modificaciones de Vacantes”, para editar la información de alguna vacante y por supuesto “Consultas de Vacantes”; para tener acceso a un listado de todas las vacantes dadas de altas.

En cuanto a las consultas por parte del “desempleado”, la base de datos deberá permitir realizar consultas con texto específico, es decir, el usuario tendrá la opción de ingresar una palabra y se tendrá que hacer la consulta con esa palabra.

## **2.2 Objetivos**

Generales:

- ✓ Crear una base de datos relacional con información sobre personas del ámbito laboral (“empleadores” y “desempleados”).
- ✓ Difundir a gran escala las ofertas de trabajo de una empresa.

Específicos:

- ✓ Permitir el registro de varias personas para la búsqueda de empleo y de personal.
- ✓ Contener la información necesaria para poder realizar el contacto entre una persona “desempleada” y un “empleador”.

- ✓ Lograr que un “empleador” encuentre a su personal desde el sistema en línea.

### 2.3 Definición de Requerimientos

El sistema Módulo de Ofertas y Demandas de Empleo en línea (MODEL) tendrá que ser capaz de realizar las operaciones requeridas, para tener un buen funcionamiento, las operaciones que debe ejecutar son:

- ✓ Alta de Usuarios
- ✓ Baja de Usuarios
- ✓ Modificación de Usuarios
- ✓ Consultas de Usuarios
- ✓ Alta de Vacantes
- ✓ Baja de Vacantes
- ✓ Modificación de Vacantes
- ✓ Consulta de Vacantes
- ✓ Búsqueda de Personal
- ✓ Búsqueda de Empleo

Al cumplir con estas operaciones se puede decir que el sistema está funcionando de manera apropiada a lo especificado en los requerimientos.

Este sistema contará con tres actores principales que son: el empleador, el desempleado y el administrador, los cuales podrán ejecutar las siguientes operaciones:

**Empleador:** Realizará todas las operaciones relacionadas con las vacantes; alta de vacantes, baja de vacantes, modificación de vacantes, consulta de vacantes y Búsqueda de personal.

*Alta de vacantes:* permite al usuario registrar una nueva vacante dentro de la base de datos.

*Baja de vacante:* aquí el usuario podrá quitar una vacante ofertada por el, que ya haya sido cubierta.

*Modificación de vacante:* en caso de haber cometido un error en la captura de información de una vacante, el usuario tendrá la opción de corregir los datos mediante esta acción.

*Consulta de vacantes:* listará todas las vacantes registradas por el empleador.

*Búsqueda de personal:* al ejecutar esta acción el empleador encontrará la información de varios desempleados registrados en el sistema y podrá buscar mediante varios parámetros como lo son: estado donde radican, profesión, edad, conocimientos.

**Desempleado:** solo tendrá la opción de realizar búsquedas de empleo con diferentes parámetros, como son: puesto, sueldo, estado, tipo de contrato, etc.

*Búsqueda de empleo:* ésta opción le permitirá al usuario encontrar información de vacantes disponibles en alguna empresa, la búsqueda la podrá realizar mediante diferentes parámetros como son: puesto, sueldo, turno, tipo de contrato.

**Administrador:** llevará el control del sistema en cuanto a usuarios se refiere, ya que podrá hacer altas, bajas, modificaciones y consultas de usuarios, del mismo modo en caso de que así se requiera, podrá realizar altas, bajas, modificaciones y consultas de vacantes.

*Alta de usuario:* en caso de que se requiera, el administrador dará de alta un nuevo usuario (empleador o desempleado).

*Baja de usuario:* el administrador podrá eliminar un usuario (empleador o desempleado) de la base de datos.

*Modificación de usuario:* si algún usuario no capturó bien sus datos y necesita una modificación, el administrador realizará esta acción (en caso que el usuario no la pueda hacer).

*Consulta de usuario:* el administrador tendrá acceso a un listado de usuarios registrados (empleadores y desempleados) en la base de datos.

## 2.4 Análisis de factibilidad

Para la realización de este sistema en cuanto al diseño, desarrollo, pruebas y documentación se requiere del siguiente equipo:

- ✓ 2 computadoras de escritorio
- ✓ Procesador Pentium IV
- ✓ Disco duro de 40 GB
- ✓ 256 Mb de Memoria RAM
- ✓ Monitor VGA de 15"
- ✓ Internet de 1 Mb de velocidad

En cuanto al software que se determinó para el desarrollo de este sistema es:

- ✓ Sistema operativo Microsoft Windows Xp
- ✓ MySql 5.0.21
- ✓ Php 4.30
- ✓ Servidor Wamp 5
- ✓ Mozilla firefox 2.0.0.14

## 2.5 Casos de uso

Anteriormente en la definición de requerimientos, se presentaron a los actores que interactuarán con el sistema y se observó que son tres, a continuación describiremos la función de cada uno de ellos.

*Empleador:* Es el actor más importante del sistema, ya que el sistema depende de la información que agregue a la base de datos, puede ser cualquier persona que quiera ofertar una vacante de empleo.

*Desempleado:* Puede ser cualquier persona que se haya registrado, y el accederá a la información que agregue el empleador.

*Administrador:* Este actor no puede ser cualquier persona, debe tener conocimiento sobre el manejo del sistema, de preferencia ser licenciado en ciencias de la computación, y su función es la de mantener el sistema libre de información que no sea verídica.

Otra parte importante del sistema son los casos de uso y en la figura 1.2 (mostrada en el Capítulo 1), se puede observar que tenemos seis casos de uso, los cuales son los siguientes:

**Administrar vacantes:** se encarga de la administración de las vacantes registradas en el sistema, y aquí se pueden hacer altas, bajas, modificaciones y consultas de vacantes, este caso de uso es propio del empleador (aunque el administrador también lo puede hacer).

**Administrar bolsa de trabajo:** se encarga de la administración de usuarios del sistema y permite hacer altas, bajas, modificaciones y consultas. Este caso de uso solo lo podrá realizar el administrador del sistema.

**Buscar empleo:** este caso de uso es especial del actor desempleado, el podrá ejecutar esta acción para encontrar información sobre las vacantes de empleo.

**Registro:** este caso de uso es tanto para el empleador como para el desempleado, ya que tiene que hacer su registro previamente para poder acceder a las demás opciones.

**Modificar perfil:** al igual que el anterior caso de uso, éste lo pueden realizar el empleador y el desempleado.

**Verificación:** este caso de uso es obligatorio para los dos actores (empleador y desempleado), se deberá ejecutar este caso de uso cada vez que ingresen al sistema.

A continuación en la figura 2.1 se muestra el caso de uso administrar vacantes. En el se puede hacer: altas de vacantes, bajas de vacantes, consultas de vacantes y modificación de vacantes.

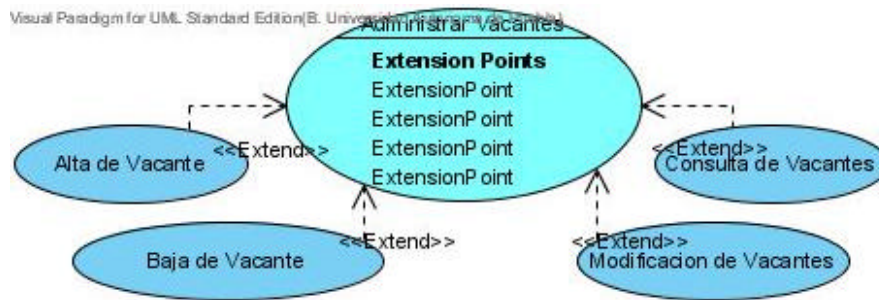


Figura 2.1 Caso de uso Administrar Vacantes.

Ahora mostraremos el caso de uso administrar bolsa de trabajo (figura 2.2) que incluye las opciones: alta de usuarios, baja de usuarios, modificación de usuarios y consulta de usuarios.

