



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE INGENIERÍA

## PROPUESTA DE DISEÑO DE UN CATÁLOGO DIGITAL DE MOLUSCOS EMPLEANDO UML.

**T E S I S**  
QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN COMPUTACIÓN  
**P R E S E N T A :**  
**VERONICA MARCIAL UNZUETA**

**DIRECTOR DE TESIS:**  
M. en C. RANULFO RODRÍGUEZ SOBREYRA

MÉXICO, D.F. AGOSTO DEL 2006



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## Agradecimientos

A la UNAM y en particular a la Facultad de Ingeniería por la oportunidad de pertenecer a ella y cursar mis estudios profesionales; agradezco a todos los profesores que compartieron sus conocimientos.

Al Instituto de Ciencias del Mar y Limnología, en especial al Laboratorio de Oceanografía Física, sobre todo al M. en C. Ranulfo Rodríguez Sobreyra, por su ayuda e infinita paciencia para concluir este tema y a todas las personas que ahí laboran, en verdad gracias!!!

A todos los integrantes del laboratorio de Malacología por permitirme trabajar con ellos para llevar a cabo este tema, en especial a la Dra. Martha Reguero Reza.

Primero que nada agradezco a Dios que me haya permitido realizar este sueño y compartirlo con mi familia y amistades.

### **Mami:**

Parte fundamental de mi vida, gracias por tener siempre las palabras y caricias exactas para cada momento de mi vida, por enseñarme a ser la mujer que soy, por disfrutar mis logros, regañar mi errores y llorar juntas mis fracasos, siempre como madre y amiga, te amo!!!.

### **Yayo:**

Siempre haz estado a mi lado con tu apoyo incondicional, compartiendo mis éxitos y fracasos, gracias por ser siempre un gran padre, por demostrar que la educación no está peleada con el amor, por ayudarme a lograr una más de mis tantas metas en la vida, eres invaluable yayo!!!!.

### **Gre:**

Gracias por estar a mi lado en gran parte de mi vida, por no dejarme vencer y ofrecerme siempre tu mano y experiencia, por contar siempre con alguien con quien desahogar mi ira y llanto. Gracias!!!! Por las muchas horas que dedicaste a mis estudios, te debo demasiado.

Me diste tantos consejos y regaños, por eso este logro es compartido y va por los dos amorcillo, te amo!!!

### **Gaby:**

Compañera y amiga inseparable de tantas aventuras, gracias por soportar incontables momentos de enojo, eres una gran mujer y amiga a la cual adoro y admiro sobre todas las cosas. Gracias por siempre confiar en mí, y ser mi hermana.

### **Clau y Marifer:**

Gracias por inyectarle a mi vida su interminable alegría y energía, la cual me ayudo a sobrellevar los incontables momentos de tristeza, por recordarme el lado divertido de las cosas y hacerme olvidar por unos momentos las presiones de la vida diaria, las adoro pequeñas!!!!!!

Y a todas aquellas personas que a lo largo de mi vida me han brindado su cariño y amistad incondicional. Gracias muchas gracias!!!

## ÍNDICE

### **INTRODUCCIÓN**

Antecedentes

Objetivo general

Objetivos particulares

### **CAPÍTULO 1 CATÁLOGOS DIGITALES**

Antecedentes

Definición

Características

Usos y Aplicaciones

Ventajas y Desventajas

### **CAPÍTULO 2 FUNDAMENTOS BÁSICOS DE BASE DE DATOS Y DE LA METODOLOGÍA UML**

Introducción

Antecedentes

Conceptos Básicos

Objetivos de una Base de Datos

Arquitectura de Base de Datos

Modelo Relacional

Álgebra Relacional

El Lenguaje SQL

Normalización

Llaves

Reglas de Integridad

Lenguaje de Modelación Unificado (UML)

### **CAPÍTULO 3 DISEÑO DE UN CATÁLOGO DIGITAL**

Diseño del Catálogo Digital

Diseño de la Base de Datos

Etapa de Diseño

Etapa de Implementación

### **CAPÍTULO 4 RESULTADOS**

Etapa de Pruebas

Comentarios

### **CAPÍTULO 5 CONCLUSIONES**

#### **ANEXOS**

ANEXO A Metodología UML

ANEXO B Diagrama de Clases del Catálogo de Moluscos

ANEXO C Modelo Entidad- Relación de la Base de Datos del Catálogo de Moluscos

ANEXO D Tablas de la Base de Datos del Catálogo de Moluscos

ANEXO E Programas en SQL de la Base de Datos del Catálogo Moluscos

#### **BIBLIOGRAFÍA**

## Introducción

### Antecedentes

A lo largo de muchos años el Laboratorio de Malacología del Instituto de Ciencias del Mar y Limnología (ICML) de la UNAM, ha logrado reunir un extenso acervo de especies, como resultado de este trabajo se tiene una colección de moluscos.

Los ejemplares que conforman la colección de moluscos han sido recolectados o donados según sea el caso, por eso es de vital importancia trascender al paso de la tecnología realizando un catálogo digital de dicha colección y poder difundir la gran cantidad de información que se tiene reunida en papel, utilizando las ventajas de las tecnologías de informática.

Los catálogos digitales son la reunión de documentos producidos por una persona física o jurídica haciendo uso de la electrónica, los datos se almacenan y transmiten por medios electrónicos lo cual nos permite llevar o transmitir información a diferentes personas y lugares sin la necesidad de estar en el sitio físico. Hoy en día los catálogos digitales se usan en diferentes áreas de la investigación obteniendo resultados satisfactorios.

En cualquier área de investigación, el usuario accede directamente a los documentos, teniendo la seguridad de que el contenido de los archivos es de valor administrativo, legal, informativo y cultural, para su conservación permanente [Ackeman y Fielding, 1995].

En el Laboratorio de Malacología se pretende a futuro incrementar el acervo a manera de integrar los diversos tipos de ejemplares de todo el mundo ya que sus características son diferentes y el estudio resulta ser más complejo.

### Objetivo General

Definir una propuesta de diseño de un catálogo digital de moluscos, empleando UML.

### Objetivos Particulares

- Estudiar los conceptos básicos de Base de Datos y de la Metodología UML.
- Aplicar la Metodología UML para el diseño del un catálogo digital de moluscos.
- Diseñar una Base de Datos para la organización de la colección de moluscos del Norte de América.
- Presentar la propuesta final del catálogo digital.

Para cumplir los objetivos de la tesis se presenta a continuación una descripción del contenido

### ***Capítulo 1 Catálogos Digitales***

Este capítulo contiene conceptos y definiciones sobre los catálogos digitales, sus usos y aplicaciones, así como las ventajas y desventajas que nos ofrecen la aplicación de estos.

### ***Capítulo 2 Fundamentos Básicos De Base De Datos y De La Metodología UML***

El capítulo 2 aborda conceptos básicos de todo lo relacionado a Base de Datos tales como: Objetivos, Arquitectura, Fases de Diseño, Sistema Manejador, Principales funciones de SDBD y Modelo Jerárquico.

Se integran temas conceptuales relacionados a la Metodología UML para la implementación del sistema, se presenta de una manera más detallada los conceptos de los componentes utilizados por UML en el anexo A.

### ***Capítulo 3 Diseño de un Catálogo Digital***

El capítulo presenta todo lo relacionado con el proceso de diseño del sistema partiendo por la etapa de estrategia y objetivos planteados, tomando en cuenta los conceptos investigados anteriormente sobre el diseño de Bases de Datos. Se presentan los criterios tomados para la elaboración de las entidades y posteriormente el diseño del diagrama de entidad relación con sus respectivas tablas.

Se integra la metodología UML, para el diseño del sistema que contiene los diagramas de casos de uso y diagramas de clase integrando una explicación de cada diagrama.

### ***Capítulo 4 Resultados***

Este capítulo contiene los resultados obtenidos en el proceso de elaboración del diseño del catálogo digital utilizando la metodología UML. Integrando comentarios del proceso de diseño.

### ***Capítulo 5 Conclusiones***

Se presenta un análisis de los resultados obtenidos, para determinar el cumplimiento de los objetivos de la presente tesis y se presentan los trabajos a futuro.

# CAPÍTULO 1 CATÁLOGOS DIGITALES

## 1.1 Antecedentes

Al introducir las tecnologías de información en las bibliotecas han surgido diferentes terminologías que comúnmente son utilizadas de manera indistinta y confusa, a continuación se describen los términos: biblioteca electrónica, biblioteca virtual y biblioteca digital. [Tramullas, 2003]

- **Biblioteca electrónica** es aquella que cuenta con sistemas de automatización que le permiten un ágil y correcta administración de los materiales que resguarda, principalmente en papel. Así mismo, cuenta con sistemas de telecomunicaciones que le permitirán acceder a su información, en formato electrónico, de manera remota o local. Proporciona principalmente catálogos y listas de las colecciones que se encuentran físicamente dentro de un edificio.
- **Biblioteca virtual**, es aquella que hace uso de la realidad para mostrar una interfaz y emular un ambiente que situé al usuario dentro de una biblioteca tradicional. Hace uso de la más alta tecnología multimedia y puede guiar al usuario a través de diferentes sistemas para encontrar colecciones en diferentes sitios, conectados a través de sistemas de cómputo y telecomunicaciones.
- **Biblioteca digital**, es un repositorio de acervos y contenidos digitalizados, almacenados en diferentes formatos electrónicos por lo que el original en papel, en caso de existir, pierde supremacía. Generalmente, son bibliotecas pequeñas y especializadas, con colecciones limitadas a sólo algunos temas.

La definición tomada por la “Digital Libraries Federation” consiste en [Tramullas, 2003]:

“Las Bibliotecas Digitales son organizaciones que proveen los recursos, incluyendo personal especializado, para seleccionar, estructurar, distribuir, controlar el acceso, conservar la integridad y asegurar la persistencia a través del tiempo de colecciones de trabajos digitales que estén fácil y económicamente disponibles para usarse por una comunidad definida o para un conjunto de comunidades”

El catálogo digital nace en el momento en que surgen las publicaciones electrónicas, disponibles tanto por medio de conexiones remotas o bien procesadas en medios electrónicos como CD-ROM.

Los sistemas tradicionales de archivo (catálogos digitales) han evolucionado muy despacio, entre 1850 y 1950 las imágenes se almacenaban en cajas, en archivadores de madera o metálicos y en sobres de papel.

A partir de los años cincuenta, con la aplicación de las bases de datos, por parte de los Estados Unidos, se incorporó la técnica del análisis de contenidos, para obtener la máxima información. Se presentaba, un problema al deteriorar la imagen original en el proceso de impresión, las firmas A.G.E. e Imagen Bank, ambas ubicadas en la ciudad de Madrid, solucionaron el problema con la edición de los primeros catálogos digitales en CD ROM [Ackeman y Fielding, 1995].

## 1.2 Definición

Los catálogos digitales son la reunión de documentos producidos por una persona física o jurídica, haciendo uso de la electrónica, que se almacenan y transmiten por medios electrónicos y asumiendo una organización respetuosa en el momento de su producción, con el fin de garantizar la información



expuesta con su valor informativo, legal y cultural, permitiendo su difusión y acceso [Ackeman y Fielding, 1995].

Una biblioteca digital es un espacio de información digital, que reúne un conjunto de recursos de información, en el marco de un proceso multifacetado de transferencia de información desde la fuente al usuario, proceso que se caracteriza por su valor añadido. El proceso de transferencia se caracteriza por su interactividad, en cuanto debe atender a la interacción con los usuarios potenciales, con los recursos de información, y la mediación entre ambos para generar valor añadido durante el propio proceso de transferencia. La biblioteca digital también es el ciclo de vida de la información, desde la creación. Hasta la diseminación, utilización y obtención de nuevo conocimiento. Colección digital, orientación al usuario y servicios de valor añadido son los tres pilares básicos de la biblioteca digital [Tramullas, 2003].

El término digitalizar debe asociarse con el almacenamiento y recuperación de la imagen, la técnica de digitalización de la imagen es por scanners de cama plana o de tambor o directamente con una cámara fotográfica digital, posteriormente manipular la imagen con los programas correspondientes.

El contenido de los archivos es de valor administrativo, legal, informativo y cultural, para su conservación permanente.

### 1.3 Características

Existen dos tipos de colecciones, las altamente estructuradas y las semiestructuradas. En el caso de las primeras, los contenidos digitales se organizan uniformemente, especificando de antemano los atributos de interés y sus tipos de datos. En el caso de las segundas, comprenden contenidos de tipo y extensión variables, son descritos por algunos metadatos que auxilian en su explotación. Los metadatos son consideradas como etiquetas asignadas a datos.

La colección de documentos digitales, forman un conjunto de documentos organizados y puestos a disposición del usuario. Las colecciones digitales no son físicas, pero si son directamente aprovechadas por los usuarios: para ello, es necesario disponer de un sistema informático que actúe como intermediario entre la colección y el usuario. Sin embargo, y al igual que en otros sistemas de información digital, es el usuario el que establece los parámetros y factores que afectan a su utilización [Ackeman y Fielding, 1995].

El usuario accede directamente a los documentos, sustituyendo el proceso humano por el proceso que proporciona un entorno informático. El origen de estas colecciones puede ser variado, en cuanto pueden provenir de procesos de digitalización de fondos propios, o bien por adquisición, o por, último, por integración de colecciones provistas por terceros. En cualquier caso, no debe pasarse por alto que una manera de recopilación de enlaces a documentos existentes en Internet, o el acceso, más o menos integrado a un conjunto de bases de datos.

Las características de los documentos digitales que forman parte de las colecciones, se pueden encontrar documentos digitales estáticos, que han sido fijados en un formato sobre el cual no cabe realizar modificaciones de estructura, contenido o autoría. Un ejemplo de este tipo lo ofrecen las colecciones digitales de artículos de revista que ofrecen las editoriales, generalmente en formato PDF. Por otro lado, los documentos digitales dinámicos están sujetos a posibles variaciones de estructura, contenido o autoría. Una simple página web, o un documento complejo estructurado en varias unidades, o un documento generado a través de un sistema de gestión de contenidos

fundamentado en bases de datos, pueden variar en cualquier momento, a través de una actualización del mismo [Tramullas, 2003].

El panorama actual de la composición de las colecciones digitales muestran una gran heterogeneidad, ya que están formadas por diferentes tipos de documentos en diferentes tipos de formatos, a lo que hay que unir diferentes interfaces, y la viabilidad técnica del mantenimiento a largo plazo, tanto de la colección en sí, como de las herramientas que facilitan el acceso y consulta de la misma [Ackerman y Fielding, 1995].

### 1.4 Usos y Aplicaciones

Las aplicaciones son muy diversas ya que es una herramienta muy útil para todo tipo de información educativa, empresarial, etc. Permite optimizar la información de manera que la consulta sea rápida y mostrar de una manera muy grafica la información requerida. Últimamente se ha incrementado el uso en colecciones de tesis digitales para su mejor consulta.

Las producciones académicas y científicas representan un área especialmente susceptible de integrar en colecciones digitales útiles y trascendentes.

Los reportes internos de investigación, las publicaciones institucionales de diversos tipos, las memorias de congresos, los materiales de apoyo a cursos, colecciones de especies zootecnistas y conferencias son solo algunas de las áreas en las cuales se ha aplicado la digitalización.

### 1.5 Ventajas y Desventajas

Consideremos algunas ventajas que se presentan con los archivos tradicionales en papel ante la digitalización (CD-ROM) [Braude, 2003]:

- ✓ En cuanto al peso del soporte para la misma cantidad de información, mientras un CD -ROM pesa 19.8 gr. en papel significa unas 4 toneladas.
- ✓ Un CD-ROM almacena 650 megabytes de información, equivalentes a unas 500.000 páginas de texto impresas.
- ✓ El tamaño del papel es variable en tanto que el del CD-ROM es normalizado, 12 cms. de diámetro.
- ✓ La duración del papel es variable, entre 50 y 100 años según la calidad, el CD-ROM puede durar más de cien años según los fabricantes.

Considerando algunas de sus desventajas [Braude ,2003]:

- × Los requerimientos personales para acceder al CD son tener conocimientos del manejo de una PC y del lenguaje SQL de las bases de datos; para acceder al soporte sólo se requiere saber leer.
- × La fragilidad, inestabilidad y vulnerabilidad de la información digital.
- × La exigencia de ambientes especiales para la producción y la conservación de los soportes
- × Incompatibilidad de los equipos tecnológicos.
- × El almacenamiento inadecuado sin clasificación ni ordenación previos y sin sistemas de recuperación
- × Obsolescencia tecnológica.

## CAPÍTULO 2 FUNDAMENTOS BÁSICOS DE BASE DE DATOS Y DEL LENGUAJE DE MODELACIÓN UNIFICADO UML

### 2.1 Introducción

El desarrollo de las bases de datos colectivas es sin duda una de las actividades más importantes en el campo de la informática.

La tasa de crecimiento anual de la capacidad de almacenamiento de las computadoras es mayor que la del tamaño o la potencia de cualquier otro componente en la extensa industria de la informática.

Para el futuro próximo, se prevén dos principales desarrollos tecnológicos para el almacenamiento de datos. Primero, el de un dispositivo electromagnético con mucha mayor capacidad que los discos, aunque de acceso menos rápido. Segundo, el de los almacenes de estado sólido, con tiempo de acceso del orden de microsegundo.

El manejo de grandes cantidades de datos exige el desarrollo de otros recursos sistemáticos, aparte del almacén en sí. Esto es, la transmisión de datos, es decir, la capacidad de acceso a la base de datos desde localidades remotas, y el diálogo entre el hombre y la computadora, el que permite al usuario hacer consultas, buscar en los archivos, modificar los datos almacenados o agregar otros, y resolver problemas aprovechando los datos almacenados [Martín, 1995].

### 2.2 Antecedentes

A finales de la década de 1960, cuando las bases de datos entraron por primera vez en el mercado de software, las bases de datos, nacieron de los sistemas de archivos, guardaban datos por periodos largos y permitían almacenar grandes cantidades de datos. Pero en general no garantizaban que éstos no se perdieran si no estaban respaldados.

Los sistemas no soportaban directamente la función de un lenguaje de consulta que facilite el acceso a los datos de archivos [Ullman, 1999].

Las primeras aplicaciones importantes de los sistemas de administración de bases de datos, se realizaron cuando los datos se componían de muchos elementos pequeños, y se efectuaban un gran número de consultas o modificaciones. A continuación se describen algunas aplicaciones:

- Sistemas de reservación de las líneas aéreas
- Sistemas bancarios
- Registros de las empresas

### 2.3 Conceptos básicos

La base de datos puede definirse como una colección de datos interrelacionados, almacenados en conjunto, sin redundancias perjudiciales o innecesarias; su finalidad es la de servir a una aplicación o más, de la mejor manera posible los datos se almacenan de modo que resulten independientes de los programas que los usan, se emplean métodos bien determinados para incluir datos nuevos y para modificar o extraer los datos almacenados. Dícese que un sistema comprende una colección de bases de datos cuando éstas son totalmente independientes desde el punto de vista estructural [Martín, 1995].

En el lenguaje coloquial, con la expresión base de datos se designa una colección de datos que es administrada por un sistema de administración de *bases de datos*, que se abrevia SMBD (Data Base Management System, sistema de administración de bases de datos o simplemente sistema de bases de datos) [Martín, 1995].

### 2.4 Objetivos de una Base de Datos

Los objetivos de este sistema son [Martín, 1995] [Ullman, 1999]:

- Permitir a los usuarios crear otras bases de datos y poder especificar su estructura lógica de datos.
- Ofrecer a los usuarios el servicio de consultar y modificar los datos, por medio de un lenguaje de consulta.
- Soporte de almacenamiento de grandes cantidades de datos, así mismo protegiéndolos contra accidentes o utilización no autorizada.
- Mantener un control del acceso simultáneo de los usuarios al introducir datos al sistema, para no causar conflictos entre los datos y los usuarios.
- Con el fin de mantener bajo el costo hay que elegir técnicas que minimizan las necesidades totales de almacenamiento.
- Eliminar los valores de datos redundantes.
- El almacenamiento de los datos y los procedimientos de actualización e inserción deben asegurar que el sistema pueda recuperarse de estas contingencias sin daños para los datos. Toda instalación debe de garantizar la integridad de la información que almacena.
- La protección de los datos contra el acceso accidental o intencional por parte de personas no autorizadas y contra su indebida destrucción o alteración. El derecho de los individuos y organismos para determinar por sí mismos cuándo, cómo y en qué medida se permitirá la transmisión a terceros de la información que les concierne.

### 2.5 Arquitectura de Base de Datos

#### 2.5.1 Sistema manejador de sistema de base de datos (SMBD)

El Sistema de Gestión de Bases de Datos (SGBD o SMBD) es el conjunto de programas que permiten la implantación, acceso y, mantenimiento de la base de datos. El SGBD, junto con la base de datos y con los usuarios, constituye el Sistema de Base de Datos [Ullman, 1999]. Ver figura 2.1.

Un SMDB debe cumplir una serie de objetivos para ser lo más rápido, eficiente, polivalente y seguro posible [Parilla, 1997].

Una base de datos tiene distintos tipos de usuarios de los cuales se pueden clasificar en usuarios informáticos y usuarios finales. Como se presenta en la figura 2.1

Los usuarios informáticos están formados por usuarios tales como: diseñadores, administradores, analistas y programadores, son los encargados de diseñar y construir sistemas computacionales con todo el entorno que éste requiere. Los usuarios finales están compuestos por: habituales y esporádicos, son aquellos que realizan consultas, altas, bajas, etc. Es decir, dan uso al sistema implementado.

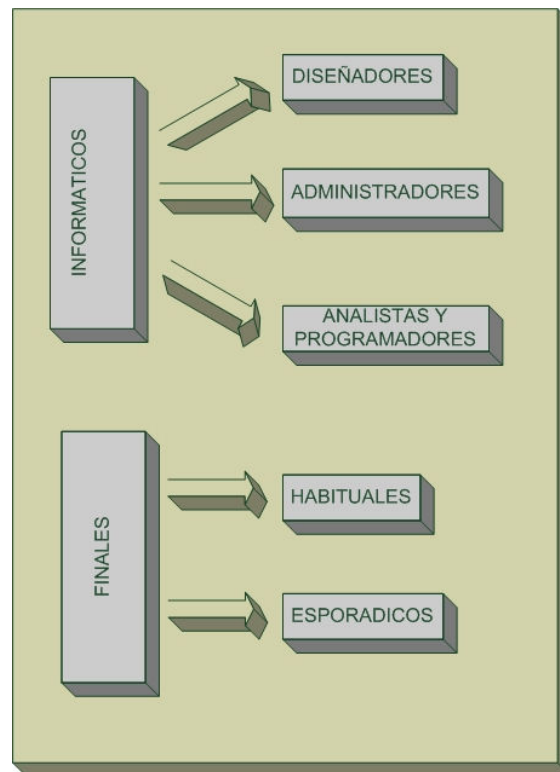


Fig.2.1 Diferentes usuarios

### 2.5.2 Principales funciones de SMBD

En una base de datos existe una gran variedad de usuarios, con necesidades diversas y variables a lo largo del tiempo, los cuales son susceptibles de trabajar simultáneamente con subconjuntos de ésta colección de datos, por lo que es imprescindible dotar al sistema de la adecuada flexibilidad para poder atender las exigencias de todos los usuarios.

Las funciones esenciales de un SMBD se describen a continuación: [De Miguel, 1999]

- **Definición:** Permite al diseñador de la base de datos especificar los elementos de datos que la integran, su estructura y las relaciones que existen entre ellos, las reglas de integridad semántica, de igual manera las características de tipo físico y las vistas lógicas de los usuarios. Por medio del lenguaje de descripción (LDD) de cada SMBD se indica al nivel interno el espacio, la longitud de los campos y su modo de representación. Para nivel externo y conceptual proporciona los instrumentos para la definición de los objetos (entidades, tablas, etc.), así como su identificación, autorizaciones de acceso, restricciones de seguridad, etc.

- **Manipulación:** Una vez creada la estructura de la base, se encuentra lista para su utilización y la manipulación de los usuarios quienes tendrán la necesidad de recuperar su información por medio de la consulta, la cual puede ser de dos formas:
  - Totalidad de los datos: Se recuperan todos los datos de la base o un determinado tipo.
  - Consulta selectiva: Se tienen que localizar los registros con ciertas condiciones que el usuario requiera.

También los usuarios podrán actualizar sus registros almacenados, es decir, poner al día la base de datos, para ello se tiene tres tipos de operaciones:

- Inserción: Agregar nuevos datos a la base
- Borrado: No necesitar el registro o desaparecer algunos elementos.
- Modificación de los datos: Aquellos registros en los cuáles se hayan producido cambios.

La función de manipulación se llevará a cabo por medio de un lenguaje de manipulación de datos (DML).

- **Control:** Reúne todas las interfaces que necesitan los diferentes usuarios para comunicarse con la base y proporciona un conjunto de procedimientos para el administrador. Comúnmente en los SDBD existen funciones de servicios, como obtener estadísticas de utilización, cargar archivos, seguridad física y de protección frente a accesos no autorizados.

La figura 2.2. presenta un resumen de las funciones esenciales de un SDBD.

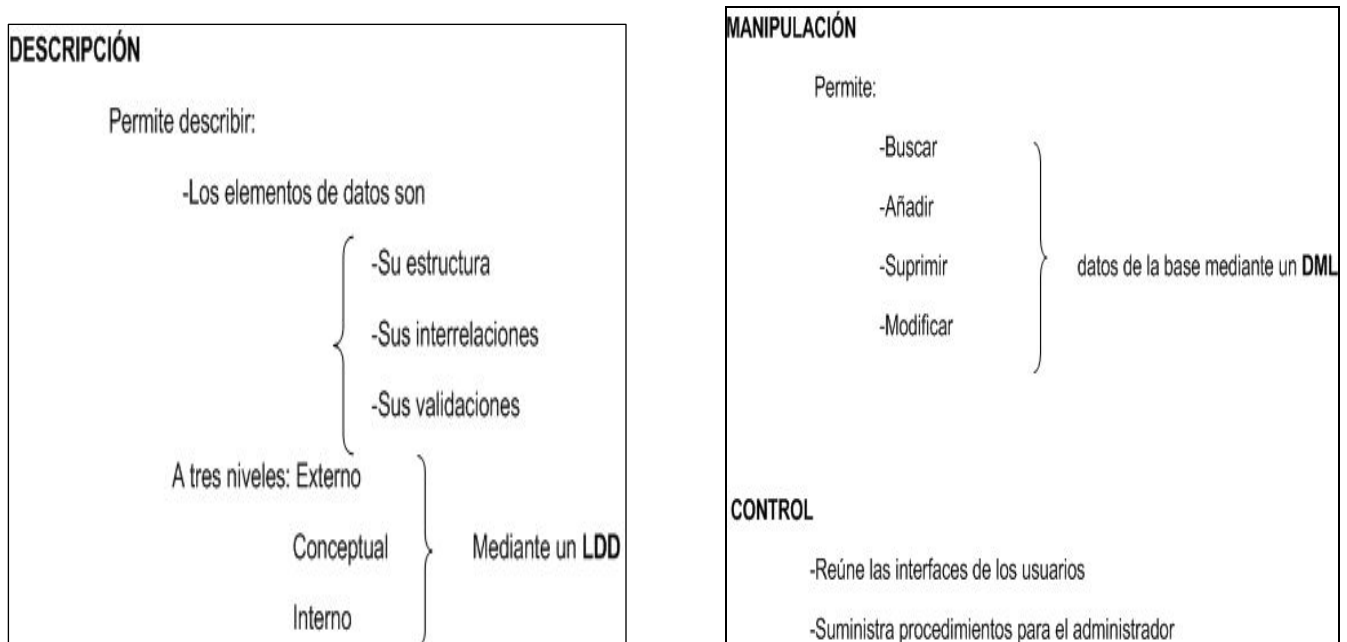


Figura 2.2

### 2.5.3 Fases del diseño de base de datos

El diseño de bases de datos se realiza por lo general en tres fases. La primera fase, llamada de diseño conceptual, produce una representación abstracta y de alto nivel de la realidad. La segunda fase, llamada de diseño lógico, convierte esta representación en especificaciones que pueden implantarse en un sistema de cómputo y ser procesadas por él. La tercera fase, llamada de diseño físico, determina las estructuras de almacenamiento físico y los métodos de consulta requeridos para un acceso eficaz a los contenidos de una base de datos a partir de dispositivos de almacenamiento secundario [Batín,1994].

Actualmente los métodos y modelos de diseño de bases de datos han evolucionado paralelamente con el progreso de la tecnología.

El diseño de una base de datos es un proceso complejo que abarca varias decisiones a muy distintos niveles. La complejidad se controla mejor si se descompone el problema en subproblemas y se resuelve cada uno de estos independientemente, usando métodos y técnicas específicas [Batín,1994].

Las fases del diseño de la base de datos mencionadas anteriormente consisten en lo siguiente:

**A) Diseño Conceptual.** El diseño conceptual parte de la especificación de requerimientos y su resultado es el esquema conceptual de la base de datos. Un esquema conceptual es una descripción de alto nivel de la estructura de la base de datos, independiente del software de SMD que se use para manipularla. El propósito del diseño conceptual es describir el contenido de información de la base de datos, más que las estructuras del almacenamiento que se necesitarán para manejar esta información.

La *abstracción* es un proceso mental que se aplica al seleccionar algunas características y propiedades de un conjunto de objetos y excluir otras no pertinentes.

Se utilizan tres tipos de abstracciones:

- a) *Clasificación.* Se usa para definir el concepto como una clase de objetos de la realidad caracterizados por propiedades comunes.
- b) *Agregación.* Define una nueva clase a partir de un conjunto de (otras) clases que representan sus partes componentes.
- c) *Generalización.* Define una relación de subconjunto entre los elementos de dos (o más) clases.

**B) Diseño lógico:** El diseño lógico parte del esquema conceptual y da como resultado un esquema lógico. El cual es una descripción de la estructura de la base de datos que puede procesar el software de SMD.

Un esquema es una representación de una parte específica de la realidad, creada usando un determinado modelo de datos.

**C) Diseño físico:** El diseño físico parte del esquema lógico y da como resultado un esquema físico, que es una descripción de la implantación de una base de datos en la memoria secundaria; describe las estructuras de almacenamiento y los métodos usados para tener acceso efectivo a los datos [Batín,1994].

### 2.6 Modelo Relacional

El modelo relacional fue propuesto en 1970 por Codd, es un modelo simple, potente y formal para representar la realidad. También ofrece una base firme para enfocar y analizar formalmente muchos problemas relacionados con la gestión de base de datos, como el diseño de la base de datos, la redundancia, la distribución, etc.

El documento de Codd propone un modelo de datos basado en la teoría de las relaciones, en donde los datos se encuentran lógicamente en forma de relaciones -tablas-, siendo un objetivo fundamental del modelo, enfocar las características de un buen diseño [Hawryszkiewycz, 1994].

El Modelo Relacional se define de la siguiente manera:

“En el nivel conceptual, el modelo relacional de datos esta representado por una colección de relaciones almacenadas (también llamadas tablas base). Cada registro de tipo conceptual en un modelo relacional de datos se implanta como un archivo almacenado distinto” [Tsai, 1990].

El documento de Codd perseguía una serie de objetivos que a continuación se presentan [De Miguel, 1999]:

- **Independencia física:** El modelo en que se almacenan los datos no debe influir en su manipulación lógica y, por tanto, los usuarios que acceden a esos datos no han de modificar sus programas por cambios en el almacenamiento físico.
- **Independencia lógica:** Añadir, eliminar y modificar cualquier elemento de la base de datos no debe repercutir en los programas y/o usuarios que están accediendo a subconjuntos parciales de los mismos (vistas).
- **Flexibilidad:** En el sentido de poder ofrecer a cada usuario los datos de forma más adecuada a la correspondiente aplicación.
- **Uniformidad:** Las estructuras lógicas de los datos presentan un aspecto uniforme (tablas), lo que facilita la concepción y manipulación de la base de datos por parte de los usuarios.
- **Sencillez:** Las características anteriores, así como los lenguajes de usuario muy sencillos, producen como resultado que el modelo de datos relacional sea fácil de comprender y de utilizar por parte del usuario final.

Codd para cubrir los objetivos antes mencionados maneja el concepto de relación (tabla) como estructura fundamental del modelo, todos los datos de una base de datos se representan por medio de relaciones cuyo contenido varía en el tiempo. Una relación, en terminología relacional, es un conjunto de filas (tuplas) con unas determinadas características.

Con respecto a la parte dinámica del modelo, se propone un conjunto de operadores que se aplican a las relaciones. Algunos de éstos operadores son clásicos de la teoría de conjuntos. Mientras que

otros fueron introducidos específicamente para el modelo relacional. Todos ellos conforman el álgebra relacional definida formalmente por Codd (1972) [De Miguel, 1999].



### 2.6.1 Estructura del modelo relacional

El elemento básico del modelo es la relación, y un esquema de base de datos **relacional** es una colección de definiciones de relaciones. El esquema de cada relación es una agregación de **atributos**; el conjunto de todos los valores que puede adoptar un atributo en particular se denomina **dominio** de ese atributo.

Un **relación** es una tabla con filas y columnas. Las columnas de las relaciones corresponden a los atributos, las filas, denominadas **tuplas**, son colecciones de valores tomados de cada atributo, y desempeñan la misma función que los casos individuales de entidades en el modelo relacional. El **grado** de una relación es el número de columnas, la **cardinalidad** de una relación es el número de tuplas [Batín, 1994].

La figura 2.3 muestra la tabla llamada "AUTOR", en donde los atributos son: nombre, nacionalidad e institución. Las tuplas son las filas para almacenar los datos, el número de tuplas depende de la cantidad de datos que se almacenen en la tabla, el grado es el total de columnas que integren a la tabla "AUTOR".

AUTOR



Fig.2.3 muestra los conceptos mencionados anteriormente de manera esquemática.

## 2.7 Álgebra Relacional

El álgebra relacional consta de las nueve operaciones: unión, intersección, diferencia, producto cartesiano, selección "join", proyectar, reunión, división y asignación. Las cuatro primeras de estas operaciones se toman de la teoría de conjuntos de las Matemáticas y que son considerablemente parecidas a las operaciones encontradas allí. Esto es razonable, debido a que las relaciones son en sí mismas conjuntos, de ésta manera se le pueden aplicar las operaciones de conjunto. Las cuatro siguientes son operaciones nuevas que se aplican específicamente al modelo de datos relacional. La última operación (asignación) es la operación estándar de los lenguajes de computación de dar un valor a una variable [Hansen, 1998].

Cada operación del Algebra Relacional se explica a continuación:

### 2.7.1 Unión (U)

La operación de unión permite combinar datos de varias relaciones. No siempre es posible realizar consultas de unión entre varias tablas, para poder realizar ésta operación es necesario e imprescindible que las tablas a unir tengan las mismas estructuras y que sus campos sean iguales.

Relación 1  $\cup$  Relación 2 = Relación 3

### 2.7.2 Intersección ( $\cap$ )

La operación de intersección permite identificar filas que son comunes en dos relaciones. El resultado de una operación de intersección es la relación consistente en todas las filas que están en ambas relaciones. Esto significa que, si R3 es la intersección de Relación1 y Relación2.

Relación 1  $\cap$  Relación 2 = Relación 1 - (Relación 1 - Relación 2)

Entonces consta de aquellas filas que están en Relación1 y en Relación2.

### 2.7.3 Diferencia (-)

La operación diferencia permite identificar filas que están en una relación y no en otra. La diferencia entre dos relaciones se define como la relación consistente de todas las filas que están en la primera relación y no están en la segunda relación. (Ver figura 2.4). Así, si

Relación 1 - Relación 2 = Relación3

### 2.7.4 Producto (\*)

La operación producto consiste en la realización de un producto cartesiano entre dos tablas dando como resultado todas las posibles combinaciones entre los registros de la primera y los registros de la segunda. (Ver figura 2.4).

Relación1 x Relación 2 = Relación3

Esta operación se comprende mejor con el siguiente ejemplo:

Relación1		Relación2	
X	Y	W	Z
10	22	33	54
11	25	37	98
		42	100

El producto de Relación1 \* Relación2 daría como resultado la siguiente tabla [Scott,1996]:

Relación3			
10	22	33	54
10	22	37	98
10	22	42	100
11	25	33	54
11	25	37	98
12	25	42	100

### 2.7.5 Selección ( $\sigma$ )

La operación selección consiste en recuperar un conjunto de registros de una tabla o de una relación indicando las condiciones que deben cumplir los registros recuperados, de tal forma que los registros devueltos por la selección han de satisfacer todas las condiciones que se hayan establecido. Ésta operación es la que normalmente se conoce como consulta.

En este tipo de consulta se emplean los operadores de comparación tales como "<" y ">". Hay cinco operadores de comparación: =, <, >, <=, >=. Para cada uno de estos hay un operador correspondiente que usa el operador booleano "not". Así, se tiene "= y "not =", "<" y "not <" y así sucesivamente. También se pueden usar los conectores booleanos "and" y "or". El operador "not" se puede usar para negar una condición entera. (Ver figura 2.4).

$\sigma$  condición\_de\_selección (nombre\_de\_relación)

### 2.7.6 Proyección ( $\Pi$ )

Una proyección es un caso concreto de la operación selección, ésta última devuelve todos los campos de aquellos registros que cumplen la condición que se ha establecido. Una proyección es una acción en la que seleccionamos aquellos campos que deseamos recuperar. (Ver figura 2.4).

$\Pi$  lista\_de\_atributos (nombre\_de\_relación)

### 2.7.7 Reunión

La operación de reunión (join) se usa para conectar datos a través de relaciones. Existen varias versiones: la reunión natural (natural join), la reunión theta (theta join) y la reunión externa (outer join). De éstas, la reunión natural es la más importante

La reunión natural, es la operación de reunión que conecta relaciones cuando las columnas comunes tienen iguales valores.

La reunión theta, es la operación de reunión que conecta relaciones cuando los valores de determinadas columnas tienen una interrelación específica.

La reunión externa, expansión de la reunión natural que incluye todas las filas de ambas relaciones.

### 2.7.8 División

La operación división consiste en crear una nueva relación, seleccionando filas en una relación que se corresponden con todas las filas en otra relación.

$$\text{Relación1: Relación2} = \Pi_A (\text{Relación1}) - \Pi_A [ (\Pi_A(\text{Relación1}) \times \text{Relación2}) - \text{Relación1}]$$

$$A = \{ \text{Atributos Relación1} - \text{Atributos Relación2} \}$$

### 2.7.9 Asignación

La operación asignación asigna un nombre a una relación.

## OPERADORES PRIMITIVOS

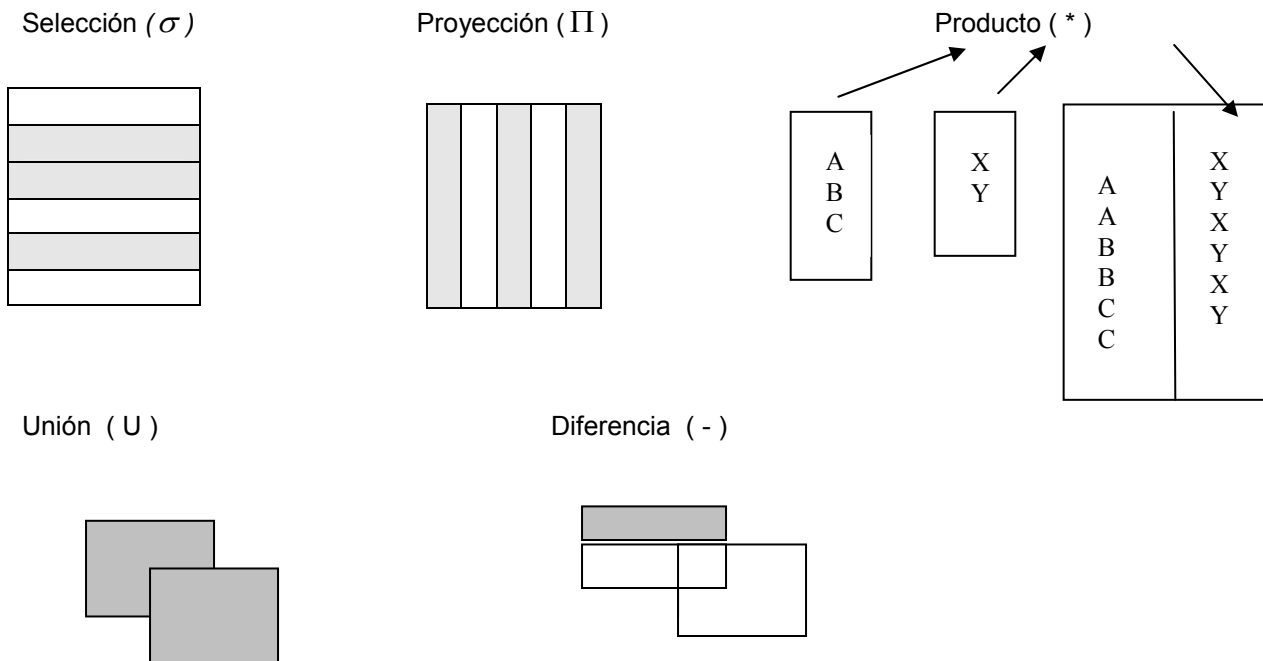


Figura 2.4 Operadores del álgebra relacional

### 2.8 El Lenguaje SQL

#### 2.8.1 Introducción

El SQL ( Structured Query Language), llamado anteriormente SEQUEL, es uno de los lenguajes relacionales más importantes para el manejo de datos basado en los principios del cálculo relacional. IBM desarrolló la implementación prototipo de la versión original de SQL en su laboratorio de San José Research Laboratory California, como un lenguaje de manejo de datos para el SYSTEM R, a principios de 1970.

En 1986, ANSI (American National Standards Institute, Instituto Nacional Americano de Normalización) e ISO (International Standards Organization, Organización Internacional de Normalización) Publicaron una norma de SQL denominada SQL-86. En 1987 IBM publicó su propia norma de SQL denominada SAA-SQL (System Application Architecture Database Interfaz, Interfaz de base de datos para arquitecturas de aplicación de sistemas). En 1989 se publicó una norma extendida para SQL (SQL-89) y actualmente los SGBD son compatibles al menos con esta norma. La norma actual de SQL de ANSI/ISO es la SQL-92. Se debe tener en cuenta que algunas implementaciones de SQL pueden ser compatibles sólo con SQL-89, no siéndolo con SQL-92.

El SQL se puede usar tanto como lenguaje de definición de datos como lenguaje de manejo de datos. Sus proposiciones se pueden emitir interactivamente con una Terminal o pueden estar dentro de algún lenguaje central de programación. Los comandos interactivos se procesan con el procesador interactivo ISQL [Tsai, 1990].

#### 2.8.2 SQL como lenguaje de definición de datos (DDL- Data Definition Language)

Un esquema de bases de datos se representa mediante un sublenguaje especial llamado lenguaje de definición de datos. El resultado de la compilación de estas instrucciones es un conjunto de tablas, relaciones y reglas cuyas definiciones quedan almacenadas en un archivo ( tabla u otro medio de almacenamiento) que contiene “metadatos”, ésto es, datos acerca de datos. Este archivo comúnmente llamado diccionario de datos (o catálogo de sistema) es el que se consulta toda vez que se quiere leer, modificar o eliminar los datos de la base de datos [Tsai, 1990].

#### 2.8.3 SQL como lenguaje de manipulación de datos (DML- Data Manipulation Language).

Se entenderá por manipulación de datos a las acciones siguientes:

- Recuperación de Información.
- Inserción de nueva información.
- Eliminación (Borrado) de información existente.
- Modificación de Información Almacenada

En resumen SQL permite:

- Definir una base de datos mediante tablas.
- Almacenar información en tablas.
- Seleccionar la información que sea necesaria de la base de datos.
- Realizar cambios en la información y estructura de los datos.
- Combinar y calcular datos para conseguir la información necesaria.

## 2.8.4 Sentencias de creación y manipulación de datos en SQL

### 2.8.4.1 Creación de una tabla

El comando **CREATE TABLE** indica al servidor qué nombre queremos poner a la tabla, los nombres de las columnas de la tabla y el tipo de información que cada columna va a contener, cada columna ha de contener el tipo de información que almacenará, de la misma forma se debe indicar la longitud máxima de cualquier dato que pueda ser almacenado en las columnas [De Miguel,1999].

La palabra NOT NULL (NN), significa que cada campo de esa columna debe contener un valor.

```
CREATE TABLE Nombre Tabla (  
Nombre Columna1 Tipo Información (Longitud Máxima Dato),  
Nombre Columna2 Tipo Información (Longitud Máxima Dato) NOT NULL  
  
)  
;
```

### 2.8.4.2 Introducción de la información

Tan pronto como se ha creado la tabla, puede comenzarse a introducir líneas o registros de información mediante el comando **INSERT**.