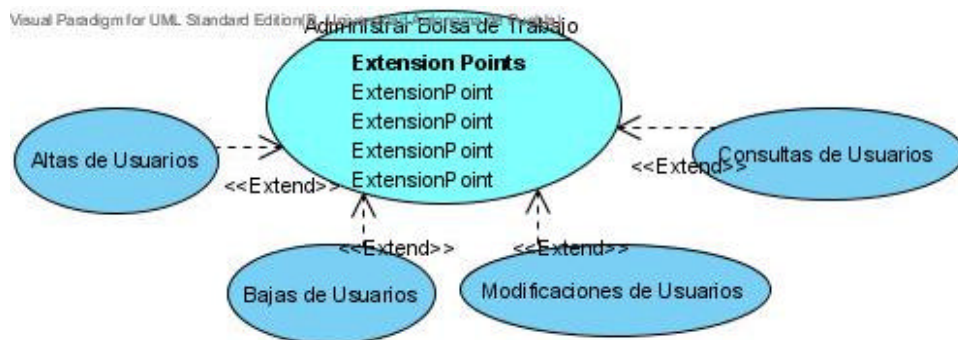


Figura 2.2 Caso de uso administrar bolsa de trabajo.

El caso de uso que se muestra en la figura 2.3 es buscar empleo y le corresponde al actor desempleado.

El caso de uso Registro mostrado en la figura 2.4, estará disponible para los dos actores (empleador y desempleado) y lo deberán ejecutar solo la primera vez, para quedar registrados en la base de datos del sistema.

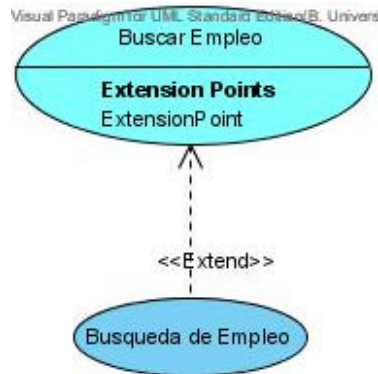


Figura 2.3 Caso de uso Buscar Empleo.

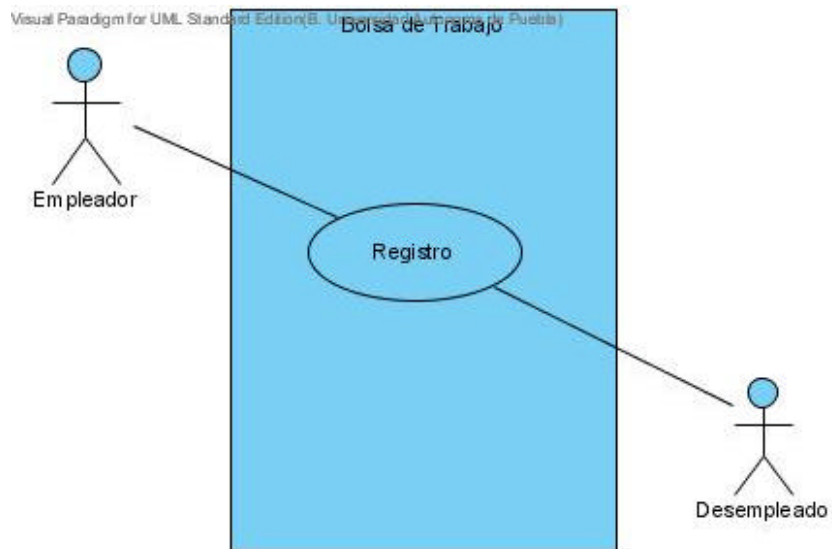


Figura 2.4 Caso de uso Registro.

El caso de uso modificar perfil (figura 2.5), también estará disponible para los dos actores del sistema (empleador y desempleado), aquí podrán cambiar sus datos de registro en caso de ser necesario.

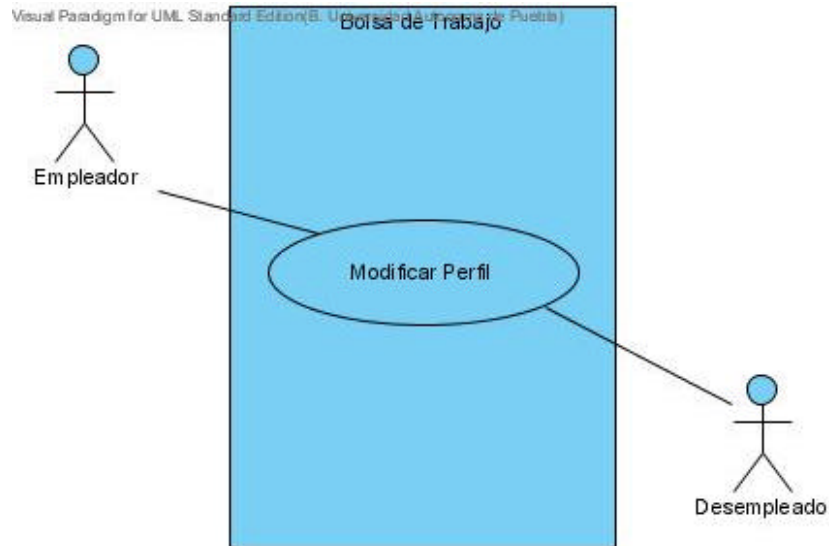


Figura 2.5 Caso de uso Modificar Perfil.

Por último se presenta el caso de uso verificación (figura 2.6), y de igual manera está disponible para ambos actores del sistema, pero también para el administrador y es necesario ejecutar este caso de uso cada vez que se presente un ingreso al sistema para verificar el nivel de usuario (usuario normal o administrador).

Estos son todos los casos de uso que intervienen en el sistema y que nos ayudarán a tener una perspectiva mejor del entorno del sistema.

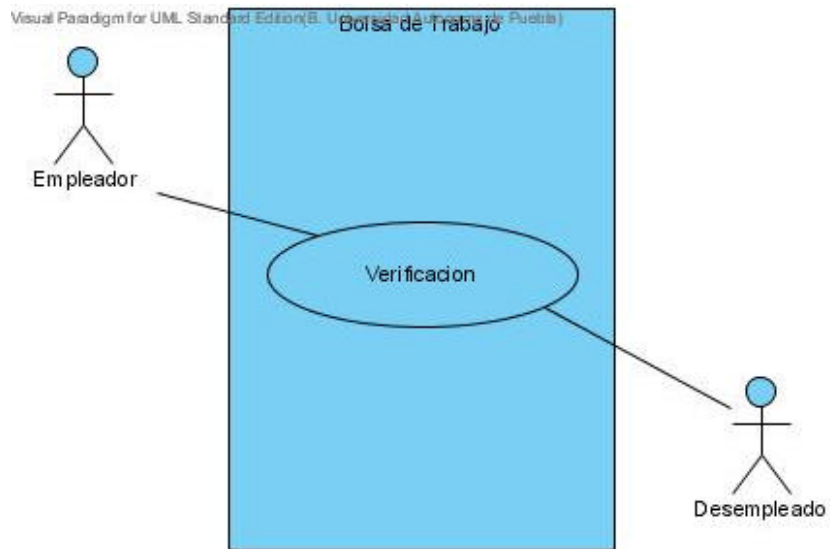


Figura 2.6 Caso de uso Verificación.

## 2.6 Diagrama de Secuencia

Como se mencionó anteriormente el diagrama de secuencia sirve para mostrar la interacción entre los objetos en un sistema, muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase.

El diagrama de secuencia (figura 2.7) se refiere al proceso que debe seguir el desempleado para registrarse. Se observa que el sistema solicita que se ingresen los datos y posteriormente se hace una petición al gestor de operaciones para verificar que esos datos sean correctos y en caso de que eso suceda se hace la inserción de los datos del nuevo usuario.