En éste comando, primero se nombra la tabla de la base de datos en la que quiere insertarse los datos, y a continuación la lista de valores que van a ir en cada columna [De Miguel,1999].

```
INSERT INTO Nombre Tabla  
VALUES (Valor1, Valor2,...,ValorN);
```

### 2.8.4.3 Borrado de datos

La sentencia **DELETE** permite borrar datos de la tabla [De Miguel,1999].

```
DELETE FROM Nombre Tabla  
WHERE Condición Búsqueda;
```

### 2.8.4.4 Recuperación de datos

Una expresión de consulta SQL consta de uno o más bloques de recuperación llamados bloques SELECT-FROM-WHERE. Cada bloque tiene la siguiente estructura [De Miguel,1999]:

```
SELECT columnas (campos)  
FROM tablas (relaciones)  
[WHERE condiciones lógicas]
```

La cláusula **SELECT** especifica una o más columnas objetivo para recuperar, y la cláusula **FROM** especifica las tablas fuente (ya sea de la tabla base o de escenarios) de donde se deben obtener las columnas deseadas. La cláusula opcional **WHERE** especifica las condiciones para elegir renglones.

a) Recuperación de todas las filas de una tabla

```
SELECT *  
FROM Nombre Tabla;
```

b) Combinación de Tablas

La combinación cruzada corresponde al producto cartesiano

```
SELECT *  
FROM Referencia a Tabla CROSS JOIN Referencia a Tabla ;
```

c) Las consultas que no son combinaciones de tablas incluyen, además de tablas simples, la unión (**UNION**), intersección (**INTERSECT**) y diferencia (**EXCEPT**) de tablas. Éste tipo de operaciones requiere que las tablas sean compatibles en dominio. A continuación se tienen las siguientes sentencias:

### **EXCEPT**

```
SELECT *  
FROM Expresión Consulta Combinación  
EXCEPT  
SELECT *  
FROM Expresión Consulta No Combinación;
```

### **UNION CORRESPONDING**

```
SELECT *  
FROM Expresión Consulta Combinación  
UNION CORRESPONDING (Lista De Columnas De Correspondencia)  
SELECT *  
FROM Expresión Consulta Combinación;
```

#### 2.8.4.5 Predicados de la cláusula WHERE

La cláusula **WHERE** sirve para filtrar el resultado, ya que en éste no aparecerán las filas que no cumplan la condición de búsqueda.

El SQL admite distintos tipos de predicados, los más utilizados son los siguientes [De Miguel, 1999]:

##### A) Predicado de **Comparación**

Utiliza los símbolos tradicionales: igual ("**=**"), distinto ("**<>**"), menor que ("**<**"), mayor que ("**>**"), menor o igual a ("**<=**") y mayor o igual a ("**>=**").

B) Predicado **Between**

Es un predicado de comparación especial que permite obtener los valores que se encuentran dentro de un intervalo (extremos inclusive).

C) Predicado **In**

Permite comparar un valor con un alista (de valores) o con una subconsulta.

D) Predicado **Like**

Permite utilizar condiciones a la hora de consultar la base de datos, para lo que se utiliza el carácter “%” para indicar cero o más caracteres en el patrón de búsqueda, y el carácter “\_” para indicar exactamente un carácter.

E) Predicado **Null**

Sirve para determinar si una columna tiene valores nulos.

F) Predicado **UNIQUE**

Sirve para determinar si existen filas duplicadas en la subconsulta que lleva asociado.

G) Predicado de **Comparación Cuantificado**

El lenguaje SQL permite también aplicar cuantificadores existenciales y universales en las consultas por medio de las partículas SOME, ANY y ALL.

2.8.5 Tipos de datos en SQL

Los siguientes tipos de datos son los comúnmente soportados por el estándar SQL.

<b>Datos Numéricos</b>	INTEGER	Entero con signo de 31 bits
	SMALLINT	Entero con signo de 15 bits
	DECIMAL(p,q)	Número con signo de p dígitos, q decimales
	FLOAT(p)	Número de punto flotante, p bits de precisión
<b>Datos String</b>	CHAR(n)	Cadena de texto de largo fijo de n bytes
	VARCHAR(n)	Cadena de texto de largo variante, hasta los n bytes
	GRAPHIC(n)	Cadena de texto de largo fijo, n*2 bytes
	VARGRAPHIC(n)	Cadena de texto de largo variante, hasta los n*2 bytes
<b>Datos de Fecha y Hora</b>	DATE	Fecha (aaaammdd)
	TIME	Hora (hhmmss)
	TIMESTAMP	Combinación de fecha y hora

Tabla 1 - Tipos de datos de SQL



### 2.9 Normalización

#### 2.9.1 Definición

La normalización se deriva del trabajo de Ted Codd, un matemático de Cambridge a principios de los 70's, el notó una serie de formas normales por las cuales debían de pasar los datos para crear un conjunto de tablas relacionales. Académicamente se trabaja con un considerable número de formas normales.

La teoría de la normalización, cuyas tres primeras formas normales fueron introducidas por Codd desde sus primeros trabajos, elimina dependencias entre atributos que originan anomalías en la actualización de la base de datos y proporciona una estructura más regular en la representación de relaciones, constituyendo el soporte para el diseño de bases de datos relacionales.

#### 2.9.2 Razones y aplicaciones de la normalización

La normalización está basada en las matemáticas y es una técnica que se ocupa para:

- Revisar un Modelo Entidad- Relación creado en cualquier forma.
- Crear un Modelo Entidad- Relación desde el inicio.
- Adherir detalles a un Modelo Entidad- Relación.

#### Razones para la Normalización

La técnica permite crear un conjunto completo de datos relevantes y entonces utilizar un grupo de reglas para eliminar redundancia a través de una serie de formas normales. Los términos Tercera Forma Normal (TFN) y Análisis de Datos Relacionales (RDA) son utilizados como sinónimos de normalización.

Al final se tiene un conjunto de grupos de datos que pueden llegar a ser entidades, cada uno con atributos y UIDs, además con llaves foráneas definidas. Se puede usar este grupo de datos para crear un Diagrama Entidad- Relación. El objetivo principal es una mejor comprensión y entendimiento de los requerimientos, para soportar la funcionalidad del área enfocada.

#### 2.9.3 Terminología de la normalización

La normalización es una técnica por separado y como tal, tiene su propia terminología. Cuando se aplique en un modelo Entidad- Relación se pueden utilizar los siguientes términos equivalentes:

##### a) Grupo de Datos

Con cada forma normal, los datos son divididos en grupos de acuerdo con la regla que se haya aplicado. Ninguno de estos son entidades o tablas.

##### b) Grupos Repetidos

Cuando una porción de datos puede tener más de un valor para una sola llave, se le llama repetición. Un grupo repetido es un grupo de datos que puede tener más de un valor.

### c) Dependencia Funcional

Cuando un dato se dice que es dependiente de otro, esto implica que el dato no tiene significado sin el otro, y por lo tanto no puede ser accesado sin el determinante.

### d) Determinante

Es un dato sobre el cual depende otro.

### e) Llave

En la normalización, la llave es el dato que identifica de manera única a un grupo. Si se utiliza normalización en el modelo Entidad- Relación, las llaves y UUIDs son lo mismo.

### f) Llave Foránea

Es un dato dentro de un grupo y es utilizado como llave primaria dentro de otra relación. Si se utiliza normalización en el modelo Entidad- Relación, las llaves foráneas y las relaciones son lo mismo.

### g) Transitivo

Éste término es usado cuando un dato depende de una llave, pero ésta llave depende de otra llave. Ésto pasa cuando un dato no está disponible para la normalización, pero es encontrado después.

## 2.9.4 Reglas de normalización

Al conjunto de reglas se les denomina *Reglas de normalización de relaciones* y a la teoría en la que se basan se le denomina *Teoría de normalización de relaciones* [Luque y Nieto ,2002].

La aplicación de una regla de normalización es una operación que toma una relación como argumento de entrada y da como resultado dos o más relaciones, y:

- La relación objeto de la aplicación de la regla es desestimada en el nuevo esquema relacional considerado.
- No se introducen nuevos atributos en el esquema relacional resultante de la normalización.
- Los atributos de la relación objeto de la normalización pasan a formar parte de la intención de una o más relaciones resultantes.

En la aplicación de la regla de normalización se ha debido eliminar, al menos, una dependencia existente entre los atributos de la relación objeto de la normalización [Luque y Nieto ,2002].

Cuando se inicia la normalización, éstos son los pasos a seguir. Los primeros siete pasos son iterativos y conducidos hacia el documento a ser normalizado. Los últimos cinco pasos son realizados por fuera, con la combinación de resultados [Scott,1996].

- |   |              |
|---|--------------|
| 1. Recolectar los datos en bruto                                  | <b>0NF</b>   |
| 2. Enlistar los datos en bruto con su llave                       |              |
| 3. Remover los grupos repetidos                                   | <b>1NF</b>   |
| 4. Remover dependencias parciales entre los componentes de llaves | <b>2NF</b>   |
| 5. Remover dependencias entre datos                               | <b>3NF</b>   |
| 6. Remover las dependencias entre las llaves                      | <b>BC NF</b> |

7. Realizar pruebas
8. Optimizar
9. Identificar dependencias transitivas
10. Reaplicar pruebas
11. Dibujar el Diagrama Entidad- Relación
12. Utilizar el modelo

### *Etapa 1 y 2.- Recolectando y listando los datos en bruto.*

Recavar datos de cualquier fuente posible, por ejemplo, formas, reportes, pantallas, notas de entrevistas, documentación existente y archivos. Ésto es esencial para comprender los datos que serán normalizados.

Uno de los grandes beneficios de la normalización es el conocimiento pleno de los datos realmente necesitados.

Escribir toda pieza de dato disponible de una fuente específica. No intentar normalizar un sistema completo de una vez. La optimización es el paso que conjunta todos los grupos de datos. Asegurarse de que se utilizan nombres de datos que reflejan la naturaleza de los datos originales. No inventar nombres. Cuestionar a los usuarios, sobre lo que significa exactamente cada término, no suponer [Scott, 1996].

### Seleccionar una llave

- ✓ De la lista de datos que se tiene, seleccionar una llave. Se deben tomar ciertos aspectos al pensar en como escoger una llave.
- ✓ Debe tener un único valor dentro de lo que será normalizado.
- ✓ Que deba ocupar pocos datos, si es posible.
- ✓ Que sea no textual, de ser posible.
- ✓ Si hay más de una candidata, usar la que mejor refleje el documento que se está normalizando.
- ✓ Solo inventar un dato para utilizarlo como llave, si no existen alternativas prácticas.

Al cubrir los puntos anteriores los datos se encuentran en la Forma Normal Cero (0NF)

### *Etapa 3.- Remover los grupos repetidos*

Observar y listar todos los datos que puedan tener más de un valor para cada llave. Éste es un grupo repetido.

### Seleccionar una llave para el Grupo Repetido

Identificar una llave de los datos que se han extraído (o removido). En ocasiones sólo un dato se repite. Copiar la llave original para unirla con la nueva llave seleccionada y marcar la original como llave foránea y la nueva llave como primaria. Ésto preserva la asociación entre el grupo repetido y los datos restantes.

La llave original es copiada no removida, sin embargo, ésta sobra con los datos que no se repiten y puede ser utilizada como llave para éstos datos también.

Notar que un grupo repetido siempre tiene más de un valor en su llave.

### Grupos Repetidos Anidados

Dentro de un grupo repetido, puede haber otros subgrupos repetidos. Tratar estos grupos de la misma forma: remover los datos, seleccionar una llave de los datos removidos y copiar la llave original.

### Enlistar los Datos Restantes

Listar todos los datos restantes y la llave original como un grupo final o relación.

Al cubrir los puntos anteriores los datos se encuentran en la Primera Forma Normal (1NF)

**Primera Forma Normal (1NF):** Establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas [Scott,1996].

### *Etapa 4.- Remover dependencias parciales entre los componentes de llaves .*

Por cada uno de los grupos de datos en 1NF, identificar todos los datos que son dependientes parcialmente de la llave.

### Copiar la llave

Copiar la parte de la llave sobre la cual los datos son dependientes, para unir el grupo y marcarla como llave foránea. Esto preserva la asociación entre grupos.(La llave es copiada y por lo tanto también pertenece con el grupo original).

Enlistar todos los datos restantes y la llave original, como un grupo o relación.

Al cubrir los puntos anteriores los datos se encuentran en la Segunda Forma Normal (2NF)

**Segunda Forma Normal (2NF):** Establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la clave de la tabla para identificarlos [Scott,1996].

### *Etapa 5.- Remover dependencias entre datos.*

Omitir cualquier dato ya marcado como una llave. Enlistar todos los datos que son dependientes de otro y no de la llave.

### Copiar la llave

Copiar los datos de los cuales dependen los otros y marcar la combinación como una llave. Establecerla como una llave foránea en el grupo original. A diferencia de la primera y segunda formas normales.

La llave es copiada y por lo tanto también pertenece en el grupo original. Se repite este proceso para todas las posibles combinaciones de datos.

Enlistar todos los grupos restantes y la llave original, como un grupo o relación.

Al cubrir los puntos anteriores los datos se encuentran en la Tercera Forma Normal (3NF)

**Tercera Forma Normal (3NF):** Señala que hay que eliminar y separar cualquier dato que no sea llave. El valor de esta columna debe depender de la clave. Todos los valores deben de identificarse únicamente por la llave [Scott,1996].

### *Etapa 6.- Remover dependencias entre llaves*

Esta regla es aplicable solo a grupos de datos con una llave formada por varias partes. La forma normal de Bóyce/ Codd es frecuentemente encontrada donde un grupo de datos tiene una llave multipartito, pero no datos. Esta comúnmente dirigida a resolver la relación con una sola llave. También es conocida como la Tercera y Media Forma Normal, y es en realidad una extensión de la Tercera Forma Normal.

La dependencia multievaluada (DMV), es una restricción que garantiza la independencia mutua de atributos multievaluados.

### *Etapa 7.- Utilizando las reglas para probar*

Realizar las siguientes preguntas para saber si se han aplicado las reglas de normalización correctamente:

1. ¿Dependen los datos solamente de su llave completa? Si la respuesta es negativa, entonces no se está en Tercera Forma Normal. Investigar y corregir.
2. ¿Es posible solo un valor para los datos de una llave? Si la respuesta es negativa, entonces no se está completamente en tercera forma. Investigar y corregir.

Una vez que las respuestas anteriores son afirmativas se lleva a cabo la aplicación de la cuarta forma normal y si se considera necesario la quinta forma normal.

**Cuarta Forma Normal (4NF):** Establece que una relación que está en 3FN y no tiene dependencias multievaluadas.

### *Etapa 8.- Optimizar*

Unir grupos de datos donde la llave es idéntica. Optimizar un solo documento es raramente necesario. Esto solo puede pasar como resultado de la Forma Normal de Bóyce/ Codd [Scott,1996].

### *Etapa 9.- Identificando dependencias transitivas*

Una dependencia transitiva se conoce en el análisis y es cuando los datos no dependen realmente de la llave que se eligió, ya que la llave misma depende de algún otro dato que no aparece en el objeto que se está normalizando.

Errores Comunes

Nombre similares en datos de un grupo, puede provocar confusión, por ejemplo:

Un atributo llamado fecha puede tener muchas interpretaciones diferentes en cada grupo, por ejemplo, fecha de nacimiento, fecha de inicio, etc.

Introducir grupos puede provocar dependencias entre datos, donde antes no los había.

Llaves Foráneas

Revisar todas las llaves foráneas. Recordar que una llave foránea es un dato que aparece como llave primaria.

*Etapa 10.- Reaplicar las pruebas*

Al aplicar varias veces las pruebas se asegura que se ha realizado el análisis de datos relacionales exitosamente y que las desiciones tomadas donde se requirieron fueron correctas y que nada cambio como resultado de la optimización.

*Etapa 11.- Dibujar el diagrama entidad- relación*

Una vez que se tienen todos los grupos, se puede iniciar a realizar el Diagrama de Entidad- Relación.

Datos como Atributos

Tomar cada conjunto de datos dentro del grupo y escribirlos dentro de la entidad como atributos. Omitir cualquier dato que este marcado como llave foránea. Esto incluye cualquier llave primaria marcada también como llave foránea.

Llaves Foráneas como Relaciones

Dibujar una relación entre la entidad que contiene la llave foránea y la entidad que contiene la llave primaria. Colocar del lado de la llave foránea una relación de muchos.

Llaves como UIDs

Marcar cada llave primaria como UID en la entidad. Donde la llave foránea sea también llave primaria, marcar la relación como una barra.

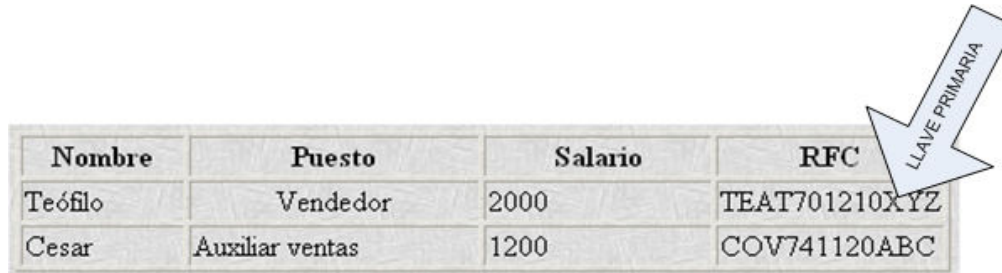
### 2.10 Llaves

En el modelo relacional, una llave de una relación es un conjunto de atributos de la relación que identifica de manera única cada tupla de cada extensión de esa relación. Una relación puede tener más de una llave, y cada llave se denomina llave candidata, dentro de las cuales podemos distinguir las siguientes [De Miguel, 1999]:

- **Llave primaria:** es aquella llave candidata que el usuario escogerá, por consideraciones ajenas al modelo relacional, para identificar las tuplas de la relación. Cuando solo existe una llave candidata, está será la llave primaria, como lo muestra la figura 2.4.
- **Llaves alternativas:** son aquellas llaves candidatas que no han sido escogidas como llave primaria, también se les llama secundarias.

→ **Llaves compuestas:** Es una llave que esta compuesta de más de un atributo.

La siguiente figura muestra los atributos: nombre, puesto, salario y RFC donde el atributo RFC es la llave primaria, ya que es única e irrepitable, los demás atributos pueden ser elegidos como llaves secundarias.



Nombre	Puesto	Salario	RFC
Teófilo	Vendedor	2000	TEAT701210XYZ
Cesar	Auxiliar ventas	1200	COV741120ABC

Figura 2.5

## 2.11 Reglas de Integridad

Las reglas de integridad que pueden ser especificadas en un esquema relacional. Se espera que tales reglas, una vez especificadas se cumplan para cada caso de base de datos de ese esquema, siendo preciso distinguir entre restricciones implícitas y restricciones de usuario [De Miguel, 1999].

### 2.11.1 Restricciones implícitas

Los modelos de datos tienen restricciones que impone el mismo modelo, el cual no admite ciertas estructuras; son las restricciones inherentes, que no son definidas por los usuarios sino obligadas por el propio modelo, lo que quita flexibilidad a la hora de representar el mundo real.

De la definición matemática de relación se deduce una serie de características propias de una relación que se han de cumplir obligatoriamente, ya que diferencian una relación de una tabla; estas características son [De Miguel, 1999]:

- No hay dos tuplas iguales.
- El orden de las tuplas no es significativo.
- El orden de los atributos no es significativo.

La **regla de integridad** también se define por la definición de relación, la cual impone que: "ningún atributo que forme parte de la clave primaria de una relación puede tomar un valor nulo", esto es un valor desconocido o inexistente.

### 2.11.2 Restricciones de usuario

Las restricciones de usuario son facilidades que el modelo ofrece a los usuarios a fin de que estos puedan reflejar en el esquema, lo más fielmente posible, la semántica del mundo real.

Las principales restricciones de usuario del modelo relacional son las siguientes:

**Clave primaria (PRIMARY KEY).** Permite declarar un atributo o un conjunto de atributos como clave primaria de una relación, por lo que sus valores no se podrán repetir ni se admitirán los nulos. La obligatoriedad de la clave primaria es una restricción inherente del

modelo relacional, sin embargo, la declaración de un atributo como clave primaria de una relación es una restricción de usuario que responde a la necesidad del usuario de imponer que los valores del conjunto de atributos que constituyen la clave primaria no se repitan en la relación ni tampoco tomen valores nulos [De Miguel, 1999].

La integridad referencial es una importante restricción de usuario que viene impuesta por el mundo real, siendo el usuario quien la define al describir el esquema relacional, y el modelo la reconoce sin necesidad de que se programe ni de que se tenga que escribir ningún procedimiento para obligar a su cumplimiento.

Hay que determinar las consecuencias que pueden tener ciertas operaciones (borrado y modificación) realizadas sobre tuplas:

- Operación restringida (NO ACTION): El borrado de tuplas de la relación que contiene la clave referenciada(o a la modificación de dicha clave) solo se permite si no existen tuplas con este valor en la relación que contiene la clave ajena.
- Operación con transmisión en cascada (CASCADE): El borrado de tuplas de la relación que contiene la clave candidata referenciada(o a la modificación de dicha clave) lleva consigo el borrado en cascada de tuplas de la relación que contiene la clave ajena.
- Operación con puesta a nulos (SET NULL): El borrado de tuplas de la relación que contiene la clave candidata referenciada(o a la modificación de dicha clave) lleva consigo poner a nulos los valores de las claves ajenas de la relación que referencia.
- Operación con puesta a valor por defecto (SET DEFAULT): El borrado de tuplas de la relación que contiene la clave candidata referenciada(o a la modificación de dicha clave) lleva consigo poner el valor por defecto a la clave ajena de la relación que referencia, valor por defecto que habría sido definido al crear la tabla correspondiente.

La opción seleccionada en caso de borrado es independiente de la modificación, es decir, las opciones de borrado y modificación pueden ser distintas.

### 2.12 Lenguaje de Modelación Unificado (UML)

UML son las siglas en inglés que significan Lenguaje de Modelación Unificado. Fue desarrollado para simplificar y consolidar un número extenso de metodologías orientadas a objetos. En 1996, OMG (Object Management Group) propuso un estándar para las metodologías orientadas a objetos. Booch, Jacobson y Rumbaugh comenzaron a trabajar con las metodologías y eventualmente todo el trabajo terminó por desarrollar UML.

#### 2.12.1 Definición

Este lenguaje nos permite hacer una visualización, construcción y documentar los diagramas que ocupemos en el diseño de software. UML se usa para la construcción, diseño y entendimiento de sistemas (Rumbaugh, et al, 1998).



### 2.12.2 Características

Se introduce el término unificado, porque dentro de este se encierran acepciones importantes, tales como:

- A. A través de la historia de métodos, es decir, la combinación de conceptos de varios métodos de objetos orientados.
- B. El siguiente es que el ciclo de vida de este término se refiere a que los diferentes conceptos y las notaciones, pueden emplearse en diferentes escenarios.
- C. Con la implementación de los lenguajes y plataformas, que incluye diferentes lenguajes de programación, base de datos, a diferencia de lo que encontramos con los otros tipos de modelación, es que UML incluye conceptos semánticos, ciclos de vida, notaciones, etc.
- D. Se cuenta con ambientes estáticos y dinámicos. Cuando se habla del ambiente estático se refiere a cuales son los objetos que se consideran en el diagrama y la relación entre ellos. En el ambiente dinámico se refiere a cual es el tiempo que se llevaran los objetos en realizar las operaciones y cuanto es el tiempo que tardan en finalizar sus tareas en conjunto.
- E. Una de las herramientas importantes es que podemos generar código.
- F. Cuenta con una estructura visual donde se puede hacer la representación de las personas que forman parte del sistema, así como de los objetos. Los objetos representan, cualquier cosa física o conceptual, por ejemplo libros, imágenes, automóviles etc.