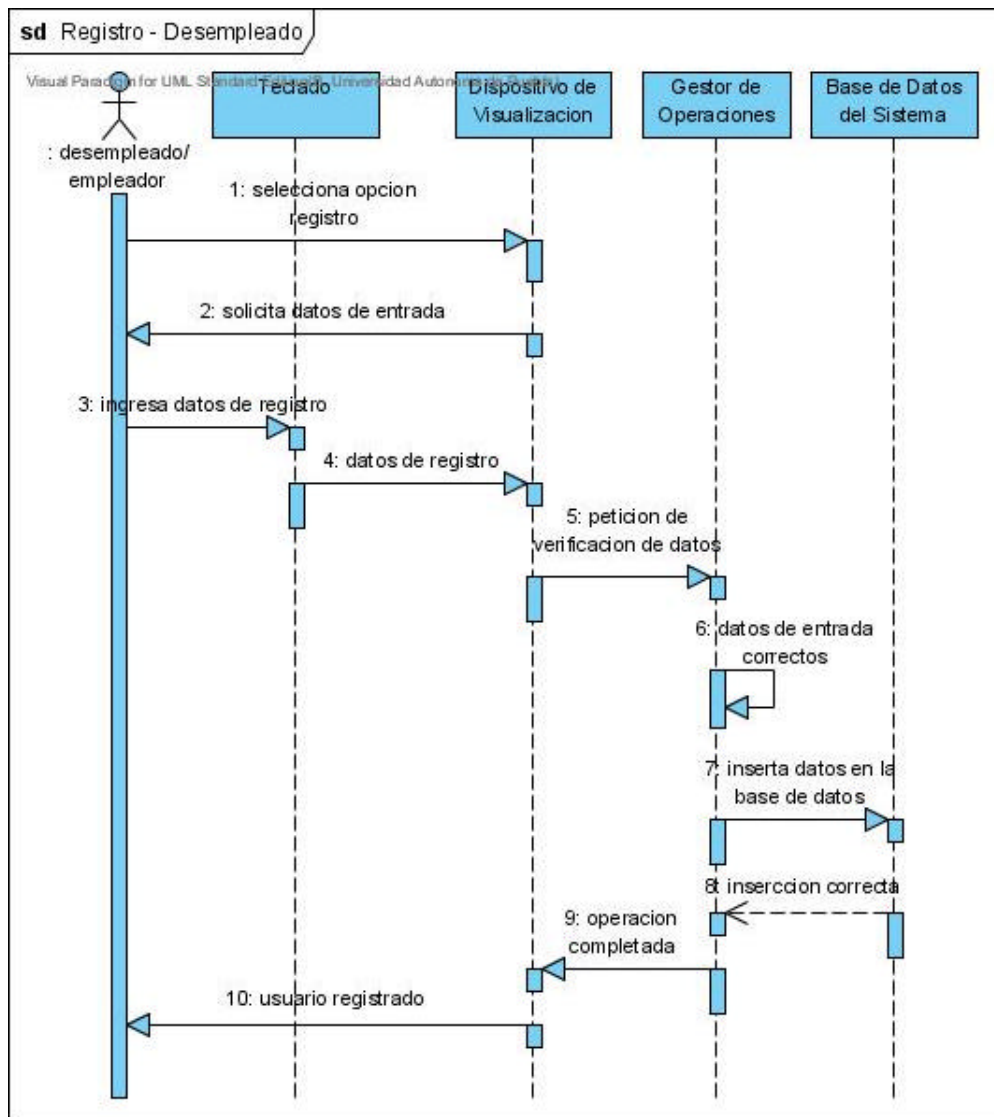


Figura 2.7 Diagrama de secuencia "Registro de un Desempleado".

En el diagrama de secuencia buscar personal (figura 2.8) se muestra el proceso a seguir por parte del empleador para buscar personal dentro del sistema, (es necesario que previamente se haya registrado).

Puede buscar personal por diferentes parámetros, puede ser por título (Licenciado, Ingeniero, Técnico, etc.), ingresa un título y se hace la solicitud de consulta con ese título y si existe alguna coincidencia con ese título en la base de datos, se mostrará la información ligada a ese título.

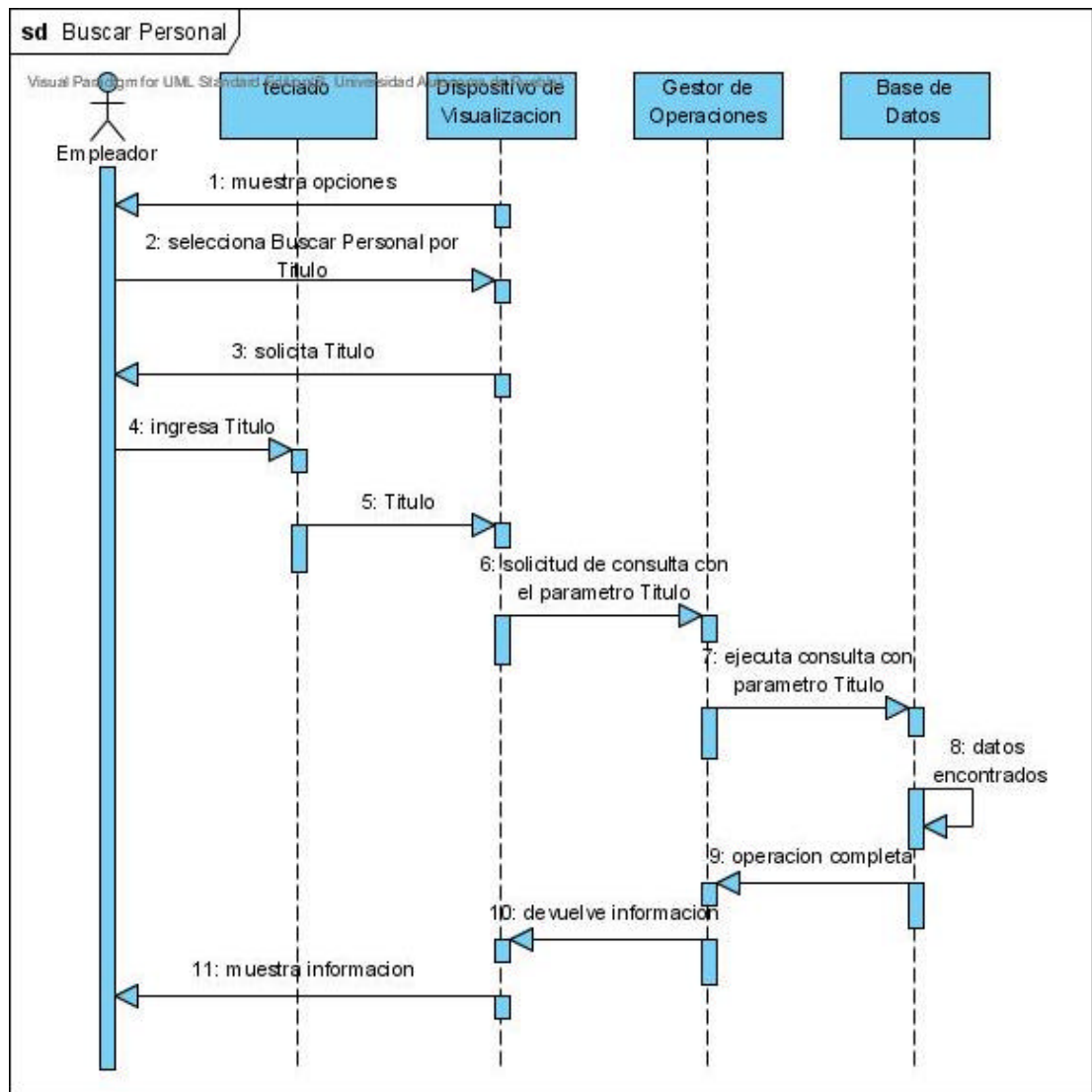


Figura 2.8 Diagrama de Secuencia Buscar Personal.

A continuación se presenta el diagrama de secuencia Búsqueda de Empleo, figura 2.9, que es ejecutado por el actor "Desempleado", de igual forma debe estar registrado para poder realizar esta operación.

El proceso inicia cuando se selecciona la opción búsqueda de empleo, el sistema muestra las diferentes opciones de búsqueda, después de que el usuario selecciona búsqueda por puesto, el sistema solicita puesto y hace una consulta con ese parámetro y dependiendo de lo encontrado se mostrará la información.