### 2.12.3 Descripción general

UML trabaja con modelos por separado, se tienen los diagramas de clase, de colaboración, de secuencia, etc. Cada uno de estos diagramas pertenece a una estructura que al final se colocará en paquetes que ayudan a entender como trabaja el sistema, y al final todos tendrán una relación.

El propósito de UML es que todos los usuarios que modelan en algún lenguaje lo puedan usar (Rumbaugh. Et al, 1998). La intención de esta unificación de los tres tipos de modelación OMT, Booch y Rumbaugh, es que sea un proceso completo de desarrollo. Se puede concluir que la construcción de UML, con todos sus diagramas, reglas y conceptos se agrupo en los siguientes panoramas: estructura estática, comportamiento dinámico, la construcción de la implementación, organización del modelo y por último los mecanismos.

Una de las ventajas que tenemos con UML, es que podemos hacer anotaciones para especificar el porque de ese elemento en el diagrama. Esto se debe a que si hay varios modeladores pueda haber una mejor comunicación en la revisión de los diseños. Anteriormente se planteo un panorama general de lo que contenía UML. Para tener una mejor idea de cómo se compone totalmente UML, dentro del área estructural se tienen siete diagramas elementales que se muestran en la siguiente tabla 2:

**Diagramas**

<p>Diagramas de Casos de Uso</p>	<p>Los diagramas de Casos de Uso son diálogo entre el actor y el sistema, y contienen los siguientes elementos:</p> <ul style="list-style-type: none"> <li>✓ Actores</li> <li>✓ Asociaciones</li> <li>✓ Extensiones</li> <li>✓ Incluyentes</li> <li>✓ Generalización de los uso de caso.</li> </ul>
<p>Diagramas de Clase</p>	<p>Las clases son grupos de objetos con propiedades y comportamientos en común. Los diagramas de clases son creados para representar una vista de todas las clases en el modelo. Las clases con características en común se agrupan en paquetes, que es lo que llamamos bibliotecas, se componen a su vez de:</p> <ul style="list-style-type: none"> <li>✓ Clases</li> <li>✓ Asociaciones</li> <li>✓ Generalización</li> <li>✓ Dependencia</li> <li>✓ Realización</li> <li>✓ Interfaces</li> </ul>
<p>Diagramas Secuencia</p>	<p>Un diagrama de secuencia muestra la interacción de los objetos y las clases envueltas en escenarios y la secuencia de mensajes entre objetos, y se componen de:</p> <ul style="list-style-type: none"> <li>✓ Objetos</li> <li>✓ Líneas de tiempo</li> <li>✓ Fechas (representación del envío de mensajes)</li> </ul>
<p>Diagramas de Estados</p>	<p>Los diagramas de estado coordinan los mensajes que puedan recibir y mandar un objeto. Los estados son una condición durante la vida de un objeto durante la cual se satisfacen algunas condiciones, se desarrolla alguna acción, o se espera por un evento. El estado de un de un objeto se puede caracterizar por el valor de uno o más atributos de una clase. Estos diagramas se componen de:</p> <ul style="list-style-type: none"> <li>✓ Estados de transición</li> <li>✓ Estado inicial</li> <li>✓ Estado final</li> </ul>

<p>Diagramas de Componentes</p>	<p>Un componente es la representación física de un archivo como puede ser en java como archivo.java. Las clases son mapeadas a un paquete quedando esta fusión como un componente, y se representan por:</p> <ul style="list-style-type: none"> <li>✓ Componentes</li> <li>✓ Líneas de unión</li> </ul>
<p>Diagramas de Colaboración</p>	<p>Estos diagramas muestran la interacción de los objetos organizado alrededor de los objetos y la relación entre cada uno, se componen de:</p> <ul style="list-style-type: none"> <li>✓ Objetos (Dibujados en forma de rectángulos)</li> <li>✓ Líneas de conexión</li> <li>✓ Textos (Mensajes)</li> <li>✓ Fechas (Apuntando del que envía el mensaje al que lo recibe)</li> </ul>
<p>Diagrama de Actividades</p>	<p>Estos diagramas presentan la parte dinámica del sistema, los diagramas de flujo muestran las actividades y el control en el sistema, también muestran que actividades se pueden realizar en paralelo, y cualquier camino a través del flujo de actividades. Contiene actividades, transiciones entre las actividades y puntos de decisión, se componen de:</p> <ul style="list-style-type: none"> <li>✓ Actividad</li> <li>✓ Transición</li> <li>✓ Decisión</li> <li>✓ Barras de sincronización</li> </ul>

Tabla 2. Tabla de Diagramas en UML

### CAPÍTULO 3 DISEÑO DE UN CATÁLOGO DIGITAL

#### 3.1 Diseño del Catálogo Digital

##### 3.1.1 Introducción

Emplear la metodología UML (Unified Modeling Language) para el modelado del sistema, permite tener una visión amplia del proyecto a realizar, de una manera estática y dinámica por lo que es muy útil a futuro poder estudiar los diferentes diagramas (clase, casos de uso, etc.), los cuales permiten analizar de diferentes vistas el modelado del sistema, de igual forma permite modificar la parte lógica y física del sistema sin afectarse ambas partes.

Al momento del diseño del sistema UML, nos permite tener un amplio panorama sobre el diseño del funcionamiento entre usuario y sistema, al ser diseñado un catálogo digital, su actividad principal son las consultas y por tal motivo el diagrama de clases ofrece el diseño de las acciones a realizar del sistema con el usuario. Con todas sus restricciones y permisos.

Los lenguajes que soporta la implementación de una modelación en UML son muy variados por citar algunos tenemos: java. C++,C , etc.

Si por algún motivo se necesitase modificar o extender ciertas aplicaciones (componentes) del sistema, UML nos permite hacerlo por medio de estereotipos y restricciones.

##### 3.1.2 Diseño del sistema aplicando metodología UML

Al tener diferentes entrevistas con el personal del laboratorio de Malacología y comprender la forma en que realizaban el proceso para agregar una nueva especie a la colección. Se definió el sistema que desean proponer, por una parte se tiene, la planeación y diseño de la base de datos, y por otra se tiene que definir una propuesta de diseño del sistema que cubra las necesidades de los usuarios.

Las herramientas con las que cuenta UML, permite diseñar parte por parte el sistema y analizar de una manera clara y precisa el funcionamiento del sistema, al momento de la construcción del sistema es útil para el programador tener los esquemas que UML permite trabajar.

##### 3.1.2.1 Objetivo

El propósito de la propuesta del diseño del catálogo digital, es convertir la colección de moluscos físicos a un catálogo digital funcional y de rápida consulta que permita dar a conocer la información de la colección de moluscos con la que cuenta el Laboratorio de Malacología de la UNAM, de una manera optimizada y con las características de las especies, ya que dicho diseño esta basado en las necesidades de los investigadores y sus usuarios más frecuentes, por lo que se pretende cubrir en gran porcentaje los objetivos planteados.

Las características del catálogo permite ser utilizado tanto por especialistas en el área como a usuarios aficionados al tema.

El modelo esta pensado en la fácil comprensión para la etapa de implantación, se proporcionan diagramas con la metodología UML y todo lo relacionado con las actividades del usuario con el sistema, parte fundamental en el momento de la implementación de un sistema.

En un futuro contar con un valioso acervo de información, que permita tener confiabilidad, rapidez en la consulta de un catálogo digital de moluscos, y lo más importante, diversas personas tengan acceso a todo el acervo, sin necesidad de estar en el sitio.

### 3.1.2.2 Situación actual

Hoy en día la inmensa colección de moluscos, se encuentra en el laboratorio antes mencionado organizada por zona geográfica, cada ejemplar cuenta con su respectiva tarjeta de información, en la cual se integra datos específicos del ejemplar y su modo de recolección. No se cuenta con un acervo detallado con cada una de las características del ejemplar, por tal motivo se tuvieron que analizar uno por uno.

Se cuenta con un gran número de moluscos en observación o pre-clasificación, es decir, en frascos de formol, en espera que sean estudiados e integrarlos a la colección.

### 3.1.3 Etapa de diseño

Después de analizar las propuestas por parte de los investigadores, se planteo el desarrollo del sistema separando por partes como lo aplica UML, para una mejor identificación de los componentes y una mejor comprensión del mismo. El uso del lenguaje mencionado nos ayuda a visualizar de una manera muy amplia el diseño del sistema e identificar errores específicos.

#### 3.1.3.1 Casos de uso

Se comenzó con la identificación de los componentes del sistema, entre los cuales tenemos los actores, clases, etc. Una vez identificados los actores para el sistema. El siguiente paso es identificar como interactúa el actor con el sistema.

Los siguientes casos de uso son los procesos en los cuales cada actor (usuario) puede hacer uso en el sistema y lo que representa, en nuestro caso se identifica el actor como un usuario en general.

Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y como se relaciona con su entorno (usuario u otras aplicaciones).

Se representa de una manera global ya que el actor interviene en todo el sistema y sus respectivos procesos, como se muestra en la figura 3.1.

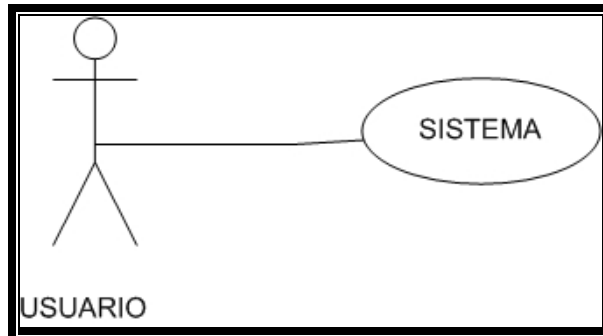


Figura 3.1

El siguiente diagrama de casos de uso, ver figura 3.4, representa la funcionalidad del sistema, a continuación se explica cada uno de los casos de uso que se presentan.

- **Inicio:** Es el ingreso al sistema, debe ser realizado por todos los usuarios, por eso se enlaza con los demás casos de uso.
- **Consultar catálogo digital:** Este proceso indica la consulta al catálogo digital, y todas las opciones que ofrece.
- **Galería:** Contiene un acervo de fotografías, seleccionadas por el administrador de la base de datos.
- **Ingresar como administrador:** Indica el proceso de ingreso a la base de datos, donde solamente el administrador tiene los permisos de borrar, agregar y modificar información de la base de datos. Por tal razón, se solicita una clave para el acceso.
- **Consultar directorio:** Proceso el cual ingresa a un acervo de información sobre los investigadores sobre moluscos y sus datos personales, por si en algún momento el usuario necesita intercambiar información o desea proporcionar información. Los datos que lo integran por mencionar algunos son: nombre del investigador, Institución, especie en estudio, etc.
- **Ver Créditos:** Es una parte donde se habla sobre el Instituto de Ciencias del Mar y Limnología del laboratorio de Malacología, UNAM. Dar a conocer el lugar donde se encuentra la colección de los moluscos físicamente. En esta parte se integran datos sobre el investigador a cargo de la colección Dra. Martha Reguero Reza a modo de sustentar la credibilidad del catálogo de moluscos.
- **Ayuda:** Proceso en el cual se proporciona al usuario información sobre posibles dudas que se le lleguen a presentar en el transcurso del uso del sistema.

Los conectores que se presentan en el Diagrama de Casos de uso, es de tipo PARTICIPACION, ya que permite a los actores relacionarse con los diferentes casos de uso, mostrando si existe algún tipo de interacción entre ellos. Ver figura 3.2

Figura 3.2

El único caso de uso que presenta conector tipo INCURSION CONDICIONADA, es el caso de uso INGRESAR COMO ADMINISTRADOR, ya que se adicionan las acciones del caso de uso que recibe de la relación, pero solo cuando se cumple una condición dado, en este caso es la clave para ingresar a la base de datos, ya que tiene permisos de borrar, modificar, etc., la información almacenada. Ver figura 3.3

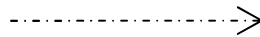


Figura 3.3

En la siguiente figura 3.4 se puede observar de una manera general el diagrama de Casos de Uso del diseño el sistema, se observa que el inicio es el caso de uso que conecta al usuario con el catálogo digital.

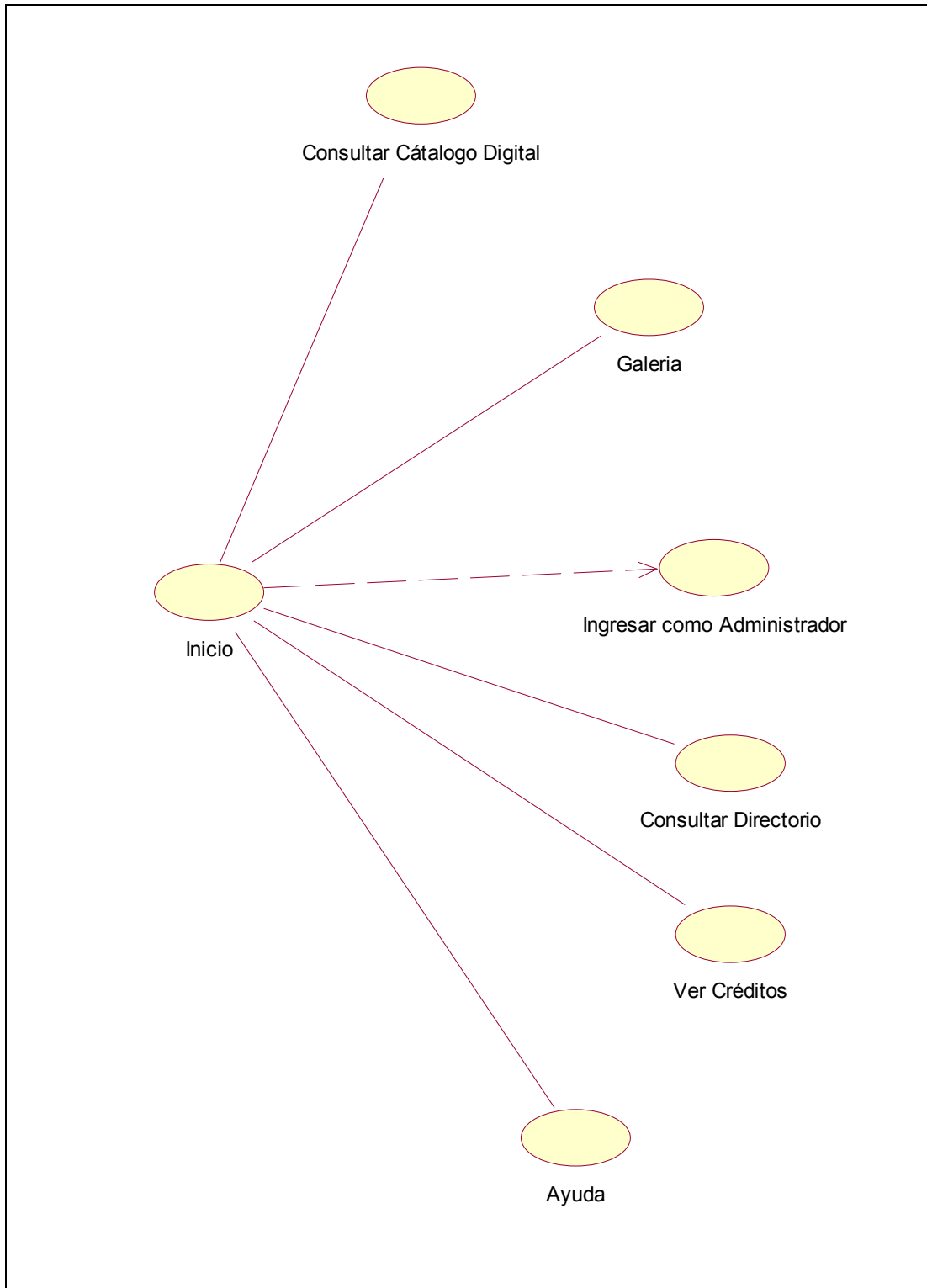


Figura 3.4



### 3.1.3.2 Diagrama de clases

En el momento que se termina con la identificación de los casos de uso el siguiente paso es la definición de las clases, es decir, la descripción de un grupo de objetos que tienen propiedades o características en común.

Clase, es la unidad básica que encapsula a toda la información de un objeto (un objeto es una instancia de una clase).

En la metodología UML, una clase es representada por un rectángulo que posee tres divisiones, las cuales significan lo siguiente:

- ❖ Superior: Contiene el nombre de la clase.
- ❖ Intermedio: Contiene los atributos que caracterizan a la clase (pueden ser privados, públicos o protegidos).
- ❖ Inferior: contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad).

Para un mejor análisis del diagrama de clases se dividió en tres partes, en el anexo B se integra el diagrama completo. En este diagrama de Clases todas las relaciones son de “Asociación”.

El diagrama de clase es de suma importancia ya que nos permite tener una visualización del comportamiento del sistema, a continuación se muestra en la figura 3.5 la primera parte de este.

La clase *Crear Reportes*, contiene un atributo llamado, *Tipo Reporte*, ya que lo caracteriza exclusivamente y es de tipo público, sus operaciones con las cuales interactúa con su entorno son: obtener reporte, mostrar reporte, agregar datos, agregar nuevo y guardar y también son de tipo público. Es decir, no tiene ninguna restricción de visibilidad.

Se agrego la clase *Crear Reporte*, ya que en el sistema se diseño una opción donde los usuarios puedan obtener un reporte final de la consulta realizada, de tal manera que en el transcurso de la consulta se pueda integrar ó eliminar información de acuerdo a las consultas realizadas por el usuario.

Las clases *Imprimir* y *Guardar*, se relacionan con la clase *Crear Reportes*, ya que dentro de dicha clase se puede llevar a cabo otros procesos como imprimir y guardar y cada uno contiene diferentes operaciones a realizar. Ver figura 3.5

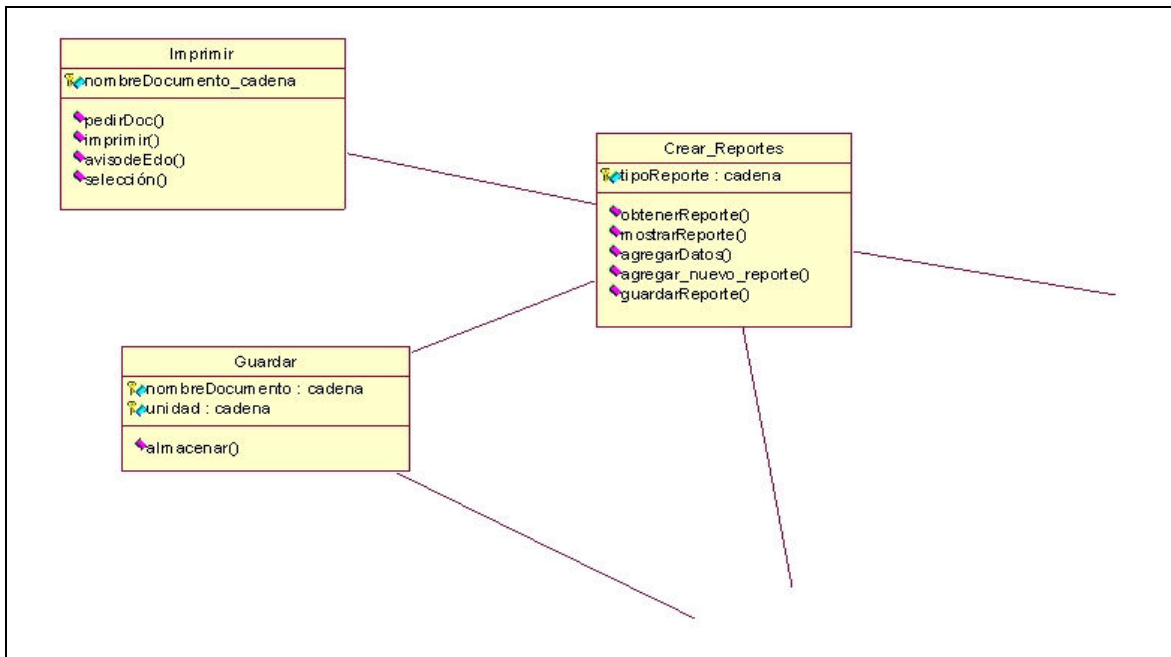


Figura 3.5

En la segunda parte del diagrama, ver figura 3.6, se presenta la clase *Consulta*, se relaciona con diversas clases similares a ella, pero cada una contiene su atributo personal, en la parte de los métodos se puede apreciar que dependiendo de la clase a la que se refiera son sus operaciones, es decir, la clase *Consulta* contiene las operaciones generales tales como: pedir, buscar, mostrar, limpiar, que son las adecuadas en una consulta a un catálogo, pero si la consulta se especifica los atributos cambian, por mencionar un ejemplo, la clase *Consulta Galería* contiene como atributo el nombre de la foto y sus métodos son: mostrar imagen y buscar imagen exclusivamente y de igual forma contiene las operaciones de la clase *Consulta*. Ver figura 3.6

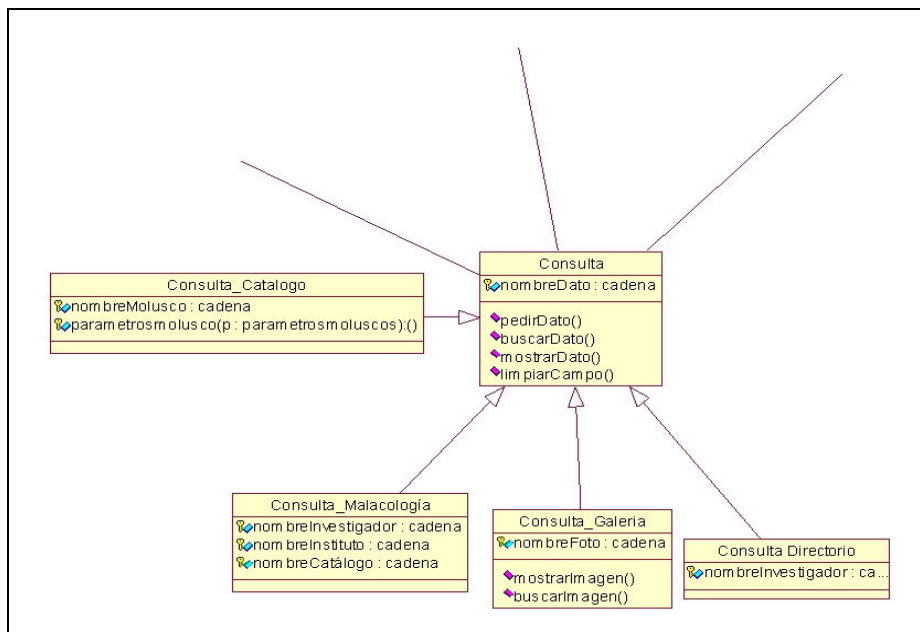


Figura 3.6

Las clases que integran la tercera parte del diagrama, es la clase *Usuario*, la cual carece de atributos por que representa el ingreso al sistema de cualquier usuario, y por ende sus operaciones son: ingresar, salir y regresar inicio. A diferencia de la clase *Usuario Administrador*, donde sus atributos nombre administrador y clave son de carácter privado, ya que se conecta con la clase *Molusco*.

La clase *Molusco* representa el ingreso del administrador al sistema de la base de datos, donde los atributos característicos son nombre científico y nombre común. Sus operaciones son los permisos que tiene el administrador en la base de datos. Sus operaciones son públicas. Ver figura 3.7

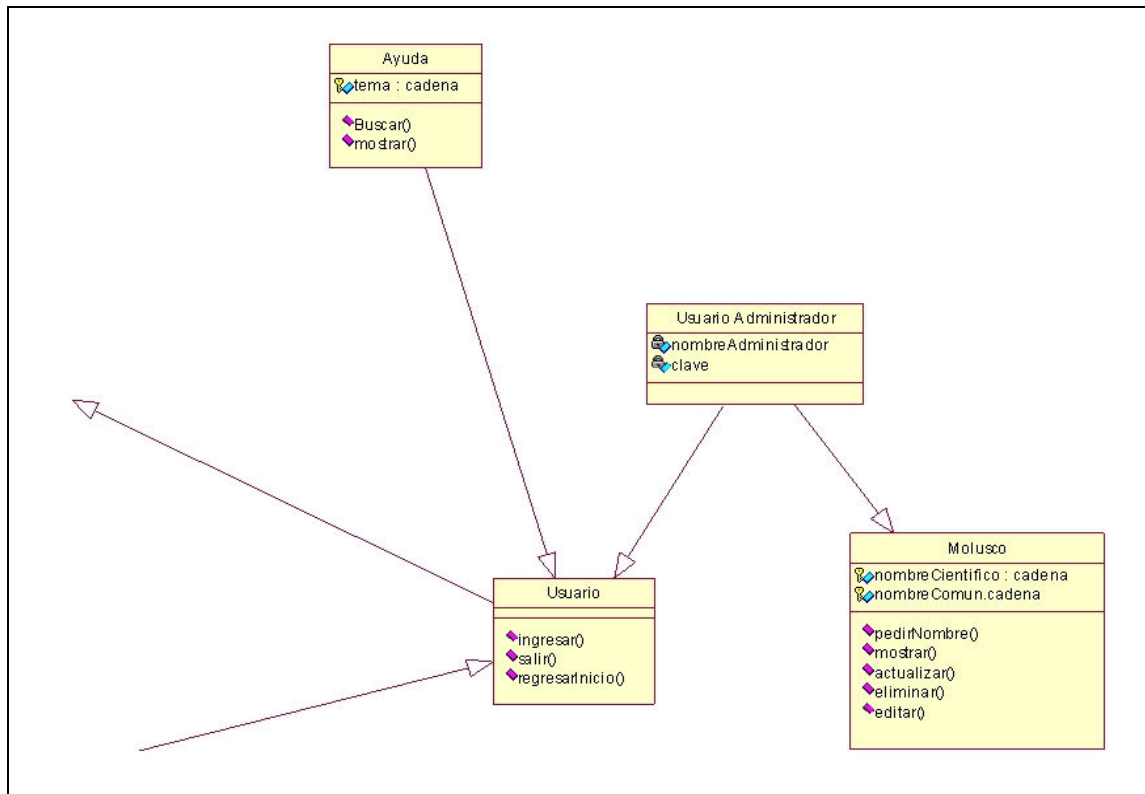


Figura 3.7

## 3.2 Diseño de la Base de Datos

### 3.2.1 Antecedentes

La intensa actividad de investigación que se desarrolla en el Laboratorio de Malacología del Instituto de Ciencias del Mar y Limnología (ICML) de la UNAM, ha logrado reunir a lo largo de varios años un acervo de especies y de las cuales se tiene información importante. Como resultado de este trabajo sean clasificado más de 10 000 especies de las cuales se obtuvieron sus características generales.

La colección esta compuesta de ejemplares que sirven de referencia y apoyo a estudios de investigación a diversos niveles académicos, ya se ha señalado la importancia que tiene consultar y estudiar de primera mano los materiales depositados en una colección de investigación [1985, Argumedo].

Los ejemplares depositados en la colección han sido recolectados en algunos casos de forma circunstancialmente, cuando en el campo han encontrado un molusco, otros como actividad de diferentes proyectos específicos y, para ello se han empleado diversos métodos de colecta. Otra forma de incrementar el acervo ha sido por donación de colecciones completas o de pocos ejemplares.

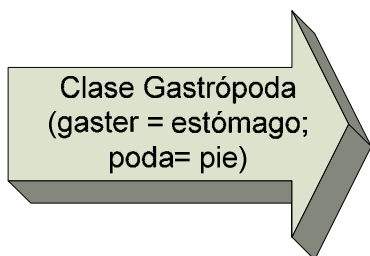
La colección ha sido fuente de consulta para la investigación científica y a partir del estudio de sus ejemplares se han realizado tesis de licenciatura y maestría, además de formación de recursos humanos a través del Servicio Social

Los encargados al recibir las especies, estas vienen acompañadas de una etiqueta con datos de su procedencia. Una vez recibidos los moluscos son estudiados para obtener una clasificación detallada, para posteriormente colocarlos en las cajas de disección e integrarlos a la colección.

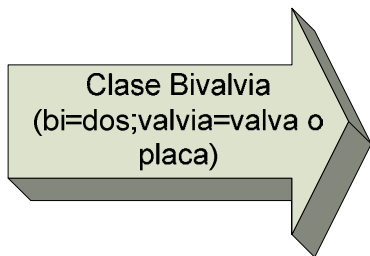
### 3.2.2 Etapa de estrategia

Partiendo de diferentes entrevistas con el personal que labora en el Laboratorio de Malacología; revisando los datos de la colección de moluscos, se obtuvo la siguiente información para la elaboración del catálogo, en base a:

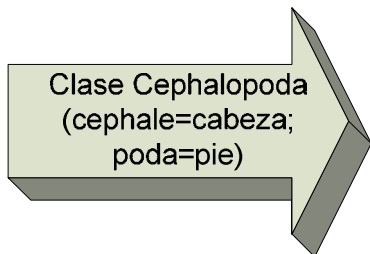
El estudio de los invertebrados, se tiene la siguiente clasificación:



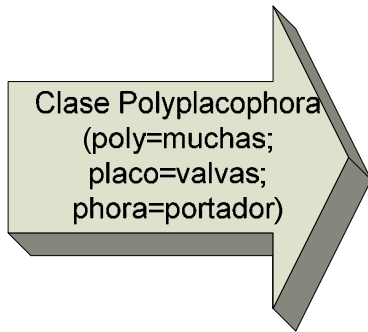
Existen aproximadamente unas 50 mil especies descritas. Se pueden encontrar en casi todo tipo de ambientes (inclusive en desiertos). Se conocen como caracoles, liebres de mar, babosas terrestres y marinas. Se pueden localizar en aguas saladas, dulces y en zonas terrestres.



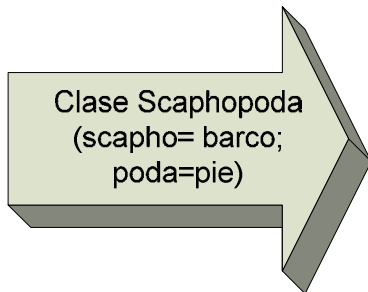
Existen unas ocho mil especies. Entre los bivalvos más conocidos podemos nombrar a las ostras, cholgas, choros, almejas y mejillones. Como su nombre lo indica, el manto produce una concha dividida en dos valvas unidas dorsalmente por un ligamento elástico. El pie se ha modificado para excavar y tiene forma de hacha. Se encuentran en ambientes marinos, salubres y dulceacuícolas.



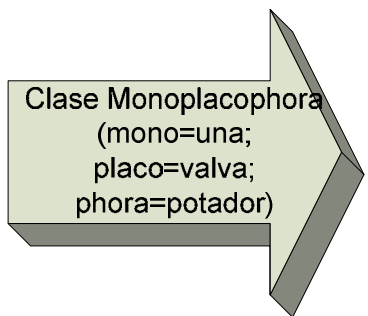
Existen unas 650 especie, comúnmente llamadas pulpos, calamares, jibias y nautilus. La concha esta muy reducida o se ha perdido. Tienen cabeza muy desarrollada, con ojos complejos. El pie se ha modificado en anillo del cual salen entre 8 y 90 brazos o tentáculos.



Existen unas 600 especies de quintotes o cucarachas de mar, carecen de tentáculos y ojos. Su manto produce una serie de siete a ocho placas que le sirven de concha. La mayoría son herbívoros de la zona rocosa pero también se conocen algunas especies de aguas profundas.



Existen unas 350 especies. El manto segrega una concha tubular recta o curva abierta en ambos extremos. Carece de cabeza, tentáculos y ojos. Poseen una serie de filamentos llamados captáculos que les sirven para atraer y capturar presas.



Se conocen unas 11 especies, se trata de un grupo exclusivamente marino en forma de lapa, tienen órganos seriados repetidos: músculos, agallas y nefridios. Tiene una cabeza pequeña pero definida, con tentáculos sólo cerca de la boca, no tiene ojos.

### 3.2.3 Objetivo

El propósito de desarrollar un “catálogo digital” es difundir y sistematizar la información capturada de cada una de las especies que lo conforman. A manera tal que al momento de realizar una consulta de algún molusco sea de manera rápida y precisa, así mismo integrar información al catálogo y poder transportar la colección a otros lugares, para dar la oportunidad a diferentes usuarios.

La base de datos esta pensada para almacenar grandes cantidades de información y realizar búsquedas específicas por los usuarios dependiendo sus requerimientos, tomando en cuenta características tales como: familia, género, profundidad, salinidad, etc.

### 3.3 Etapa de Diseño

Para diseñar es importante definir los requerimientos, y por otra parte definir la estrategia para plantear la solución tecnológica y tratar de cubrir en gran porcentaje las expectativas planteadas.

De las diversas técnicas con las que se cuenta para modelar una base de datos se empleará el modelo de entidad-relación, ya que permite tener independencia física y lógica, por lo que al momento de almacenar, borrar o modificar algún elemento de la base de datos no repercute en el diseño físico. La amplia flexibilidad, uniformidad y sencillez permite el fácil manejo tanto para el usuario y programador.

La base para el estudio de los moluscos es la taxonomía (clasificación de los organismos teórica y prácticamente) trabajo que se realiza principalmente en una colección [Del Campo y Sánchez, 1984,]. Por lo anterior se requiere agregar datos sobre la recolección del ejemplar y por ende ampliar el acervo teórico. Para un investigador es de suma utilidad tomar información del molusco, para realizar estudios específicos.

El estudio de los moluscos se dividió en cuatro partes para su mejor comprensión.

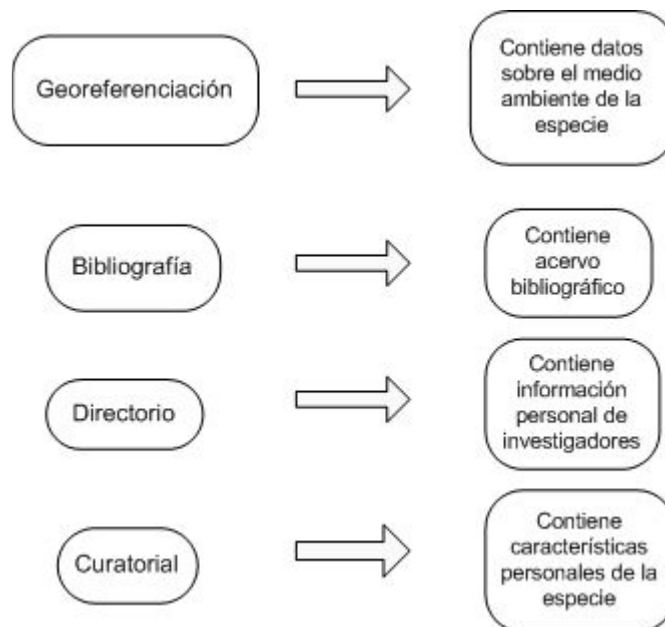


Figura 3.8

La figura 3.8 muestra de una manera resumida las cuatro partes en las que se resumieron las características generales de los moluscos en estudio.

Una vez definidas las características para el análisis de los moluscos en su taxonomía sirven de base para la identificación de entidades de la base de datos, tomando en cuenta ciertos aspectos fundamentales para el buen desarrollo del mismo.

- Todas las entidades deben tener descripción.
- Todas las entidades deben de tener nombre.
- Todas las relaciones muchos a muchos deben de estar resueltas.
- Todas las entidades deben tener un identificador.
- Todos los atributos opcionales deben ser especificados.
- Todos los atributos deben tener un formato.
- El modelo debe estar completamente normalizado.
- Identificación de entidades

El criterio elegido para establecer las entidades fue principalmente los campos de interés para los investigadores en el estudio de los moluscos. Partiendo de las entidades, se definieron los atributos de cada entidad:

- Se utilizaron abreviaciones para evitar confusiones
- Se ocuparon partes de los nombres para formar un nombre final y fuese fácil la identificación.
- Se evito el uso de palabras reservadas de SQL, en los nombres de columnas.
- No se utilizó el mismo nombre de la tabla con sus respectivas columnas.

### 3.3.1 Diseño del modelo

A continuación se describe cada uno de los componentes necesarios para llevar a cabo el modelo Entidad- Relación, tales como: entidades, asociaciones, atributos, diccionario de datos y la construcción de las relaciones entre entidades. Consultar anexo C.

### 3.3.2 Descripción de entidades

<b>Nombre Entidad</b>	<b>Definición Entidad</b>
<b>BOLETÍN</b>	Referencia impresa que pudiese contener un ejemplar, la cual contiene los datos de un boletín tales como: nombre del boletín, autor, fecha de publicación e institución de donde proviene.
<b>CATÁLOGO</b>	Referencia impresa que pudiese contener un ejemplar, la cual contiene los datos de un catálogo tales como: nombre del catálogo y el nombre de la colección a la que pertenece.
<b>CLASE</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>DONADOR</b>	Persona que entrega al Laboratorio de Malacología ejemplares para la colección, registrando los siguientes datos: nombre del donador, nombre de la institución y el número de ejemplares que hayan sido entregados.
<b>ESPECIE</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>FAMILIA</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>FOLLETO</b>	Referencia impresa que pudiese contener un ejemplar, la cual contiene los datos de un folleto tales como: nombre del folleto y el nombre del autor.

## Capítulo 3 Diseño de un Catálogo Digital

---

<b>GÉNERO</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>HÁBITAT</b>	Descripción del ambiente donde habita el molusco, los datos en descripción son: micro hábitat, sustrato, ambiente y tipo del cuerpo de agua el cual contempla los siguientes tipos:  mar, lago, río, laguna, estuario, estero, pantano y marisma.
<b>INVESTIGADOR</b>	Persona la cual tiene en estudio ciertos ejemplares, anexar sus datos para tener el contacto con diferentes investigadores e intercambiar información. Los datos que se registran son los siguientes: nombre del investigador, nombre de la institución a la que pertenece, dirección ya sea particular o de la institución, correo electrónico, especialidad, grado académico y teléfono.
<b>LIBRO</b>	Referencia impresa que pudiese contener un ejemplar, la cual contiene los datos de un libro tales como: título del libro, autor, editorial, lugar, ISBN (International Standard Book Number), ISSN( International Standard Serial Number), los cuales son registros internacionales de cierta bibliografía que se maneja en el área de la Biología, URL(página de Internet que pudiese tener el libro), año de publicación, fecha de consulta, grado académico del libro y observaciones que los investigadores quisieran agregar al haber consultado dicha referencia.
<b>MOLUSCO</b>	Ejemplar en el cual se integran sus características taxonómicas tales como: tamaño, sexo, edad, fotografía, forma de la concha, ornamentado, espinas, postulas, labios, tipo de costilla (se manejan tres tipos: axiales, espirales y reticuladas) , tipo de canal (se manejan dos tipos: sifonal y anal), giro de la concha (se manejan dos tipos: sinistrosa y dextrorsa), ápice, vuelta corporal, número de vueltas y el número de CONABIO (Comisión Nacional para el Conocimiento y Uso de la Biodiversidad).
<b>NOMBRE_TAXONÓMICO</b>	Identifica al molusco por su nombre científico y común.
<b>ORDEN</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>PAÍS</b>	Hace referencia geográfica del ejemplar, integra los siguientes datos: nombre del país, estado o distrito y el nombre de la ciudad.
<b>RECOLECTOR</b>	Persona que recaba los ejemplares en diferentes actividades y posteriormente las integra a la colección, registrando los siguientes datos: nombre del curador, grado del curador, correo electrónico, nombre de la institución, ciudad, teléfono y número de ejemplares.
<b>REF_IMPRESA</b>	Enlace para la aplicación del arco con las diferentes entidades de referencias bibliográficas.
<b>REGIÓN</b>	Datos regionales tales como: nombre de la región, clave de la región, longitud, altitud, latitud, hora, profundidad, salinidad y ambiente terrestre.
<b>SUBCLASE</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>SUBFAMILIA</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>SUBGÉNERO</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>SUBORDEN</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.
<b>SUPERFAMILIA</b>	Nivel de clasificación que se emplea en los moluscos, para la identificación de los mismos.



### 3.3.3 Asociaciones

Al comenzar a analizar las relaciones entre entidades se propusieron los siguientes supuestos y así poder obtener una lista de asociaciones de cada una de las entidades.

- Un libro puede ser solicitado por una o más referencias impresas.
- Una referencia impresa debe ser asignada a uno y solo un libro.
- Un boletín puede ser solicitado por una o más referencias impresas.
- Una referencia impresa debe ser asignada a uno y solo un boletín.
- Un folleto puede ser solicitado por una o más referencias impresas.
- Una referencia impresa debe ser asignada a uno y solo un folleto.
- Un catálogo puede ser solicitado por una o más referencias impresas.
- Una referencia impresa debe ser asignada a uno y solo un folleto.
- Una región debe pertenecer a uno y solo un país.
- Un país puede estar integrado por una o más regiones.
- Un molusco debe destacar uno y solo un nombre taxonómico.
- Un nombre taxonómico debe identificar uno y sólo un molusco.
- Un nombre taxonómico debe estar descrito por una y solo una especie.
- Una especie puede estar formada por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por un y solo un subgénero.
- Un subgénero puede estar formado por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por uno y solo un género.
- Un género puede estar formado por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por una y solo una subfamilia.
- Una subfamilia puede estar formada por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por una y solo una familia.
- Una familia puede estar formada por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por una y solo una superfamilia.
- Una superfamilia puede estar formada por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por uno y solo un suborden.
- Un suborden puede estar formado por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por uno y solo un orden.
- Un orden puede estar formado por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por una y solo una subclase.
- Una subclase puede estar formada por uno o más nombres taxonómicos.
- Un nombre taxonómico debe estar descrito por una y solo una clase.
- Una clase puede estar formada por uno o más nombres taxonómicos.
- Un molusco puede contener muchas referencias impresas.
- Una referencia impresa puede estar integrado por muchos moluscos.
- Un molusco ser inscrito por muchos donadores.
- Un donador puede registrar muchos moluscos
- Un molusco puede ser asignado a muchos hábitats.
- Un hábitat puede estar integrado por muchos moluscos.
- Un molusco puede ser localizado en muchas regiones.
- Una región puede estar integrada por muchos moluscos.
- Un molusco puede ser inscrito por muchos recolectores.
- Un recolector puede ingresar muchos moluscos.
- Un molusco puede ser estudiado por muchos investigadores
- Un investigador puede estar estudiando muchos moluscos.

Al término de obtener la anterior lista de supuestos se llevo acabo la modelación de las relaciones, obteniendo los tres tipos de relaciones: uno ó más (I-M), una y solo una (I-I) y muchos a muchos (N-M).