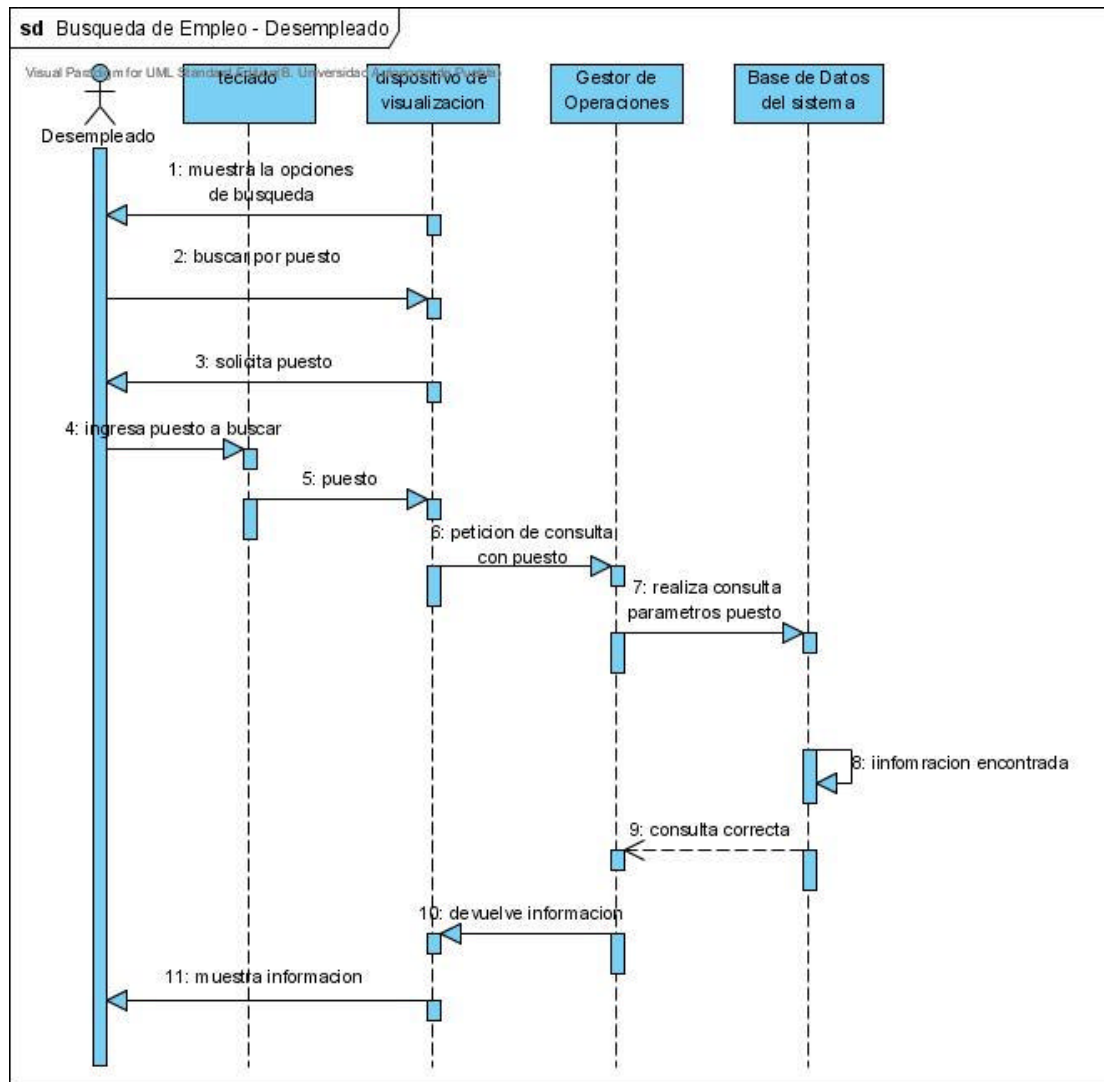


Figura 2.9 Diagrama de secuencia de búsqueda de empleo.

## CAPITULO 3. ANÁLISIS Y DISEÑO DE LA BASE DE DATOS ASOCIADA AL SISTEMA.

### 3.1 Diseño Conceptual

Es el proceso de construcción de un modelo de los datos utilizados en una organización, independientemente de las consideraciones físicas. Es una descripción detallada de la estructura de la base de datos, y no importa cuál es el SGBD que se va a utilizar para manipular la base de datos.

Un modelo conceptual es un lenguaje que se utiliza para describir esquemas conceptuales. El objetivo del diseño conceptual es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información.

El modelo conceptual de datos que se va a utilizar es el modelo Entidad-Relación (E-R) (figura 3.1), es uno de los modelos más expresivos y con conceptos únicos representado cada uno con un símbolo, permitiendo que se pueda leer de manera fácil y que pueda ser comprendido de igual manera.

En el diagrama E - R que se muestra en la figura 3.1, están contenidas las siguientes entidades:

**Empleador:** Representa al contacto de la empresa y contiene los campos sobre la información del contacto de la empresa.

**Desempleado:** Contiene información sobre las personas que buscan trabajo.

**Vacantes:** Almacena todo lo relacionado sobre las ofertas de trabajo disponibles en las empresas registradas.

**Solicitudes:** Es la entidad que se relaciona con vacantes y desempleado, por medio de esta se podrá saber que personas buscan trabajo.

### 3.2 Diagrama E-R

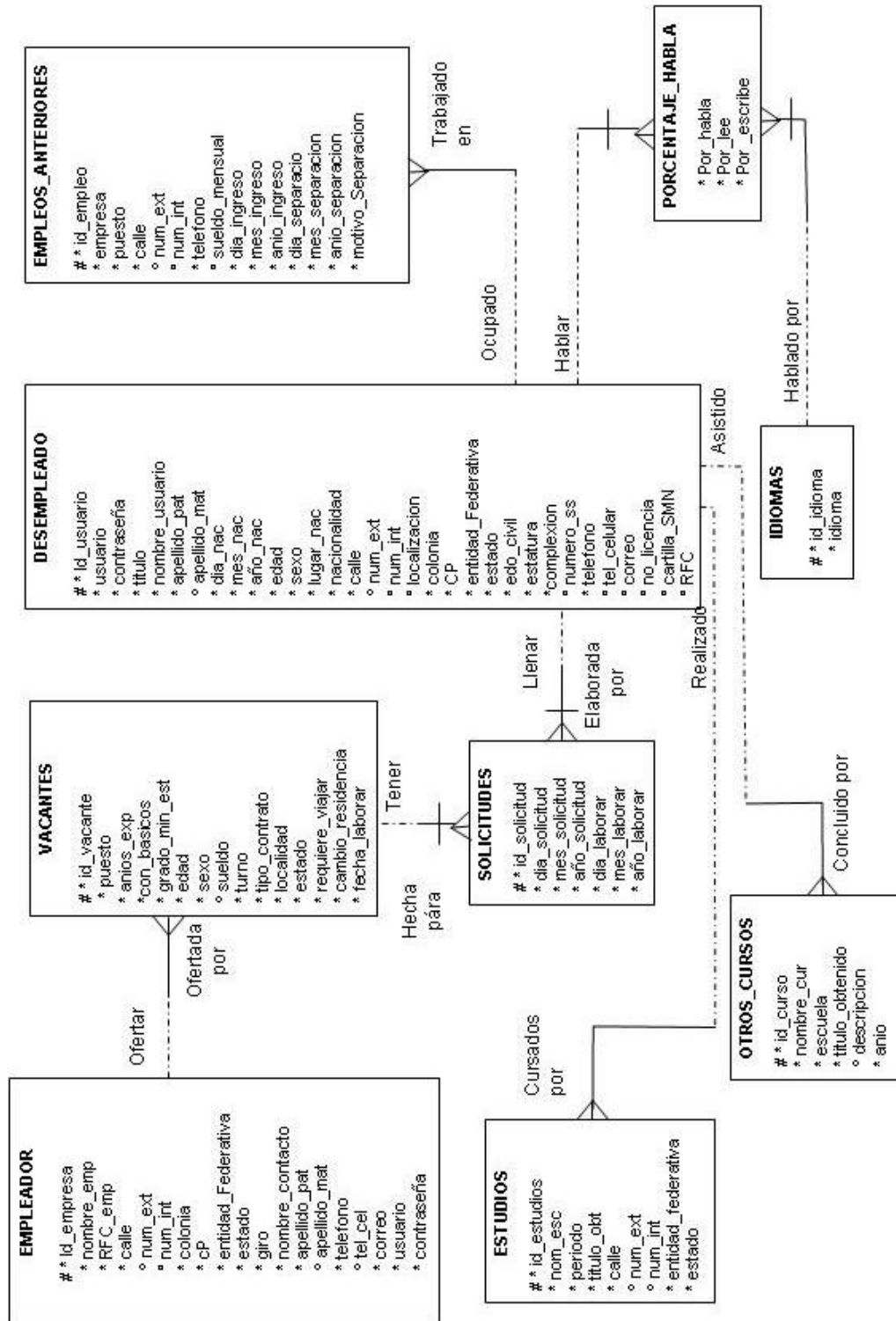


Figura 3.1 Diagrama E – R

Para el manejo de la información del desempleado se utilizan cinco entidades las cuales son:

**Estudios:** Almacena la información de los estudios realizados por los desempleados.

**Otros cursos:** Guarda información de cursos extras tomados por los desempleados.

**Empleos anteriores:** En esta entidad están registrados los empleos que pudo haber ocupado un desempleado.

**Idiomas:** Posee los idiomas mas conocidos y hablados, para que el desempleado elija los que habla y los relacione con la entidad porcentaje habla.