- LIBRO - REF\_IMPRESA (I-M)
- BOLETÍN - REF\_IMPRESA (I-M)
- FOLLETO - REF\_IMPRESA (I-M)
- CATÁLOGO - REF\_IMPRESA (I-M)
- REGIÓN - PAÍS (I-M)
- MOLUSCO - NOMBRE\_TAXONÓMICO (I-I)
- NOMBRE\_TAXONÓMICO - ESPECIE (I-M)
- NOMBRE\_TAXONÓMICO - SUBGÉNERO (I-M)
- NOMBRE\_TAXONÓMICO - GÉNERO (I-M)
- NOMBRE\_TAXONÓMICO - SUBFAMILIA (I-M)
- NOMBRE\_TAXONÓMICO - FAMILIA (I-M)

## Capítulo 3 Diseño de un Catálogo Digital

---

- NOMBRE\_TAXONOMICO – SUPERFAMILIA (I-M)
- NOMBRE\_TAXONOMICO – SUBORDEN (I-M)
- NOMBRE\_TAXONOMICO – ORDEN (I-M)
- NOMBRE\_TAXONOMICO – SUBCLASE (I-M)
- NOMBRE\_TAXONOMICO – CLASE (I-M)
  
- ✓ MOLUSCO – REF\_IMPRESA (N-M)
- ✓ MOLUSCO- DONADOR (N-M)
- ✓ MOLUSCO- HÁBITAT (N-M)
- ✓ MOLUSCO- REGIÓN (N-M)
- ✓ MOLUSCO- RECOLECTOR (N-M)
- ✓ MOLUSCO- INVESTIGADOR (N-M)

Al momento de la modelación, para eliminar las asociaciones muchos a muchos creamos las entidades siguientes:

- MOLUSCO-DET\_REF(1-M)
- REF\_IMPRESA-DET\_REF(1-M)
- MOLUSCO-DET\_DONADOR (1-M)
- DONADOR-DET\_DONADOR (1-M)
- MOLUSCO-DET\_HÁBITAT (1-M)
- HÁBITAT-DET\_HÁBITAT (1-M)
- MOLUSCO-DET\_RGIÓN (1-M)
- REGIÓN-DET\_REGION (1-M)
- MOLUSCO-DET\_RECOLECTOR (1-M)
- RECOLECTOR-DET\_RECOLECTOR (1-M)
- MOLUSCO-DET\_INVESTIGADOR (1-M)
- INVESTIGADOR-DE\_INVESTIGADOR (1-M)

### 3.3.4 Definición de atributos

Uso Entidad	Nombre atributo	Definición Atributo
<b>BOLETÍN</b>	Nmbre_boletín	Título del boletín
<b>BOLETÍN</b>	Autor_boletín	Autor boletín
<b>BOLETÍN</b>	Fecha_pub	Fecha de publicación
<b>BOLETÍN</b>	Inst_boletín	Institución de procedencia
<b>CATÁLOGO</b>	Nmbre_catálogo	Título del catálogo
<b>CATÁLOGO</b>	Nmbre_colec	Nombre de la colección
<b>CLASE</b>	Nmbre_clase	Nombre del nivel de clasificación clase
<b>DET_DNADOR</b>	Comntario_dnador	Comentarios sobre el donador
<b>DET_HÁBITAT</b>	Comntario_hábitat	Comentarios sobre el hábitat
<b>DET_RECOLEC</b>	Comntario_recolec	Comentarios sobre el recolector
<b>DET_REF</b>	Comntario_ref	Comentarios sobre la referencia
<b>DET_REGION</b>	Comntario_region	Comentarios sobre la región
<b>DET_INV</b>	Comntario_inv	Comentarios sobre el investigador
<b>DONADOR</b>	Nmbre_dnador	Nombre del donador
<b>DONADOR</b>	Inst_dnadora	Institución donadora
<b>DONADOR</b>	Num_ejemplres	Número de ejemplares

## Capítulo 3 Diseño de un Catálogo Digital

---

<b>ESPECIE</b>	Nmbre_especie	Nombre del nivel de clasificación especie
<b>ESPECIE</b>	Dscubrdr_especie	Descubridor de la especie
<b>ESPECIE</b>	Fcha_especie	Fecha de la especie en la cual fue descubierta
<b>FAMILIA</b>	Nmbre_fam	Nombre familia
<b>FOLLETO</b>	Nmbre_folleto	Nombre folleto
<b>FOLLETO</b>	Autor_folleto	Autor del folleto
<b>GÉNERO</b>	Nmbre_genero	Nombre del género
<b>GÉNERO</b>	Dscubrdr_género	Descubridor del género
<b>GÉNERO</b>	Fcha_género	Fecha del género en el cual fue descubierto
<b>HÁBITAT</b>	Tipo_cuerpo_agua	Los diferentes tipos de cuerpo de agua
<b>HÁBITAT</b>	Microhábitat	Microhábitat en el que vive el molusco
<b>HÁBITAT</b>	Sustrato	Sustrato
<b>HÁBITAT</b>	Ambiente	Ambiente
<b>INVESTIGADOR</b>	Nmbre_inves	Nombre del investigador el cual estudia cierto ejemplar
<b>INVESTIGADOR</b>	Nmbre_institción	Nombre de la institución de donde procede el investigador
<b>INVESTIGADOR</b>	Drección	Dirección particular del investigador o de la institución
<b>INVESTIGADOR</b>	C_elect	Correo electrónico
<b>INVESTIGADOR</b>	Grdo_académico	Grado académico del investigador
<b>INVESTIGADOR</b>	Tel	Teléfono particular del investigador o de la institución
<b>LIBRO</b>	Título	Título del libro
<b>LIBRO</b>	Autor_libro	Autor del libro
<b>LIBRO</b>	Editorial	Editorial del libro
<b>LIBRO</b>	Lugar	Lugar de impresión
<b>LIBRO</b>	ISBN	International Standard Book Number
<b>LIBRO</b>	ISSN	International Standard Serial Number
<b>LIBRO</b>	Año_pub	Año de publicación
<b>LIBRO</b>	Fecha_consulta	Fecha de consulta
<b>LIBRO</b>	URL	Página de Internet del libro
<b>LIBRO</b>	Grado_lib	Grado del libro
<b>LIBRO</b>	obsrvacions	Observaciones
<b>MOLUSCO</b>	Tamaño	Tamaño que presenta el ejemplar
<b>MOLUSCO</b>	Sexo	Sexo que presenta el ejemplar, ya sea masculino o femenino
<b>MOLUSCO</b>	Edad	Edad que presenta el ejemplar
<b>MOLUSCO</b>	Fotografía	Imagen del ejemplar (si la hubiese)

## Capítulo 3 Diseño de un Catálogo Digital

<b>MOLUSCO</b>	Firma_concha	Forma de la concha
<b>MOLUSCO</b>	Ornamentado	Tipo de ornamenta
<b>MOLUSCO</b>	Espinas	Descripción de espinas
<b>MOLUSCO</b>	Postulas	Descripción de postulas
<b>MOLUSCO</b>	Labios	Descripción de labios
<b>MOLUSCO</b>	Tipo_costilla	Selección de costillas
<b>MOLUSCO</b>	Tipo_canal	Selección de canal
<b>MOLUSCO</b>	Giro_concha	Descripción de la concha
<b>MOLUSCO</b>	Ápice	Descripción del ápice
<b>MOLUSCO</b>	Vuelta_corporal	Descripción de las vueltas corporales que muestre la concha
<b>MOLUSCO</b>	Num_vueltas	El número de vueltas visibles que presente el molusco
<b>MOLUSCO</b>	Num_conabio	Número otorgado por la Comisión Nacional para el Conocimiento y Uso de la Biodiversidad
<b>NOMBRE_TAXONÓMICO</b>	Nmbre_común	Nombre común del molusco
<b>NOMBRE_TAXONÓMICO</b>	Nmbre_cientfco	Nombre científico del molusco
<b>ORDEN</b>	Nmbre_orden	Nombre del nivel de clasificación orden
<b>PAÍS</b>	Nmbre_país	Nombre del país donde habita o se encuentra el molusco
<b>PAÍS</b>	estadodistrito	Estado o distrito según le corresponda al país
<b>PAÍS</b>	Ciudad	Ciudad según le corresponda al país
<b>RECOLECTOR</b>	Nmbre_curador	Nombre de la persona quien recolecta el ejemplar
<b>RECOLECTOR</b>	Grado_curador	Grado académico del recolector
<b>RECOLECTOR</b>	Correo_e	Correo electrónico del recolector
<b>RECOLECTOR</b>	Nmbre_ins	Nombre de la institución a la que pertenece el recolector
<b>RECOLECTOR</b>	Cd_inst	Ciudad donde se encuentra la institución
<b>RECOLECTOR</b>	Teléfono	Teléfono de la institución
<b>RECOLECTOR</b>	Num_ejemplares	Número de ejemplares recolectados
<b>REF_IMPRESA</b>	Coment_ri	Comentarios acerca de la referencia impresa
<b>REGIÓN</b>	Nmbre_región	Nombre de la región del ejemplar
<b>REGIÓN</b>	Clave_región	Modo de identificar a las regiones geográficamente por medio de claves
<b>REGIÓN</b>	Longitud	Dimensión de un área
<b>REGIÓN</b>	Latitud	Localidad considerada respecto a su distancia con el ecuador
<b>REGIÓN</b>	Altitud	Distancia vertical de un punto de la superficie terrestre respecto al nivel del mar

## Capítulo 3 Diseño de un Catálogo Digital

<b>REGIÓN</b>	Hora	Hora en la cual fue recolectado el ejemplar
<b>REGIÓN</b>	Profundidad	Profundidad que presenta la región
<b>REGIÓN</b>	Salinidad	Grado de salinidad que presenta la región
<b>REGIÓN</b>	Ambiente_terrestre	Descripción del ambiente terrestre donde se encuentra el molusco
<b>SUBCLASE</b>	Sub_clase	Nombre del nivel de clasificación subclase
<b>SUBFAMILIA</b>	Sub_familia	Nombre del nivel de clasificación familia
<b>SUBGÉNERO</b>	Sub_género	Nombre del nivel de clasificación subgénero
<b>SUBORDEN</b>	Sub_orden	Nombre del nivel de clasificación suborden
<b>SUPERFAMILIA</b>	Super_familia	Nombre del nivel de clasificación superfamilia

### 3.3.5 Diccionario de datos

Nombre Entidad	Nombre Atributo	Nombre Columna
<b>BOLETÍN</b>	Nmbre_boletín	NMBRE_BOLETIN VARCHAR2(50) NOT NULL
<b>BOLETÍN</b>	Autor_boletín	AUTOR_BOLETÍN VARCHAR2(50) NOT NULL
<b>BOLETÍN</b>	Fecha_pub	FECHA_PUB DATE
<b>BOLETÍN</b>	Inst_boletín	INST_BOLETÍN VARCHAR2(80) NULL
<b>CATÁLOGO</b>	Nmbre_catálogo	NMBRE_CATALOGO VARCHAR2(70) NOT NULL
<b>CATÁLOGO</b>	Nmbre_colec	NMBRE_COLEC VARCHAR(70) NOT NULL
<b>CLASE</b>	Nmbre_clase	NMBRE_CLASE VARCHAR2(20) NOT NULL
<b>DET_DNADOR</b>	Comntario_dnador	COMNTARIO_DNADOR VARCHAR2(100) NULL
<b>DET_HÁBITAT</b>	Comntario_hábitat	COMNTARIO_HÁBITAT VARCHAR2(100) NULL
<b>DET_RECOLEC</b>	Comntario_recolec	COMNTARIO_RECOLEC VARCHAR2(100) NULL
<b>DET__REF</b>	Comntario_ref	COMNTARIO_REF VARCHAR2(100) NULL
<b>DET_REGIÓN</b>	Comntario_región	COMNTARIO_REGIÓN VARCHAR2(100) NULL
<b>DET_INV</b>	Comntario_inv	COMNTARIO_INV VARCHAR2(100) NULL
<b>DONADOR</b>	Nmbre_dnador	NMBRE_DNADOR VARCHAR2(60) NOT NULL
<b>DONADOR</b>	Inst_dnadora	INST_DNADORA VARCHAR(80) NOT NULL
<b>DONADOR</b>	Num_ejemplres	NUM_EJEMPLRES INTEGER
<b>ESPECIE</b>	Nmbre_especie	NMBRE_ESPECIE VARCHAR2(40) NOT NULL
<b>ESPECIE</b>	Dscubrdor_especie	DSCUBRDOR_ESPECIE VARCHAR2(50) NOT NULL
<b>ESPECIE</b>	Fcha_especie	FCHA_ESPECIE DATE
<b>FAMILIA</b>	Nmbre_fam	NMBRE_FAMILIA VARCHAR2(20) NOT NULL

## Capítulo 3 Diseño de un Catálogo Digital

<b>FOLLETO</b>	Nmbre_folleto	NMBRE_FOLLETO VARCHAR2(50) NOT NULL
<b>FOLLETO</b>	Autor_folleto	AUTOR_FOLLETO VARCHAR2(50) NOT NULL
<b>GÉNERO</b>	Nmbre_genero	NMBRE_GENERO VARCHAR2(15) NOT NULL
<b>GÉNERO</b>	Dscubrdor_género	DSCUBRDOR_GÉNERO VARCHAR2(50) NOT NULL
<b>GÉNERO</b>	Fcha_género	FCHA_GÉNERO DATE
<b>HÁBITAT</b>	Tipo_cuerpo_agua	TIPO_CUERPO_AGUA VARCHAR2(256) NOT NULL
<b>HÁBITAT</b>	Microhábitat	MICROHABITAT VARCHAR2(256) NOT NULL
<b>HÁBITAT</b>	Sustrato	SUSTRATO VARCHAR2(256) NOT NULL
<b>HÁBITAT</b>	Ambiente	AMBIENTE VARCHAR2(256) NOT NULL
<b>INVESTIGADOR</b>	Nmbre_inves	NMBRE_INVES VARCHAR2(50) NOT NULL
<b>INVESTIGADOR</b>	Nmbre_institución	NMBRE_INSTITUCIÓN VARCHAR2(80) NOT NULL
<b>INVESTIGADOR</b>	Drección	DRECCIÓN VARCHAR2 (150) NOT NULL
<b>INVESTIGADOR</b>	C_elect	C_ELECT VARCHAR2(30) NOT NULL
<b>INVESTIGADOR</b>	Grdo_académico	GRDO_ACADÉMICO VARCHAR2(30) NOT NULL
<b>INVESTIGADOR</b>	Tel	TEL INTEGER
<b>LIBRO</b>	Título	TITULO VARCHAR2(50) NOT NULL
<b>LIBRO</b>	Autor_libro	AUTOR_LIBRO VARCHAR2(80) NOT NULL
<b>LIBRO</b>	Editorial	EDITORIAL VARCHAR2(20) NOT NULL
<b>LIBRO</b>	Lugar	LUGAR VARCHAR2(30) NOT NULL
<b>LIBRO</b>	ISBN	ISBN VARCHAR2(50) NULL
<b>LIBRO</b>	ISSN	ISSN VARCHAR2(50) NULL
<b>LIBRO</b>	Año_pub	AÑO_PUB DATE
<b>LIBRO</b>	Fecha_consulta	FECHA_CONSLTA DATE
<b>LIBRO</b>	URL	URL VARCHAR2(100) NULL
<b>LIBRO</b>	Grado_lib	GRADO_LIB VARCHAR2(20) NOT NULL
<b>LIBRO</b>	obsrvacions	OBSRVACIONES VARCHAR2(256) NULL
<b>MOLUSCO</b>	Tamaño	TAMANO NUMBER
<b>MOLUSCO</b>	Sexo	SEXO VARCHAR2(2) NOT NULL
<b>MOLUSCO</b>	Edad	EDAD VARCHAR2(7) NOT NULL
<b>MOLUSCO</b>	Fotografía	FOTGRAFIA VARCHAR2(20) NOT NULL
<b>MOLUSCO</b>	Fрма_concha	FRMA_CONCHA VARCHAR2(20) NOT NULL
<b>MOLUSCO</b>	Ornamentado	ORNAMENTADO VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Espinas	ESPINAS VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Postulas	POSTULAS VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Labios	LABIOS VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Tipo_costilla	TIPO_COSTILLA VARCHAR2(256) NOT NULL

## Capítulo 3 Diseño de un Catálogo Digital

<b>MOLUSCO</b>	Tipo_canal	TIPO_CANAL VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Giro_concha	GIRO_CONCHA VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Ápice	APICE VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Vuelta_corporal	VUELTA_CORPORAL VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Num_vueltas	NUMERO_VUELTAS VARCHAR2(256) NOT NULL
<b>MOLUSCO</b>	Num_conabio	NUM_CANABIO VARCHAR2(10) NOT NULL
<b>NOMBRE_TAXONÓMICO</b>	Nmbre_común	NMBRE_COMUN VARCHAR2(20) NOT NULL
<b>NOMBRE_TAXONÓMICO</b>	Nmbre_cientfco	NMBRE_CIENTFCO VARCHAR2(20) NOT NULL
<b>ORDEN</b>	Nmbre_orden	NOMBRE_ORDEN VARCHAR2(20) NOT NULL
<b>PAÍS</b>	Nmbre_país	NMBRE_PAIS VARCHAR2(30) NOT NULL
<b>PAÍS</b>	estadodistrito	ESTADODISTRITO VARCHAR2(30) NOT NULL
<b>PAÍS</b>	Ciudad_país	CIUDAD_PAÍS VARCHAR2(256) NOT NULL
<b>RECOLECTOR</b>	Nmbre_curador	NMBRE_CURADOR VARCHAR2(50) NOT NULL
<b>RECOLECTOR</b>	Grado_curador	GRADO_CURADOR VARCHAR2(20) NOT NULL
<b>RECOLECTOR</b>	Correo_e	CORREO_E VARCHAR2(50) NOT NULL
<b>RECOLECTOR</b>	Nmbre_inst	NMBRE_INST VARCHAR2(80) NOT NULL
<b>RECOLECTOR</b>	Cd_inst	CD_INST VARCHAR2(50) NOT NULL
<b>RECOLECTOR</b>	Teléfono	TELEFONO VARCHAR2(15) NULL
<b>RECOLECTOR</b>	Num_ejemplares	NUM_EJEMPLARES INTEGER
<b>REF_IMPRESA</b>	Coment_ri	COMMENT_RI VARCHAR2(256)
<b>REGIÓN</b>	Nmbre_región	NMBRE_REGION VARCHAR2(50) NOT NULL
<b>REGIÓN</b>	Clave_región	CLAVE_REGION VARCHAR2(12) NOT NULL
<b>REGIÓN</b>	Longitud	LONGITUD VARCHAR2(6) NOT NULL
<b>REGIÓN</b>	Latitud	LATITUD VARCHAR2(6) NOT NULL
<b>REGIÓN</b>	Altitud	ALTITUD VARCHAR2(6) NOT NULL
<b>REGIÓN</b>	Hora	HORA DATE
<b>REGIÓN</b>	Profundidad	PROFUNDIDAD VARCHAR2(10) NOT NULL
<b>REGIÓN</b>	Salinidad	SALINIDAD VARCHAR2(10) NOT NULL
<b>REGIÓN</b>	Ambiente_terrestre	AMBIENTE_TERRESTRE VARCHAR2(256) NOT NULL
<b>SUBCLASE</b>	Nmbre_sub_clase	NOMBRE_SUBCLASE VARCHAR2(20) NOT NULL
<b>SUBFAMILIA</b>	Nmbre_subfamilia	NOMBRE_SUBFAMILIA VARCHAR2(20) NOT NULL
<b>SUBGÉNERO</b>	Nmbre_subgénero	NMBRE_SUBGENERO VARCHAR2(15) NOT NULL
<b>SUBGÉNERO</b>	Dscubrdor_subgénero	DSCUBRDOR_SUBGÉNERO VARCHAR2(50) NOT NULL
<b>SUBGÉNERO</b>	Fcha_subgénero	FCHA_SUBGÉNERO DATE

## Capítulo 3 Diseño de un Catálogo Digital

---

<b>SUBORDEN</b>	Nmbre_suborden	NMBRE_SUBORDEN VARCHAR2(20) NOT NULL
<b>SUPERFAMILIA</b>	Nmbre_superfamilia	NMBRE_SUPERFAMILIA VARCHAR2(20) NOT NULL

### 3.3.6 Construcción de relaciones

A continuación se presentan las relaciones finales obtenidas del modelo entidad-relación anterior habiendo concluido su etapa de normalización.

- ✓ R1(Coment\_ri)
- ✓ R2(Título, Autor\_libro, Editorial, Lugar, ISBN, ISSN, Año\_pub, Fecha\_consulta, URL, Grado\_lib, observacions)
- ✓ R3(Nmbre\_boletín, Autor\_boletín, Fecha\_pub, Inst\_boletín)
- ✓ R4(Nmbre\_folleto, Autor\_folleto )
- ✓ R5(Nmbre\_catálogo, Nmbre\_colec)
- ✓ R6(Comentario\_ref)
- ✓ R7(Nmbre\_dnador, Inst\_dnadora, Num\_ejemplres )
- ✓ R8(Comentario\_dnador)
- ✓ R9(Comentario\_hábitat)
- ✓ R10(Tipo\_cuerpo\_agua, Microhábitat, Sustrato, Ambiente )
- ✓ R11(Comentario\_región)
- ✓ R12(Nmbre\_región, Clave\_región, Longitud, Latitud, Altitud, Hora, Profundidad, Salinidad, Ambiente\_terrestre)
- ✓ R13(Nmbre\_país, estadodistrito, Ciudad\_país)
- ✓ R14(Tamaño, Sexo, Edad, Fotografía, Frma\_concha, Ornamentado, Espinas, Postulas, Labios, Tipo\_costilla, Tipo\_canal, Giro\_concha, Ápice, Vuelta\_corporal, Num\_vueltas, Num\_conabio)
- ✓ R15(Nmbre\_curador, Grado\_curador, Correo\_e, Nmbre\_inst, Cd\_inst, Teléfono, Num\_ejemplares )
- ✓ R16(Comentario\_recolec)
- ✓ R17(Nmbre\_común, Nmbre\_cientfco )
- ✓ R18(Nmbre\_clase)
- ✓ R19(Nmbre\_sub\_clase)
- ✓ R20(Nmbre\_orden)
- ✓ R21(Nmbre\_suborden)
- ✓ R22(Nmbre\_superfamilia)
- ✓ R23(Nmbre\_fam)
- ✓ R24(Nmbre\_subfamilia)
- ✓ R25(Nmbre\_genero, Dscubrdor\_género, Fcha\_género )
- ✓ R26(Nmbre\_subgénero, Dscubrdor\_subgénero, Fcha\_subgénero)
- ✓ R27(Nmbre\_especie, Dscubrdor\_especie, Fcha\_especie )
- ✓ R28(Comentario\_inv)
- ✓ R29(Nmbre\_inves, Nmbre\_institución, Dirección, C\_elect, Grdo\_académico, Tel )



### 3.3.7 Diseño de las tablas

El mapeo de entidad a tabla es directamente, se amolda exactamente en el momento de la implantación. En la figura 3.9, se muestra el diseño final de la tabla “Molusco”, integrando los detalles anteriormente explicados tales como: entidad, atributos, etc.

En el anexo D se muestra el diseño de todas las tablas que integran el modelo entidad-relación.

Nombre de la tabla: MOLUSCO

Columna	Numero_conabio	Ápice	Vuelta _corporal	Edad	Id_molusco	Ornamentado	Tamaño	Postulas
Tipo llave	PK							
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Sin num	No descrito	Cinco o siete vueltas convexas delgadas y resistentes	l año	M001	No descrito	2	postnucleres

Columna	Giro_concha	Fotografia	Tipo_canal	Tipo_costilla	Labios	Frma_concha	Sexo	Número _vueltas
Tipo llave								
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Muestran líneas tenues axiales ampliamente esparcidas	Sin imagen	sifonal	ornamentado	Labio interno muestra un doblez	Aovada cónica	f	7

Columna	Espinas	Nt_id_nom_tax
Tipo llave		<b>PF</b>
Nulo/ Único	NN	NN
Ejemplos	No descrito	Nt001

Figura 3.9

## 3.4 Etapa de Implementación

Oracle es un administrador de una base de datos relacional. El lenguaje utilizado para acceder a las bases de datos relacionales es el llamado Lenguaje de Consulta Estructurado (SQL). SQL es un lenguaje flexible y eficiente, con características que han sido diseñadas para la manipulación y examen de datos relacionales.

Al implementar la base de datos en SQL, nos permite definir por medio de tablas: almacenar información, manejar una serie de selecciones de información de acuerdo a los requerimientos de la base, realizar cambios en la información y estructura de los datos.

### 3.4.1 Creación de tablas

Con la combinación de SQL se realizó la implementación de la Base de Datos del Catálogo de Moluscos, es decir las entidades, atributos y relaciones obtenidas en el Diagrama de Entidad-Relación y en el diseño de la base de datos, fueron mapeadas a tablas, columnas, llaves primarias y llaves foráneas, mediante sentencias de SQL.