**Porcentaje de habla:** Se guardan los porcentajes de habla, lee y escribe de un idioma, para cada uno de los desempleados.

### 3.3 Diagrama Entidad – Relación en UML

Para ver mejor la estructura de la base de datos que se utilizará en el sistema, se muestra el Diagrama E – R de UML en la figura 3.2.

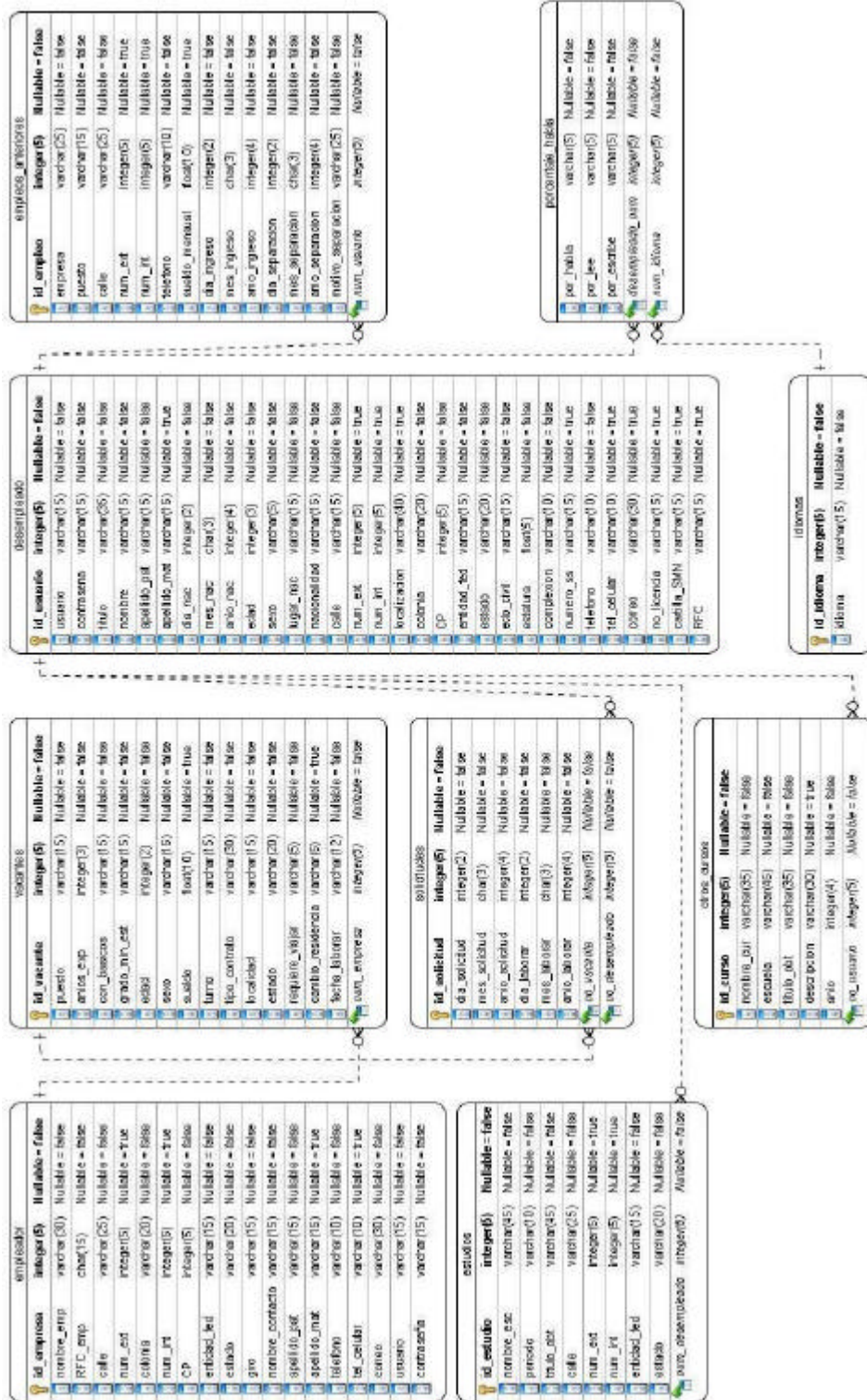


Figura 3.2 Diagrama Entidad – Relación en UML.

### 3.4 Diseño Lógico

A continuación se mostrará el mapeo a tablas utilizando como base diagrama E – R en UML (figura 3.2), las tablas se muestran a continuación.

**Tabla: Empleador**

Nom. de La columna	id_empresa	nombre_emp	RFC_emp	calle	num_ext	num_int	colonia	CP
Keytipo	PK							
Nulls	NN	NN	NN	NN			NN	NN

entidad_fed	estado	giro	nombre_contacto	apellido_pat	apellido_mat	telefono	tel_celular	correo
NN	NN	NN	NN	NN		NN		NN

usuario	contraseña
NN	NN

**Tabla: Vacantes**

Nom. de La columna	id_vacante	puesto	anios_exp	con_basicos	grado_min_est
Keytipo	PK				
Nulls	NN	NN	NN	NN	NN

edad	sexo	sueldo	turno	tipo_contrato	localidad	estado	requiere_viajar	cambio_residencia
NN	NN		NN	NN	NN	NN	NN	NN

fecha_laborar	num_empresa
NN	NN FK1

**Tabla: Solicitudes**

Nom. de La columna	id_solicitud	dia_solicitud	mes_solicitud	anio_solicitud	dia_laborar	mes_laborar
Keytipo	PK					
Nulls	NN	NN	NN	NN	NN	NN

anio_laborar	no_vacante	no_desempleado
NN	NN FK1	NN FK2

**Tabla: Estudios**

Nom. de La columna	id_estudio	nombre_esc	periodo	titulo_obt	calle	num_ext	num_int
Keytipo	PK						
Nulls	NN	NN	NN	NN	NN		

entidad_fed	estado	num_desempleado
		FK1
NN	NN	NN

**Tabla: Empleos Anteriores**

Nom. de La columna	id_empleo	empresa	puesto	calle	num_ext	num_int	telefono	sueldo_mensual
Keytipo	PK							
Nulls	NN	NN	NN	NN			NN	

dia_ingreso	mes_ingreso	anio_ingreso	dia_separacion	mes_separacion
NN	NN	NN	NN	NN

anio_separacion	motivo_separacion	num_usuario
		FK1
NN	NN	NN

**Tabla: Desempleado**

Nom. de La columna	id_usuario	usuario	contraseña	titulo	nombre	apellido_pat	apellido_mat	dia_nac
Keytipo	PK							
Nulls	NN	NN	NN	NN	NN	NN		NN

mes_nac	anio_nac	edad	sexo	lugar_nac	nacionalidad	calle	num_ext	num_int
NN	NN	NN	NN	NN	NN	NN		

localizacion	colonia	CP	entidad_fed	estado	edo_civil	estatura	complexion
	NN	NN	NN	NN	NN	NN	NN

numero_ss	telefono	tel_celular	corro	no_licencia	cartilla_SMN	RFC
	NN					

**Tabla: otros\_cursos**

Nom. de La columna	id_curso	nombre_cur	escuela	titulo_obt	descripcion	anio	no_usuario
Keytipo	PK						FK1
Nulls	NN	NN	NN	NN		NN	NN

**Tabla: idioma**

Nom. de La columna	id_idioma	idioma
Keytipo	PK	
Nulls	NN	NN

**Tabla: porcentaje\_habla**

Nom. de La columna	por_habla	por_lee	por_escribe	desempleado_num	num_idioma
Keytipo				FK1	FK2
Nulls	NN	NN	NN	NN	NN

### 3.5 Normalización:

Basándonos en el esquema lógico, se puede notar que todas las tablas están completamente normalizadas, ya que cumplen las reglas de primera forma normal (1FN), segunda forma normal (2FN), tercera forma normal (3FN), forma normal de Boyce/Codd (BFCN) y cuarta forma normal (4FN) mencionadas en el Capítulo 1.