Tal como se observa en los siguientes fragmentos de código SQL, muestra como se crearon algunas tablas. Ver anexo E

#### Tabla "MOLUSCO"

```
PROMPT Creating Table 'MOLUSCOS'  
CREATE TABLE MOLUSCOS,  
(NUM_CANABIO VARCHAR2(10) NOT NULL,  
  ÁPICE VARCHAR2(256) NOT NULL,  
  VUELTA_CORPORAL VARCHAR2(256) NOT NULL,  
  EDAD VARCHAR2(7) NOT NULL,  
  ID_MOLUSCO VARCHAR2(20) NOT NULL,  
  ORNAMENTADO VARCHAR2(256) NOT NULL,  
  TAMAÑO NUMBER NOT NULL,  
  POSTULAS VARCHAR2(256) NOT NULL,  
  
  GIRO_CONCHA VARCHAR2(256) NOT NULL,  
  FOTOGRAFIA VARCHAR2(20) NOT NULL,  
  TIPO_CANAL VARCHAR2(256) NOT NULL,  
  TIPO_COSTILLA VARCHAR2(256) NOT NULL,  
  LABIOS VARCHAR2(256) NOT NULL,  
  FRMA_CONCHA VARCHAR2(20) NOT NULL,  
  SEXO VARCHAR2(2) NOT NULL,  
  NUM_VUELTAS VARCHAR2(256) NOT NULL,  
  ESPINAS VARCHAR2(256) NOT NULL,  
  NT_ID_NOM_TAX VARCHAR2(20) NOT NULL,  
)  
/
```

#### Tabla "ESPECIE"

```
PROMPT Creating Table 'ESPECIES'  
CREATE TABLE ESPECIES  
(FCHA_ESPECIE DATE NOT NULL,  
  DSCUBRDOR_ESPECIE VARCHAR2(50) NOT NULL,  
  ID_ESPECIE VARCHAR2(20) NOT NULL,  
  NMBRE_ESPECIE VARCHAR2(40) NOT NULL  
)  
/
```

### 3.4.2 Creación de llaves primarias y foráneas

```
PROMPT Creating Primary Key on 'MOLUSCOS'  
ALTER TABLE MOLUSCOS  
ADD (CONSTRAINT MOL_PK PRIMARY KEY  
(ID_MOLUSCO))  
/
```

```
PROMPT Creating Primary Key on 'ESPECIES'  
ALTER TABLE ESPECIES  
ADD (CONSTRAINT ESP_PK PRIMARY KEY  
(ID_ESPECIE))
```

### 3.4.3 Creación de índices

```
PROMPT Creating Index 'RI_BOL_FK_I'  
CREATE INDEX RI_BOL_FK_I ON REF_IMPRESAS  
(BOL_ID_BOLETÍN)  
/
```

```
PROMPT Creating Index 'RI_LIB_FK_I'  
CREATE INDEX RI_LIB_FK_I ON REF_IMPRESAS  
(LIB_ID_LIBRO)  
/
```

```
PROMPT Creating Index 'RI_FOL_FK_I'  
CREATE INDEX RI_FOL_FK_I ON REF_IMPRESAS  
(FOL_ID_FOLLETÓ)  
/
```

```
PROMPT Creating Index 'RI_CAT_FK_I'  
CREATE INDEX RI_CAT_FK_I ON REF_IMPRESAS  
(CAT_ID_CATÁLOGO)  
/
```

```
PROMPT Creating Index 'DH_HÁB_FK_I'  
CREATE INDEX DH_HÁB_FK_I ON DET_HÁBITATS  
(HÁB_ID_HÁBITAT)  
/
```

```
PROMPT Creating Index 'DH_MOL_FK_I'  
CREATE INDEX DH_MOL_FK_I ON DET_HÁBITATS  
(MOL_ID_MOLUSCÓ)  
/
```

```
PROMPT Creating Index 'DR2_MOL_FK_I'  
CREATE INDEX DR2_MOL_FK_I ON DET_RECOLECS  
(MOL_ID_MOLUSCO)  
/
```

```
PROMPT Creating Index 'DR2_REC_FK_I'  
CREATE INDEX DR2_REC_FK_I ON DET_RECOLECS  
(REC_ID_RECOLECTOR)  
/
```

```
PROMPT Creating Index 'MOL_NT_FK_I'  
CREATE INDEX MOL_NT_FK_I ON MOLUSCOS  
(NT_ID_NOM_TAX)  
/
```

```
PROMPT Creating Index 'NT_SUB2_FK_I'  
CREATE INDEX NT_SUB2_FK_I ON NOMBRE_TAXONÓMICOS  
(SUB2_ID_SUBORDEN)  
/
```

```
PROMPT Creating Index 'NT_GEN_FK_I'  
CREATE INDEX NT_GEN_FK_I ON NOMBRE_TAXONÓMICOS  
(GEN_ID_GÉNERO)  
/
```

```
PROMPT Creating Index 'NT_SUP_FK_I'  
CREATE INDEX NT_SUP_FK_I ON NOMBRE_TAXONÓMICOS  
(SUP_ID_SUPERFAMILIA)  
/
```

```
PROMPT Creating Index 'NT_SUB1_FK_I'  
CREATE INDEX NT_SUB1_FK_I ON NOMBRE_TAXONÓMICOS  
(SUB1_ID_SUBFAMILIA)  
/
```

```
PROMPT Creating Index 'NT_CLA_FK_I'  
CREATE INDEX NT_CLA_FK_I ON NOMBRE_TAXONÓMICOS  
(CLA_CLA_ID)  
/
```

### CAPÍTULO 4 RESULTADOS

#### 4.1 Etapa de pruebas

Una vez concluido con la programación del código de la Base de Datos se llevo a cabo la depuración de los errores que surgieron de la ejecución del código en SQL, se realizaron varias tipos de pruebas al paso de la creación del código, con el objetivo de obtener las consultas deseadas por parte del Laboratorio de Malacología.

El propósito de las pruebas en la base de datos es determinar la calidad con que la base de datos cumple con los requisitos.

##### 4.1.1 Pruebas de caja blanca

En este tipo de prueba, siempre se está observando el código, hasta obtener una cobertura aceptable de los requerimientos planteados inicialmente. Con la finalidad de verificar lo siguiente:

- A) La importancia del resultado de la búsqueda. Se realizaron diversas consultas para obtener múltiples búsquedas de los ejemplares deseados, ya sea por su nombre científico, común, especie, familia, etc.
- B) La integridad de los datos. Se probaron las acciones de creación y modificación de los datos en las tablas para una correcta funcionalidad.
- C) La validez de los datos. Los errores causados por una entrada incorrecta de datos, son probablemente los más comunes en los datos. Estos errores tienen normalmente su origen en el momento en que se introduce un gran volumen de datos.

En nuestro caso la prueba de caja blanca consistió en una primera etapa en realizar inserción de datos en la base de datos en cada una de las tablas diseñadas y una segunda etapa consistió en modificar y validar la información hasta obtener las consultas deseadas por parte del Laboratorio de Malacología. Una vez concluidas las pruebas se llevo a cabo la etapa de aceptación por parte de los investigadores de dicho instituto, validando las consultas planteadas.

##### 4.1.2 Inserción de datos

A continuación se muestran sólo unos ejemplos relevantes:

Tabla “MOLUSCO”

```
insert into moluscos values(  
'sin num','no descrito','cinco o siete vueltas convexas delgadas y resistentes','1 año','m001','no  
descrito','2','postnuclerares','muestran líneas tenues axiales ampliamente esparcidas','sin  
imagen','sifonal','ornamentado','labio interno muestra un doblez','aovada cónica','f','7','no  
descrito','nt001'  
);
```

```
insert into moluscos values(  
'sin num','desplazado un poco hacia el frente de la parte media','no descrito','3 meses','m002','costillas  
radiales','20 ','no descrito','forma oval de espira baja','sin imagen','no descrito','ornamentado','no  
descrito','forma oval','f','3','no descrito','nt002'  
);
```

```
insert into moluscos values(  

```

## Capítulo 4 Resultados

---

```
'sin num','no descrito','no descrito','sr','m003','ornamentada cada segunda costilla es más grande','30','no descrito','no descrito','sin imagen','anal','no descrito','ornamentado','forma cónica','m','6','sin espinas','nt003'  
);  
Tabla"ESPECIE"
```

```
insert into especies values(  
'01-ene-1891','Dall','e001','Solemya johnsoni'  
);
```

```
insert into especies values(  
'01-ene-1908','Dall','e002','Solemya panamesis'  
);
```

```
insert into especies values(  
'01-ene-1864','Carpenter','e003','Solemya valvulus'  
);
```

### 4.1.3 Consulta de información

Se presentan una serie de ejemplos de consultas, en base a la información que se requiere, y sus resultados.

1.- ¿Cuántas especies hay en una región por clase?

```
select nombre_especie,nombre_clase,nombre_región  
from nombre_taxonómicos,clases,especies,moluscos,det_regións,regións  
where nombre_taxonómicos.esp_id_especie=especies.id_especie  
and especies.nombre_especie='Solemya johnsoni'  
and nombre_taxonómicos.cla_cla_id=clases.cla_id  
and clases.nombre_clase='Bivalvia'  
and moluscos.nt_id_nom_tax=nombre_taxonómicos.id_nom_tax  
and det_regións.mol_id_molusco=moluscos.id_molusco  
and regións.dr_id_cr=det_regións.id_cr;
```

ESPECIE	CLASE	REGIÓN
Solemya johnsoni	Bivalvia	Campeche

2.-¿Cuántos ejemplares de una especie hay en una región?

```
select nombre_especie,nombre_región  
from nombre_taxonómicos,clases,especies,moluscos,det_regións,regións  
where nombre_taxonómicos.esp_id_especie=especies.id_especie  
and especies.nombre_especie='Solemya johnsoni'  
and nombre_taxonómicos.cla_cla_id=clases.cla_id  
and clases.nombre_clase='Bivalvia'  
and moluscos.nt_id_nom_tax=nombre_taxonómicos.id_nom_tax
```

```
and det_regións.mol_id_molusco=moluscos.id_molusco
and regións.dr_id_cr=det_regións.id_cr;
```

ESPECIE	REGIÓN
Solemya johnsoni	Campeche

3.- ¿A qué profundidad se distribuye un conjunto de especies con su clase?

```
select nmbre_especie,nmbre_clase,profundidad
from nombre_taxonómicos,clases,especies,moluscos,det_regións,regións
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and especies.nmbre_especie='Solemya johnsoni'
and nombre_taxonómicos.cla_cla_id=clases.cla_id
and clases.nmbre_clase='Bivalvia'
and moluscos.nt_id_nom_tax=nombre_taxonómicos.id_nom_tax
and det_regións.mol_id_molusco=moluscos.id_molusco
and regións.dr_id_cr=det_regións.id_cr;
```

ESPECIE	CLASE	PROFUNDIDAD
Solemya johnsoni	Bivalvia	150m

4.-¿Cuál es la jerarquización de una especie?

```
select especies.nmbre_especie,
subgéneros.nmbre_subgénero,
géneros.nmbre_género,
subfamilias.nmbre_subfamilia,
familias.nmbre_fam,
superfamilias.nmbre_superfamilia,
subordens.nmbre_suborden,
ordens.nmbre_orden,
subclases.nmbre_cubclase,
clases.nmbre_clase
from especies,subgéneros,géneros,subfamilias,familias,
superfamilias,subordens,ordens,subclases,clases
where especies.nmbre_especie='Solemyajohnsoni' and
subgéneros.nmbre_subgénero='Acharax'
and géneros.nmbre_género='Solemya' and subfamilias.nmbre_subfamilia='Monodontinae'
and familias.nmbre_fam='Caecidae' and superfamilias.nmbre_superfamilia='Solemyacea'
and subordens.nmbre_suborden='Myina'
and ordens.nmbre_orden='Solemya johnsoni'
and subclases.nmbre_cubclase='Myina'
and clases.nmbre_clase='Bivalvia';
```

## Capítulo 4 Resultados

ESPECIE	SUBGÉNERO	GÉNERO	SUBFAMILIA	FAM	SUPERFAMILIA
Solemya Jonhsoni	Acharax	Solemya	Monodontinae	Caecidae	Solemyacea

SUBORDEN	ORDEN	SUBCLASE	CLASE
Myina	Solemya jonhsoni	Myina	Bivalvia

### 5.-¿ Consultar bibliografía de la especie Solemya valvulus?

```
select nombre_especie,titulo,autor_libro
from nombre_taxonómicos,especies,moluscos,det_refs,ref_impresas,libros
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and especies.nombre_especie='Solemya valvulus'
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_refs.mol_id_molusco=moluscos.id_molusco
and ref_impresas.id_ri=det_refs.ri_id_ri
and libros.id_libro=ref_impresas.lib_id_libro;
```

ESPECIE	TITULO	AUTOR
Solemya valvulus	Catálogo de Gastropodos del Golfo de México	Martha Reguero

### 6.-¿ Cuáles especies a recolectado Jacobo Bolaños?

```
select nombre_curador,nombre_especie
from nombre_taxonómicos,especies,moluscos,det_recolecs,recolectors
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and especies.nombre_especie='Solemya panamesis'
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_recolecs.mol_id_molusco=moluscos.id_molusco
and det_recolecs.rec_id_recolector=recolectors.id_recolector;
```

CURADOR	ESPECIE
Jacobo Bolaños Ríos	solemya panamesis

### 7.-¿ Hay algún investigador que estudie a la familia Architecnicidae?

```
select nombre_fam,nombre_inves,grado_académico,nombre_institución
from nombre_taxonómicos,familias,moluscos,det_invs,investigadors
where nombre_taxonómicos.fam_id_familia=familias.id_familia
and familias.nombre_fam='Architecnicidae'
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_invs.mol_id_molusco=moluscos.id_molusco
and investigadors.id_investigador=det_invs.inv_id_investigador;
```

## Capítulo 4 Resultados

---

FAM	INVES	GRADO_ACADÉMICO	INSTITUCIÓN
Architecnicidae	Raymundo Rodriguez	Biólogo	UNAM

---

8.-¿ Cuáles especies han sido donadas?

```
select nombre_especie,int_dnadora
from nombre_taxonómicos,especies,moluscos,det_dnadors,donadors
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_dnadors.mol_id_molusco=moluscos.id_molusco
and det_dnadors.don_id_donador=donadors.id_donador;
```

ESPECIE	INT_DNADORA
Solemya johnsoni	UNAM

9.- ¿ Cuáles especies han sido donadas por la UNAM?

```
select nombre_especie,int_dnadora
from nombre_taxonómicos,especies,moluscos,det_dnadors,donadors
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_dnadors.mol_id_molusco=moluscos.id_molusco
and det_dnadors.don_id_donador=donadors.id_donador
and donadors.int_dnadora='UNAM';
```

ESPECIE	INT_DNADORA
Solemya johnsoni	UNAM
Solemya panamensis	UNAM

10.-¿A qué familia pertenece la especie Solemya valvulus?

```
select nombre_especie,nombre_fam
from especies,familias
where especies.nombre_especie='Solemya valvulus'
and familias.nombre_fam='Architecnicidae';
```

ESPECIE	FAMILIA
Solemya valvulus	Architecnicidae

11.-¿Existe algún catálogo sobre la clase Bivalvia?

```
select nombre_clase,nombre_catálogo
from nombre_taxonómicos,clases,moluscos,det_refs,ref_impresas,catálogos
```



## Capítulo 4 Resultados

---

```
where nombre_taxonómicos.cla_cla_id=clases.cla_id
and clases.nmbre_clase='Bivalvia'
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_refs.mol_id_molusco=moluscos.id_molusco
and ref_impresas.id_ri=det_refs.ri_id_ri
and catálogos.id_catálogo=ref_impresas.cat_id_catálogo;
```

CLASE	CATÁLOGO
Bivalvia	Catálogo de moluscos del pacífico mexicano

### 12.-¿Características taxonómicas de la especie Solemya panemesis?

```
select nmbre_especie,nmbre_común,edad,espinas,frma_concha,
giro_concha,ornamentado,postulas,sexo,tamaño
from nombre_taxonómicos,especies,moluscos
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax;
```

ESPECIE	NOMBRE_COMÚN	EDAD	ESPINAS	FORMA_CONCHA	GIRO_CONCHA
solemya panemesis	johnsoni	1 año	no descrito	aovada cónica	muestran líneas tenues axiales ampliamente esparcidas

ORNAMNETADO	POSTULAS	SEXO	TAMAÑO
no descrito	postnucleares	f	2

### 13.- ¿Cuáles especies se desarrollan en ambiente de aguas someras?

```
select nmbre_especie,ambiente
from nombre_taxonómicos,especies,moluscos,det_hábitats,hábitats
where nombre_taxonómicos.esp_id_especie=especies.id_especie
and nombre_taxonómicos.id_nom_tax=moluscos.nt_id_nom_tax
and det_hábitats.mol_id_molusco=moluscos.id_molusco
and det_hábitats.háb_id_hábitat=hábitats.id_hábitat;
```

ESPECIE	AMBIENTE
Solemya valvulus	aguas someras

### 4.2 Comentarios

Se obtuvo una propuesta de diseño del sistema que satisface los requerimientos de los usuarios del Laboratorio de Malacología del Instituto de Ciencias del Mar y Limnología de la UNAM, en donde analizaron diversas especies para obtener características necesarias para formar la colección de moluscos la cual es la base para el diseño del catálogo digital.

Para obtener el resultado, se consultó bibliografía referente al tema de moluscos, para lograr entender terminología referente al tema.

En el manejo de la información recabada por la investigación bibliográfica y estudio de los ejemplares físicos, se optó por dividir en cuatro partes la información, para un mejor análisis de la misma.

Al momento de analizar la clasificación taxonómica, se optó por formar nueve niveles, para un mejor estudio, se clasificaron un gran número de ejemplares de manera rústica (en papel), para observar que no hubiera algún molusco que no se pudiera colocar en los nueve niveles definidos. Esta actividad permitió analizar minuciosamente un gran número de moluscos, obteniendo características útiles para el modelado de entidades y de igual manera realizar más adelante cada entidad de los nueve niveles, obteniendo sus atributos. Todas las especies se deben acompañar de la fecha de su descubrimiento y su autor, en esta ocasión sólo se integraron dichos datos en los niveles: género, subgénero y especie. A petición de la encargada del laboratorio.

Al definir las entidades se tomaron en cuenta los atributos más específicos de las especies, ya que al manejar un gran volumen de ejemplares resulta difícil establecer características generales de su taxonomía y su hábitat. Son características importantes para los investigadores ya que es una forma de clasificar el ejemplar y más aún la manera de recolección del mismo.

Partiendo de las entidades establecidas se construyeron las tablas correspondientes a cada una de las entidades con sus respectivas llaves y restricciones.

La herramienta utilizada para el modelado del sistema fue UML, ya que nos permite tener una amplia perspectiva del diseño, se puede observar la interacción del usuario con el sistema, permite identificar errores en alguna parte del sistema si lo llegara a presentar y se puede modificar sin afectar el resto del diseño. Es compatible con programas tales como: c++, java, etc. para su implementación, además de permitir la modelación tanto estática como dinámica del sistema.

Al momento de modelar el sistema se procuró ser lo más explícito posible, por tal motivo se utilizaron herramientas amigables con la finalidad de que con dicha modelación en una etapa siguiente el programador se encargara de realizar la implementación.

### CAPÍTULO 5 CONCLUSIONES

Se realizó una revisión de los fundamentos, de las bases de datos y un estudio breve de la metodología UML, con la finalidad de aplicar estos conocimientos en la determinación de la propuesta de diseño del catálogo digital. Con esta etapa se cumplió con el objetivo de estudiar los conceptos básicos de las Bases de Datos y del Lenguaje de Modelación Unificado.

Como parte fundamental de éste trabajo se aplico la modelación en base a UML por lo que, se realizó la recolección de la información sobre las necesidades a cubrir por el sistema, en este caso el catálogo digital. En ésta etapa también se analizaron los procesos seguidos por los investigadores al actualizar y mantener su colección de moluscos, mismos que serán automatizados por el sistema.

Dentro del diseño del sistema se tiene una componente importante que es la base de datos a diseñar en función de la información recavada que sirvió para definir las características a cubrir por la base de datos a diseñar mediante el diagrama de entidad- relación, ya que permite manejar diagramas de entidades las cuales nos permiten analizar los atributos de cada una de ellas. De la misma forma se diseñaron las tablas correspondientes para la futura implementación de la base de datos, con esta tarea se cumplió con el objetivo de diseñar una base de datos del catálogo digital.

En conclusión se propone el diseño de un catálogo digital de moluscos del Norte de América, el cual incluye los diagramas de caso de uso y de clases, obtenidos por modelación UML. Así mismo se presenta el diagrama entidad-relación, el cual se propone como diseño de la base de datos.

#### Trabajo a futuro

Con el modelo propuesto se tienen como tareas a futuro definir el lenguaje de programación a utilizar para la implementación del sistema propuesto.

Con ello se genera el código requerido para la implementación del catálogo digital de moluscos, en cuanto a la base de datos los resultados obtenidos permiten implementarla en cualquier SMD, como fue implementada en Oracle.

**A  
N  
E  
X  
O  
S**

## METODOLOGIA UML

### 3.1 Introducción

UML es un lenguaje gráfico de modelamiento que usa conceptos de orientación por objetos. Este lenguaje tiene una sintaxis y una semántica bien definidas, sirviendo además para todas las etapas de desarrollo. [Robbins, 1998][Booch,1999].

A partir de 1994, Grady Booch y Jim Rumbaugh se unieron para formar la empresa “Rational Software Corporation”, así mismo unen sus conocimientos y como resultado logran sacar al mercado en 1995 UML (Unified Modeling Language) 0.8, que vendría siendo la primera versión de UML. Más tarde se une al equipo Ivan Jacobson, creador de OOSE (Object Oriented Software Engineer). Sus propósitos eran que el método que ellos presentaran fuese capaz de modelar cualquier sistema con conceptos de la orientación a objetos (OO) y se convierta en un lenguaje general para el modelado de los sistemas.

### 3.2 Conceptos Básicos

En UML se utilizan diferentes relaciones y elementos para el modelado, tiene una semántica y sintaxis definidas y nos sirven para todas las etapas del proyecto.

Para comenzar un proyecto se debe de comenzar por la identificación de los componentes que integrarán el sistema. En cada uno de los componentes se especifica los elementos que los representan en UML, de igual forma con las conexiones. Algunos componentes o estructuras no se podrán representarse, pero se utilizarían los mecanismos de extensión.

Se debe de incluir ciertas restricciones en los conectores, componentes y relaciones que se deban incluir en la estructura.

Como última parte del proyecto se deben de verificar las relaciones entre las diferentes estructuras, para que la persona que esté modelando verifique la consistencia de su sistema. [Hurtado,2003]

#### 3.2.1 Elementos básicos

Los elementos básicos de dicho lenguaje representan principalmente las partes estáticas del sistema, son:

- **Clase.**-describe un conjunto que tienen los mismos atributos, operaciones. métodos, relaciones y significado. Para UML una clase es una implementación de un tipo. Se representa por medio de un rectángulo con tres divisiones internas, la primera es para indicar el nombre, la segunda para los atributos (estructura) y la tercera para los métodos (operaciones).

Sus componentes son:

- a) Atributo. Son las propiedades de una clase o un tipo, se puede identificar por medio de un nombre. Encontramos atributos simples y complejos.
- b) Operación o método. Es la función que le indica a las clases su comportamiento de sus objetos. Para separar las listas de atributos y métodos, se puede ocupar estereotipos.
- c) Objeto: es una instancia de una clase. Se caracteriza por tener un estado definido por un conjunto de valores de atributos, un comportamiento representado por sus operaciones y métodos, tiene una identidad única.