### 3.6 Diccionario de Datos del Sistema

En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario de datos guarda los detalles y descripción de todos estos elementos.<sup>[9]</sup>

#### Empleador

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_empresa	Simple monovalorado	Identificador de empresa	Int Autoincrementable	5	No nulo
nombre_emp	Simple monovalorado	Nombre de la empresa	varchar	30	No nulo
RFC_emp	Simple monovalorado	Rfc de la empresa	char	15	No nulo
calle	Simple monovalorado	Dirección de la emp.	varchar	25	No nulo
num_ext	Simple monovalorado	Numero exterior	int	5	Ninguna
num_int	Simple monovalorado	Numero interior	int	5	Ninguna
colonia	Simple monovalorado	Colonia	varchar	20	No nulo
cp	Simple monovalorado	Código postal	int	5	No nulo
entidad_fed	Simple monovalorado	Entidad federativa	varchar	15	No nulo
estado	Simple monovalorado	Estado	varchar	20	No nulo
giro	Simple monovalorado	Giro de la empresa	varchar	15	No nulo
nombre_contacto	Simple monovalorado	Nombre del contacto	varchar	15	No nulo
apellido_pat	Simple monovalorado	Apellido paterno	varchar	15	No nulo
apellido_mat	Simple monovalorado	Apellido materno	varchar	15	Ninguna
telefono	Simple monovalorado	Telefono del contacto	varchar	10	No nulo
tel_celular	Simple monovalorado	Tel. celular del contacto	varchar	10	Ninguna
correo	Simple monovalorado	Correo del contacto	varchar	30	No nulo
usuario	Simple monovalorado	Nombre de usuario	varchar	15	No nulo
contraseña	Simple monovalorado	Contraseña del contacto	varchar	15	No nulo

## Desempleado

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_usuario	Simple monovalorado	Identificador de usuario	Int Autoincrementable	5	No nulo
usuario	Simple monovalorado	Nombre de usuario	varchar	15	No nulo
contraseña	Simple monovalorado	Contraseña de usuario	varchar	15	No nulo
titulo	Simple monovalorado	Título de desempleado	varchar	35	No nulo
nombre	Simple monovalorado	Nombre del desempleado	varchar	15	No nulo
apellido_pat	Simple monovalorado	Apellido paterno	varchar	15	No nulo
apellido_mat	Simple monovalorado	Apellido materno	varchar	15	Ninguna
dia_nac	Simple monovalorado	Día de nacimiento	int	2	No nulo
mes_nac	Simple monovalorado	Mes de nacimiento	char	3	No nulo
anio_nac	Simple monovalorado	Año de nacimiento	int	4	No nulo
edad	Simple monovalorado	Edad del desempleado	int	3	No nulo
sexo	Simple monovalorado	Sexo del desempleado	enum	'M' 'F'	No nulo
lugar_nac	Simple monovalorado	Lugar de nacimiento	varchar	15	No nulo
nacionalidad	Simple monovalorado	Nacionalidad	varchar	15	No nulo
calle	Simple monovalorado	Nombre de la calle	varchar	15	No nulo
num_ext	Simple monovalorado	Numero exterior	Int	5	Ninguna
num_int	Simple monovalorado	Numero interior	Int	5	Ninguna
localizacion	Simple monovalorado	Localizacion del hogar	varchar	40	Ninguna
colonia	Simple monovalorado	Nombre de la colonia	varchar	20	No nulo
CP	Simple monovalorado	Código postal	int	5	No nulo
entidad_fed	Simple monovalorado	Entidad federativa	varchar	15	No nulo
estado	Simple monovalorado	Estado donde habita	varchar	20	No nulo
edo_civil	Simple monovalorado	Estado civil	varchar	15	No nulo
estatura	Simple monovalorado	Estatura	float		No nulo
complexion	Simple monovalorado	Complexion	varchar	10	No nulo
numero_ss	Simple monovalorado	Numero del seguro social	varchar	15	Ninguna
telefono	Simple monovalorado	Telefono	varchar	10	No nulo
tel_celular	Simple monovalorado	Telefono celular	varchar	10	Ninguna
correo	Simple monovalorado	Correo electrónico	varchar	30	Ninguna
no_licencia	Simple	Numero de	varchar	15	Ninguna

	monovalorado	licencia			
cartilla_SMN	Simple monovalorado	Cartilla del servicio militar	varchar	15	Ninguna
RFC	Simple monovalorado	Registro Federal de Contribuyentes	varchar	15	Ninguna

## Vacantes

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_vacante	Simple monovalorado	Identificador de vacante	Int Autoincrementable	5	No nulo
puesto	Simple monovalorado	Puesto ofertado	varchar	15	No nulo
anios_exp	Simple monovalorado	Años de experiencia	int	3	No nulo
con_basicos	Simple monovalorado	Conocimientos básicos	varchar	15	No nulo
grado_min_est	Simple monovalorado	Grado mínimo de estudios	varchar	15	No nulo
edad	Simple monovalorado	Edad requerida	Int	2	No nulo
sexo	Simple monovalorado	Sexo requerido	varchar	15	No nulo
sueldo	Simple monovalorado	Sueldo ofertado	float		Ninguna
turno	Simple monovalorado	Turno de la vacante	varchar	15	No nulo
tipo_contrato	Simple monovalorado	Tipo de contrato	varchar	30	No nulo
localidad	Simple monovalorado	Localidad	varchar	15	No nulo
estado	Simple monovalorado	Estado donde se oferta	varchar	20	No nulo
requiere_viajar	Simple monovalorado	Requiere viajar	enum	'S' 'N'	No nulo
cambio_residencia	Simple monovalorado	Requiere cambio de residencia	enum	'S' 'N'	No nulo
fecha_laborar	Simple monovalorado	Fecha para laborar	varchar	12	No nulo
num_empresa	Simple monovalorado	Identificador de la empresa	int	5	No nulo

## Solicitudes

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_solicitud	Simple monovalorado	Identificador de solicitud	Int Autoincrementable	5	No nulo
dia_solicitud	Simple monovalorado	Día de solicitud	int	2	No nulo
mes_solicitud	Simple monovalorado	Mes de solicitud	char	3	No nulo

anio_solicitud	Simple monovalorado	Año de solicitud	int	4	No nulo
dia_laborar	Simple monovalorado	Día para presentarse	int	2	No nulo
mes_laborar	Simple monovalorado	Mes para presentarse	char	3	No nulo
anio_laborar	Simple monovalorado	Año para presentarse	int	4	No nulo
no_vacante	Simple monovalorado	Identificador de vacante	int	5	No nulo
no_desempleado	Simple monovalorado	Identificador de desempleado	int	5	No nulo

### Empleos anteriores

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_empleo	Simple monovalorado	Identificador de empleo	Int autoincrementable	5	No nulo
empresa	Simple monovalorado	Nombre de empresa	varchar	25	No nulo
puesto	Simple monovalorado	Puesto ocupado	varchar	15	No nulo
calle	Simple monovalorado	Nombre de la calle	varchar	25	No nulo
num_ext	Simple monovalorado	Numero exterior	int	5	Ninguna
num_int	Simple monovalorado	Numero interior	int	5	Ninguna
telefono	Simple monovalorado	Telefono	varchar	10	No nulo
sueldo_mensual	Simple monovalorado	Sueldo obtenido	float		Ninguna
dia_ingreso	Simple monovalorado	Día de ingreso	int	2	No nulo
mes_ingreso	Simple monovalorado	Mes de ingreso	char	3	No nulo
anio_ingreso	Simple monovalorado	Año de ingreso	int	4	No nulo
dia_separacion	Simple monovalorado	Día de separación	int	2	No nulo
mes_separacion	Simple monovalorado	Mes de separación	char	3	No nulo
anio_separacion	Simple monovalorado	Año de separación	int	4	No nulo
motivo_separacion	Simple monovalorado	Motivo de separación	varchar	25	No nulo
num_usuario	Simple monovalorado	Identificador de usuario	int	5	No nulo

### Estudios

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_estudio	Simple monovalorado	Identificador de estudio	Int Autoincrementable	5	No nulo

nombre_esc	Simple monovalorado	Nombre de la escuela	varchar	45	No nulo
periodo	Simple monovalorado	Periodo que asistió	varchar	10	No nulo
titulo_obt	Simple monovalorado	Título obtenido	varchar	45	No nulo
calle	Simple monovalorado	Nombre de la calle	varchar	25	No nulo
num_ext	Simple monovalorado	Numero exterior	int	5	Ninguna
num_int	Simple monovalorado	Numero interior	Int	5	Ninguna
entidad_fed	Simple monovalorado	Entidad federativa	varchar	15	No nulo
estado	Simple monovalorado	Estado	varchar	20	No nulo
num_desempleado	Simple monovalorado	Identificador de desempleado	int	5	No nulo

### Otros cursos

Atributo	Tipo atributo	Descripción	Dom inio	Tamaño	Restricción
id_curso	Simple monovalorado	Identificador de curso	Int Autoincrem-entable	5	No nulo
nombre_cur	Simple monovalorado	Nombre del curso	varchar	35	No nulo
escuela	Simple monovalorado	Nombre de la escuela	varchar	45	No nulo
titulo_obt	Simple monovalorado	Título obtenido	varchar	35	No nulo
descripcion	Simple monovalorado	Descripcion	varchar	30	Ninguna
anio	Simple monovalorado	Año en que estudio	int	4	No nulo
no_usuario	Simple monovalorado	Identificador de desempleado	Int	5	No nub

### Idiomas

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
id_idioma	Simple monovalorado	Identificador de idioma	Int Autoincrem-entable	5	No nulo
idioma	Simple monovalorado	Nombre de idioma	varchar	15	No nulo

### Porcentaje habla

Atributo	Tipo atributo	Descripción	Dominio	Tamaño	Restricción
por_habla	Simple monovalorado	Porcentaje que habla	varchar	5	No nulo
por_lee	Simple monovalorado	Porcentaje que lee	varchar	35	No nulo
por_escribe	Simple	Porcentaje	varchar	45	No nulo

	monovalorado	que escribe			
desempleado_num	Simple monovalorado	Identificador de desempleado	int	35	No nulo
num_idioma	Simple monovalorado	Identificador de idioma	int	30	No nulo

## **CAPITULO 4. IMPLEMENTACIÓN DEL SISTEMA (MODEL)**

### **4.1 Lenguaje de programación**

Para la creación de este sistema se utilizó el lenguaje de programación PHP versión 4.30 ya que este lenguaje corre casi en cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en varias plataformas, incluyendo diferentes versiones de Unix, Windows (95,98,NT,ME,2000,XP). Como en todos los sistemas se utiliza el mismo código base, los scripts pueden ser ejecutados de manera independiente al OS.

La sintaxis de PHP es similar a la del lenguaje C, por esto cualquiera con experiencia en lenguajes del estilo C podrá entender rápidamente PHP. Entre los lenguajes del tipo C incluimos al Java y Javascript, de hecho mucha de la funcionalidad del PHP se la debe al C en funciones como fread () o strlen (), así que muchos programadores estarán familiarizados con el lenguaje PHP .

### **4.2 Manejador de Base de Datos**

En cuanto al manejador de base de datos se utilizó MySql versión 5.0.21, que viene incluido en el servidor Wamp, además de que es una versión libre, es decir no se pagan licencias de uso.

Mysql es uno de los mejores manejadores de bases de datos que existen y se puede utilizar en sistemas basados en Web.

MySql es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.

### **4.3 Herramientas utilizadas**

Como apoyo se utilizaron las siguientes herramientas:

- ✓ Servidor Wamp
- ✓ Reactor Server
- ✓ MySql Front
- ✓ MySql Administrator
- ✓ Sothink DHTML Menu

- ✓ Macromedia DreamWeaver
- ✓ Mozilla Firefox

#### 4.4 Código

Para realizar la conexión con la base de datos se utilizó el siguiente código de PHP.

```
<?php
mysql_connect("localhost", "root","root") or die ("No se Puede Conectar a la
Base de Datos");
mysql_select_db("bolsa_trabajo") or die ("No se Puede seleccionar Base de
Datos");
?>
```

Lo primero que se hace es especificar el nombre del servidor en donde se encuentra la base de datos, enseguida se especifica el nombre de usuario con el cual tratará de conectar y por último se especifica la contraseña.

En la segunda línea se especifica el nombre de la base de datos, en este caso se llama "bolsa\_trabajo", en caso de que no exista la base de datos se manda un mensaje de error.

En el código siguiente se muestra como se insertan los datos en la tabla desempleado, observamos que el primer campo está en NULL, es por que el campo fue definido como auto incrementable y no afecta el que no pongamos un valor específico

```
$query = "insert into desempleado values
(NULL,'$usuario','$pass','$titulo','$nombre','$paterno','$materno','$dia','$mes','$anio
','$edad','$sexo','$lugar','$nacionalidad','$calle','$exterior','$interior','$localizacion','$
colonia','$codigo','$entidad','$estado','$civil','$estatura','$complexion','$seguro','$tel
efono','$celular','$correo','$licencia','$cartilla','$rfc)";
mysql_query($query) or
die (mysql_error());
```

A continuación se muestra el código de una búsqueda realizada por un empleador, dónde se buscará personal de acuerdo con los siguientes parámetros:

titulo, estado donde radican y puesto desempeñado en otras empresas. La consulta sería de la siguiente manera:

```
select nombre, usuario, titulo, estado, puesto from desempleado, empleos_
anteriores where (titulo="secretaria" AND estado ="Durango"
AND puesto="Capturista") AND num_usuario = id_usuario;
```

#### 4.5 Pantallas del sistema

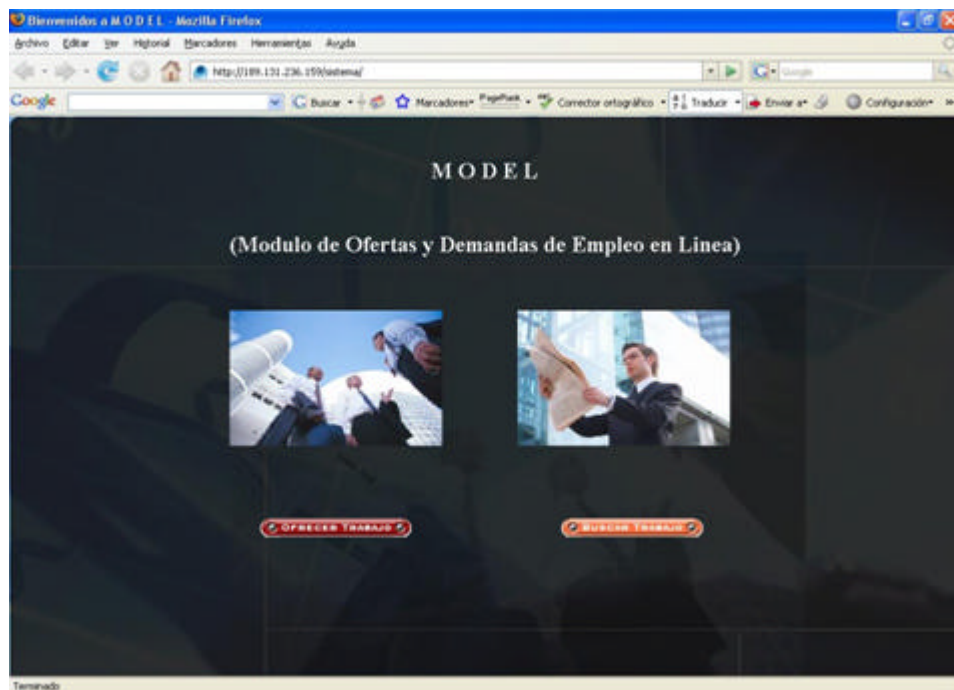


Figura 4.1 Pantalla inicial del sistema Model.

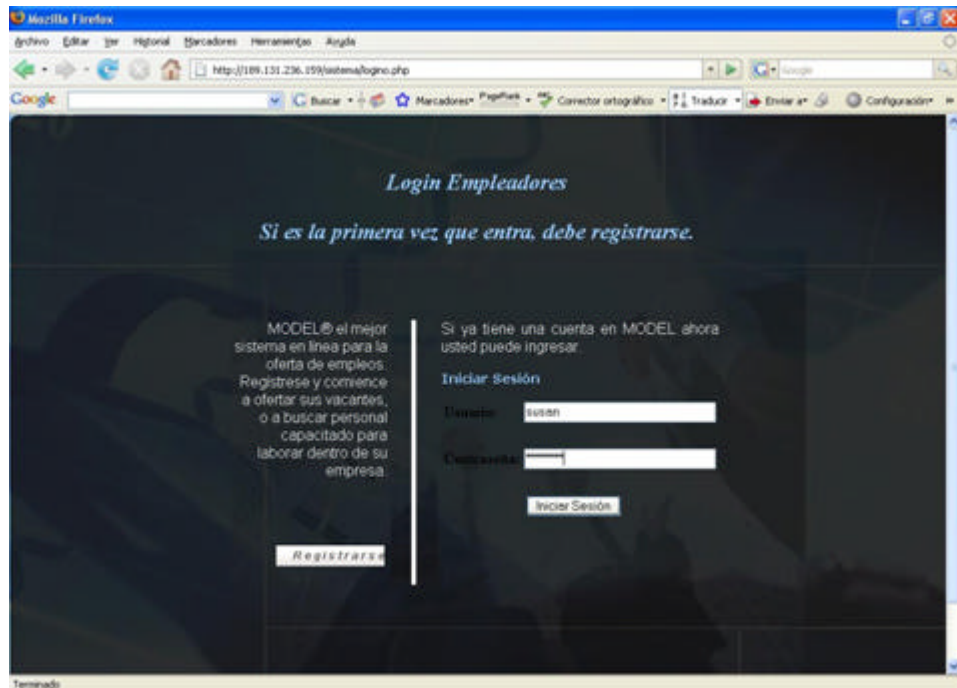


Figura 4.2 Pantalla de login para empleadores.



Figura 4.3 Alta de una vacante.



Figura 4.4 Modificación de Perfil de un Empleador.

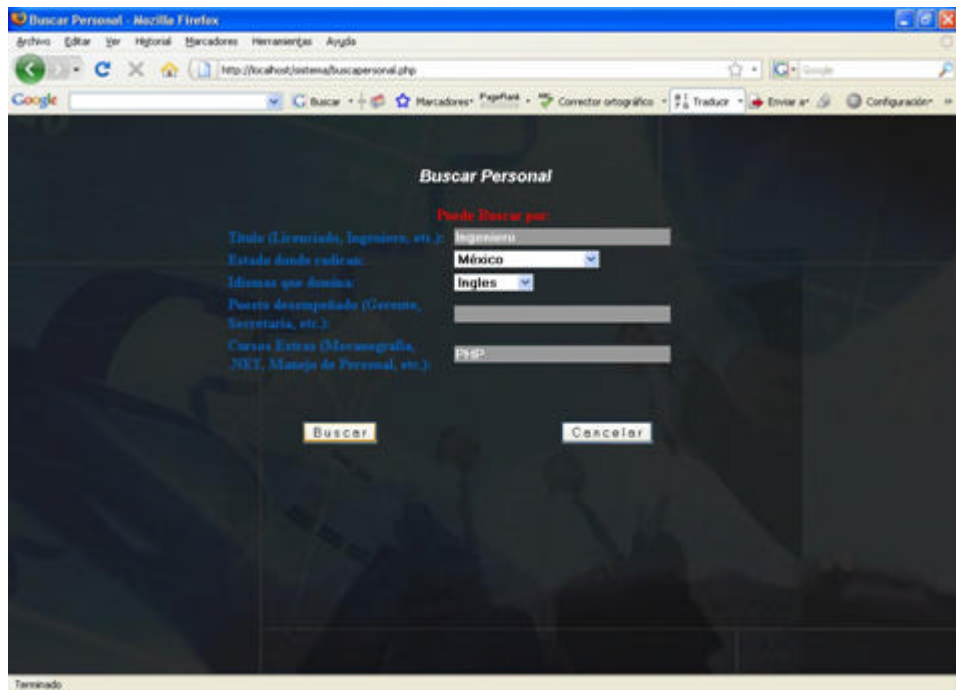


Figura 4.5 Buscar Personal

## **Conclusiones**

El presente trabajo cumple con el objetivo general y con los requerimientos establecidos en la tesis, realiza la búsqueda de empleos en línea siendo este un sistema que utiliza las tecnologías WEB para su implementación.

Se comenzó analizando, diseñando para al final desarrollando el sistema de acuerdo a los objetivos específicos y se trataron de plantear soluciones de tal manera que el sistema sea eficaz y de fácil manejo para los usuarios.

Para el desarrollo como para la ejecución del sistema y las herramientas utilizadas para este tipo de sistemas existen para la plataforma Windows y Unix, así que tal aplicación puede ser desarrollada en alguna de las plataformas soportadas, de igual manera para puesta en marcha del sistema están las dos plataformas mencionadas anteriormente.

La sencillez del lenguaje de programación es una parte muy importante para la creación de este sistema, así será fácil de dar soporte al sistema en caso de ser requerido.

## **Trabajo a futuro**

Algunos de los cambios interesantes que beneficiarían a este sistema son los siguientes:

- ✓ Permitir al usuario (desempleado) imprimir su currículum desde el sistema.
- ✓ Mostrar la opción para enviar correos electrónicos a los usuarios (desempleados) para informarles de nuevos empleos registrados.
- ✓ Difundir el sistema "Módulo de Ofertas y Demandas de Empleo en Línea (MODEL)", para que pueda ser usado en los diferentes estados de la republica.

## Bibliografía

- [1] Cota A. "Ingeniería de Software". Soluciones Avanzadas. Julio de 1994. pp. 5-13.
- [2] URL Proceso Unificado de desarrollo, Septiembre de 2007, [http://es.wikipedia.org/wiki/Proceso\\_Unificado\\_de\\_Rational#Ciclo\\_de\\_vida](http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational#Ciclo_de_vida)
- [3] Roger S. Pressman, "Ingeniería del Software un enfoque práctico", 2002, 5 edición, McGraw-Hill.
- [4] URL Diagrama de casos de uso, Septiembre de 2007, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_casos\\_de\\_uso](http://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso)
- [5] A. U. S. Gustavo Torossi, "Ingeniería de Software", Universidad Tecnológica, Argentina.
- [6] Jeffrey Ullman, "A first course in database systems", Octubre de 2001, Prentice Hall.
- [7] Wilmington, Delaware, "Sistema de Bases de Datos". Conceptos Fundamentales. 1997, 2a. Edición. Addison-Wesley Iberoamericana. (Capítulo 1 y 2).
- [8] Análisis y diseño de bases de datos. Diplomado de bases de datos. M. C. Alma Delia Ambrosio Vázquez. Capítulo 1.
- [9] URL Diccionario de datos, Octubre de 2007, [http://es.wikipedia.org/wiki/Diccionario\\_de\\_datos](http://es.wikipedia.org/wiki/Diccionario_de_datos)
- [10] Tim Converse, "PHP5 and MySQL Bible", 2004, Wiley Publishing, Inc, version electrónica.
- [11] Rafael Martínez, "Manual de PHP", 2001, Grupo de documentación de PHP, versión electrónica.

## Paginas visitadas

- [10]. <http://dev.mysql.com/doc/refman/5.0/es/tutorial.html>
- [11]. <http://www.php.net/>
- [12]. <http://www.mysql.com/>