- ❑ **Casos de uso.** Es un proceso del sistema.
- ❑ **Componentes.** Son las herramientas gráficas para modelar el sistema.
- ❑ **Nodos.** Es un componente (procesador) en el cual se localizan datos, se corren programas, etc.
- ❑ **Paquetes.** Son módulos que agrupan ciertas características y funciones que puedan dar un servicio al exterior, de forma gráfica se representan como un grupo de elementos del modelo.
- ❑ **Metaclass.** Es una clase cuyas instancias son clases.
- ❑ **Tipos:** Establece un comportamiento específico para las clases.
- ❑ **Interfaz.** Muestra el uso de un tipo para describir el comportamiento externo de cualquier elemento del modelo.
- ❑ **Relación entre clases.** Representa las relaciones entre las clases, que marcan los tipos de relaciones.

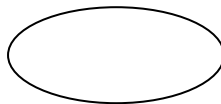
### 3.2.2 Componentes

Un componente es aquel que se puede reutilizar ó sustituir.

#### A) Casos de uso

Un caso de uso es una tarea que un actor necesita ejecutar con la ayuda del sistema, por lo que se debe de implementar en el sistema de software. [Hurtado, 2003][Stevens 2002]

Representación gráfica:



Nombre

Figura 3.1

Se usa el mismo elemento Caso de Uso de UML

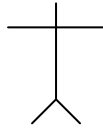
#### B) Actores

Un actor, que normalmente aparece con el símbolo de un muñeco, representa a una persona, sistema o dispositivo que interactúa con el sistema, recibiendo los resultados o participando en algunas de las acciones de un caso de uso. Una relación de comunicación entre actor y un caso no nos indica que necesariamente este implicado en la ejecución de esa tarea; significa que puede ser, dependiendo de las circunstancias. Por lo general representa un rol, por ejemplo: profesor, etc. [Hurtado,2003] [Stevens, 2002]

Actores no humanos: Se debe de considerar como un sistema externo o dispositivo.

Representación gráfica:





Nombre  
Figura 3.2

Se usa el elemento Actor de UML

### C) Módulos

Un módulo es una agrupación de funciones que tengan alguna relación entre ellas, por lo que, se puede presentar un bloque de servicio completo al exterior una vez que se haya terminado de desarrollar. [Hurtado,2003] [Stevens, 2002]

Representación gráfica:

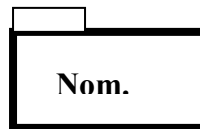


Figura 3.3

Un módulo se representa con el elemento Paquete de UML.

### D) Clases

Una clase es la representación abstracta de un conjunto de objetos que debe modelar el sistema. Cada clase contiene todas las características y el comportamiento de los objetos que representa, es quien determina incluso la manera en que reacciona un objeto.

Una clase puede ser de tipo interfaz (interactúa con el exterior), control (realiza operaciones y controla otras clases) entidad (hace persistentes los datos). En su representación gráfica la interfaz pública está formada por líneas marcadas con "+" y la interfaz privada con "-". [Hurtado,2003] [Stevens, 2002]

Representación gráfica:

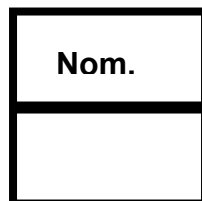


Figura 3.4

Es el mismo elemento Clase de UML

### E) Unidades de software

Es un conjunto de programas o procedimientos que ejecutan las acciones del sistema y se implementan en archivos físicos. Dichas unidades tienen asociado valor adicional, que puede ser: filtro, procedimental, objetos, repositorio de datos activo u otro. [Hurtado,2003] [Stevens ,2002]

Representación gráfica:

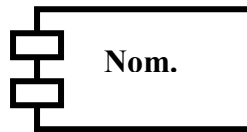


Figura 3.5

Se usa el elemento Componente de UML

## F) Sistemas Externos

Es un sistema de la organización que interactúa con el sistema que se está desarrollando. Se tiene que crear un estereotipo para poder representar este componente que se llamará Externo. [Hurtado,2003]

Representación gráfica:

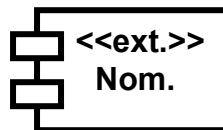


Figura 3.6

## G) Herramientas de Software

En algunas ocasiones se necesitan programas o sistemas ajenos a nuestro sistema en desarrollo para su buen funcionamiento tales como: el sistema operativo, un programa navegador de Internet, etc.

Se tiene que crear un estereotipo Herramientas, basado en el elemento componente de UML. [Hurtado,2003]

Representación gráfica:

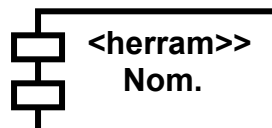


Figura 3.7

## H) Procesador



Nos representa un procesador y memoria donde podemos encontrar datos y los programas y así mismo correr los programas. Podemos ubicar su posición física y asea en la organización o en alguna ciudad. Su uso es variado ya que puede ser usado como cliente, servidor o terminal, etc. [Hurtado,2003]

Representación gráfica:

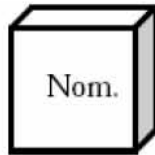


Figura 3.8

Se usa el elemento Nodo de UML.

### I) Dispositivo

Los dispositivos son elementos de hardware que presentan una interacción con el sistema, por ejemplo, un módem, multímetro, etc. [Hurtado,2003]

Representación gráfica:

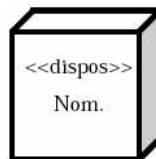


Figura 3.9

### 3.2.3 Mecanismos de extensión

UML nos permite construir o modificar la estructura del sistema ya existente por medio de tres mecanismos, los cuales son.

- Valores adicionados (tagged values): Este mecanismo nos permite agregar nuevas propiedades o atributos a los elementos del modelo.
- Restricciones (constraints): nos permiten agregar nuevas semánticas o cambiar las ya existentes.
- Estereotipos: nos permiten crear nuevos elementos para el modelo de los ya existentes. Al crear un nuevo estereotipo puede tener sus propias características (propiedades y restricciones).

### 3.3 Conectores

- Generalización (herencia):** Nos indica que un componente hereda los atributos y comportamiento del otro. [Hurtado,2003]



Figura 3.10

- **Participación:** Permite a los actores relacionarse con los diferentes casos de usos, mostrando si existen algún tipo de interacción entre ellos. [Hurtado,2003]

{tipo participación}

Figura 3.11

- **Inclusión:** Nos indica que las acciones del caso de uso que recibe la relación se sumarán a las acciones del caso que la inicia. [Hurtado,2003]

--<<include>>-->

Figura 3.12

- **inclusión condicionada:** se adicionan las acciones del caso de uso que recibe de la relación, pero solo cuando se cumple una condición dada. [Hurtado,2003]

--<<condic>>  
--<condición>-->

Figura 3.13

### 3.4 Estructuras

#### 3.4.1 Estructura funcional

Es una estructura de las más importantes, ya que se convierte en un medio de comunicación útil con los usuarios, permitiendo observar las relaciones del sistema con los usuarios.

#### 3.4.2 Estructura de llamados

Esta estructura contiene los servicios que ofrece y los eventos que se llevan acabo en cada unidad de software y sus relaciones correspondientes.

- **Servicio:** representa los servicios que ofrece una unidad de software y puede ser utilizado por otros componentes. [Hurtado,2003]

-----○  
Nombre

Figura 3.14

- **Evento:** son los eventos que produce una unidad de software o sistema externo y puede ser escuchados por otros componentes. [Hurtado,2003]

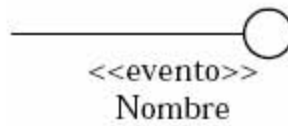


Figura 3.15

- Si el evento incluye alguna condición, se puede incluir en la representación gráfica.

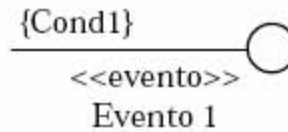


Figura 3.16

- **Uso:** Permite que se relacionen los componentes con los conectores y eventos, para indicar la interacción o dependencia entre ellos. El tipo de conector nos indica la forma en la cual se lleva a cabo la interacción. [Hurtado,2003]



Figura 3.17

### 3.4.3 Estructura modular

Muestra una división del sistema en módulos más pequeños.

- Estructura de uso**  
Nos presenta las relaciones de dependencia que se presentan en los componentes del sistema.
- Estructura de clases**  
Representa el modelo lógico de los datos necesarios en el sistema.
- Estructura de Flujo de datos**  
Modela el intercambio de datos e información que relacionan los diferentes componentes de software.
- Estructura de sincronización**  
Nos presenta las relaciones de sincronización con sus respectivas restricciones para el buen funcionamiento del sistema.
- Estructura física**  
Modela la distribución de los diferentes componentes físicos de los que se compone el sistema.

### 3.5 Diagramas

Los elementos y relaciones se agrupan en diagramas que representan diferentes formas del sistema. Los diagramas de UML son estáticos y dinámicos.

**A) Diagramas de clases:** Denota la estructura de información del sistema con las características de cada una de las clases, operaciones, interfaces, atributos, relaciones y paquetes.

Para establecer relaciones entre las clases tenemos las siguientes: [Stevens,2002][Moreno Martinez][Hernandez Domingo]

- ❑ Dependencia: se relacionan dos clases una dependiente y otra independiente (u objetos), es decir si se cambia el elemento independiente puede necesitar cambios en los dependientes. Se representa con una línea punteada direccional, indicando con una flecha el sentido de la dependencia.
- ❑ Rol: Describe la semántica de la relación en el sentido indicado, cada asociación tiene dos roles; cada rol es una dirección en la asociación. Se puede representar por el nombre de la clase.
- ❑ Multiplicidad: nos indica cuantos objetos de esa clase pueden participar en la relación.
- ❑ Asociación: Es una línea que uno dos ó más clases. Se representa con una flecha discontinua va desde la clase utilizadora a la clase utilizada.
- ❑ Asociación binaria: es una relación sencilla entre dos clases. Se representa con una línea que une dos clases.
- ❑ Asociación n-aria: Se puede dar la relación entre tres o más clases. Se representa con un diamante del cual salen las líneas de asociación a las clases.
- ❑ Composición: Es una relación fuerte ya que desaparece el elemento dependiente Si se destruye al que lo contiene, el objeto contenido es parte constitutiva e importante del que lo contiene y los objetivos que se tienen no son compartidos. Su representación gráfica es un rombo del lado de la clase que contiene a la otra en la relación.
- ❑ Generalización: se presenta en dos clases, hay una superclase y una subclase, la subclase toma los atributos y métodos de la superclase, es decir, se presenta una relación de herencia entre clases. Se representa por medio de un triangulo sin rellenar en el lado de la superclase.
- ❑ Refinamiento: la relación representa la relación entre dos elementos donde uno ya ha sido especificado a detalle.
- ❑ Agregación: relaciona unan clase ya armada con una clase componente, no necesita dependencia existencial. Su representación gráfica es un rombo sin rellenar.

**B) Diagrama de objetos:** Muestra instancias de clases (objetos ) junto con sus valores en sus atributos y relaciones.

**C) Diagrama de componentes:** Presenta la organización y dependencia entre todos los componentes físicos del sistema. Nos muestra la organización y las dependencias entre los componentes, se realizan por partes los diagramas.

Nos sirve para poder observar los componentes que se pueden compartir entre sistemas o con otras partes del sistema.

**D) Diagrama de despliegue:** Indica donde se encuentra el software en el hardware que lo contiene.

## E) Diagrama de casos de uso

Los diagramas de caso de uso nos sirven para visualizar una parte del sistema, es decir, la interacción de los usuarios (actor) con el sistema. Nos especifica como deben de comportarse y no como están implementadas las partes que define. Un buen sistema es aquel que documenta los códigos para que puedan ser reutilizables por otros desarrolladores. En pocas palabras dicho diagrama indica esta parte debe hacer esto cuando pase esto.

Se representan en el diagrama por una elipse que denota un requerimiento solucionando por el sistema.

En el diagrama nos encontramos con diferentes componentes que pueden mantener relaciones entre ellas:

- a) Caso de uso: Puede tener relación con otro caso de uso, sus relaciones son:
  - a. Extiende (extends): Denota que un caso de uso A hacia un caso de uso B indica que el caso de uso de B implementa la funcionalidad del caso de uso A. Se usa cuando se describe una variación sobre el normal comportamiento.
  - b. Incluye
  - c. Herencia (generalization): Denota cuando un caso de uso A transfiere sus atributos a un caso de uso B.
  - d. Usa (use). Relación entre dos casos de uso, denota la intervención del comportamiento de un escenario en otro. Se utiliza cuando se repite un caso de uso o más casos de uso separados. Comúnmente no hay actor asociado con el caso de uso en común.

Actores: se puede relacionar con un actor y un caso de uso ó con otro actor con la relación y es: [Hernández Domingo, Moreno Gerardo]

- a) Comunica (communicates): entre un actor y un caso de uso, o con otro actor denota la participación del actor en el caso de uso determinado.
- Parte del sistema: Representa las diferentes partes del sistema y contiene los casos de uso que lo forman, se representa por un cuadro.

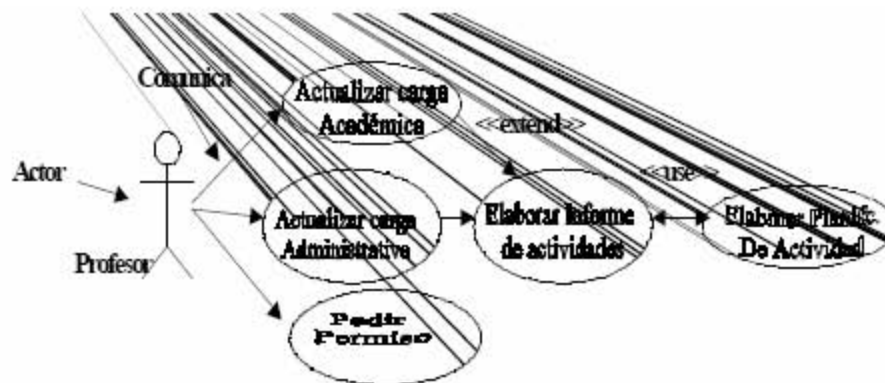


Fig. 3.18 Tomado de Hernández Domingo [www.magma.com.ni/~jorge/upoli\\_uml](http://www.magma.com.ni/~jorge/upoli_uml)

**F) Diagrama de interacción:** Presenta los diagramas de secuencia y de colaboración. Los objetos y relaciones entre ellos desde un punto de vista dinámico.

**G) Diagrama de estado:** Muestra los diferentes estados, eventos y transiciones entre las clases u objetos. Sus principales elementos son:

- ✓ Estado: Representa un lapso de tiempo del objeto (no instantáneo) el cual espera cierta operación y tiene cierto estado característico.
- ✓ Elemento: Es una relación de tres o más estados en una transición de varios destinos.
- ✓ Transición: Cuando un objeto pasa de un estado a otro.
- ✓ Existen tres tipos: simples, reflexivas y complejas.

**H) Diagrama físico:** Muestra la comunicación y distribución de todos los componentes en los dispositivos de hardware.

**I) Diagrama de secuencia (colaboración):** Muestran a los diferentes objetos y sus relaciones que se presentan entre ellos, enviando mensajes sin perder información, es decir, son las llamadas entre clases desde un punto concreto del sistema. Son muy útiles ya que nos permite observar la vida de los objetos de igual manera podemos observar dificultades para el flujo de información. Se representa un diagrama por cada llamada tomando en cuenta un punto de partida.

**J) Diagrama de actividades:** Muestra el flujo entre los objetos, donde casi todos los estados son estado de acción. Sirven para mostrar transiciones internas, in tomar en cuenta eventos externos.

Sus elementos son:

1. Estado de acción: muestra un estado con acción interna, con lo menos una transición que nos indica la terminación de la acción. Su representación gráfica es un rectángulo con bordes redondeados.
2. Estado de transición: Indica que un objeto que se encuentra en primer estado realizará una acción específica y entrara al segundo estado cuando el evento implícito ocurra, siempre y cuando se cubran las condiciones.

### 3.6 Mecanismos de Extensión

UML se ha convertido en un lenguaje de mayor demanda ya que se puede aplicar en diferentes áreas tanto en medios empresariales como académicos.

Para poder lograr que UML represente adecuadamente todos los aspectos del sistema y nos permita construir nuevos elementos o modificar la semántica de los ya existentes. UML tiene tres mecanismos son:

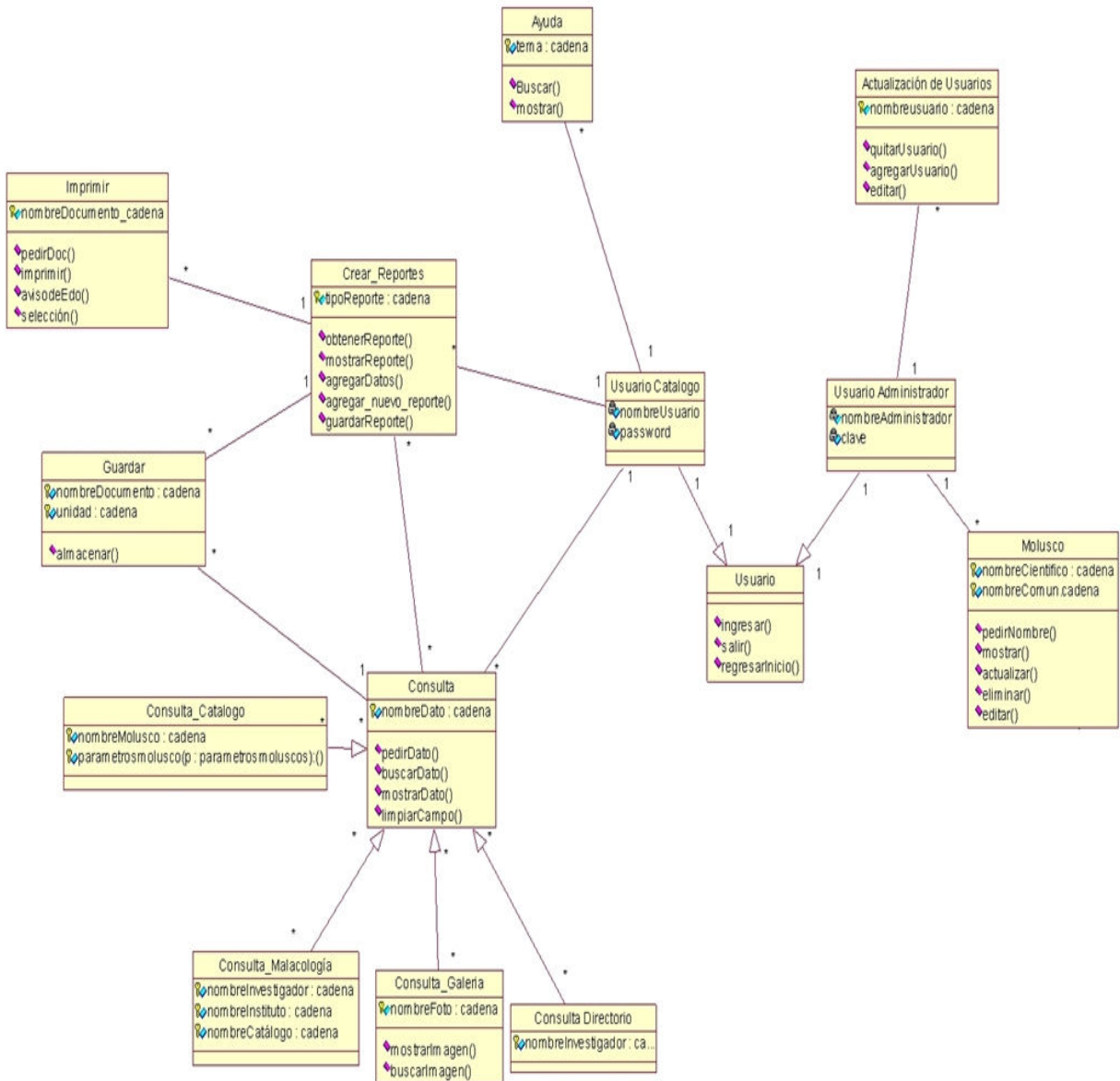
- Valores adicionados (tagged values): nos permiten adicionar nuevas propiedades o atributos a los elementos del modelo ya existentes.
- Restricciones (constrains): Nos permiten agregar o cambiar la semántica ya existente.
- Estereotipos: Nos permiten hacer nuevos elementos basados en otros ya existentes. Los estereotipos creados puede reunir sus propias propiedades y restricciones.

### 3.7 Alcances de UML

UML unificó una serie de estándares de desarrollo Orientado a Objetos, por mencionar algunos: Rumbaugh, Jacobson, Odell, Meyer, etc.

Podemos atrevernos a mencionar que UML será el lenguaje de modelado Orientado a Objetos estándar por excelencia durante los próximos años, ya que vino a llenar un importante vacío existente entre las especificaciones de diseño y el cliente.

Ahora bien, detrás de dicho lenguaje existe la participación de importantes empresas que impulsan su uso y desarrollo, la aceptación del OMG como notación estándar y la gran cantidad de herramientas que actualmente soportan la notación UML.







Nombre de la Tabla "RECOLECTORS"

Nombre de la Tabla "LIBROS"

Columna	Id_recolector	Correo_e	Cd_inst	Num_ejemplares	Nmbre_inst	Teléfono	Grado_curador	Nmbre_curador
Tipo llave	PK							
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN	NN
Ejemplos	R001	<a href="mailto:ajus@yahoo.com">ajus@yahoo.com</a>	México	15	UNAM	54152033	Técnico	Agustín Rojas Paredes
	R002	Jacob36@hotmail.com	México	5	UAM	15234478	Doctor	Jacobo Bolaños Ríos
	R003	Angela55@yahoo.com.mx	México	30	UNAM	52221355	Biologa	Ángeles Prado Uribe

Columna	URL	Fecha_consulta	Editorial	Autor_libro	Título	Año_pub	Grado_lib	Id_libro
Tipo llave								PK
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Sin url	12-sep-1992	Prentice may	Antonio García Cubas	Gastropodos y más	01-ene-1990	Maestria	L001
	Sin url	25-mar-2005	Grijalbo	Esteba González Flores	Cephalopodos	01-ene-2004	Licenciatura	L002
	Sin url	08-nov-2005	UNAM	Martha Reguero	Catálogo de Gasterópodos del golfo de México	01-mar-2004	doctorado	L003

Columna	ISBN	Observacions	lugar	ISSN
Tipo llave				
Nulo/ Único	NN	NN	NN	NN
Ejemplos	xxx	Sin observaciones	México D.F.	Xxx
	xxx	Sin observaciones	México D.F.	Xxx
	xxx	Sin observaciones	México D.F.	xxx

Nombre de la Tabla "CATÁLOGOS"

Columna	Id_catálogo	Nmbre_catálogo	Nmbre_colec
Tipo llave	PK		
Nulo/ Único	NN	NN	NN
Ejemplos	C001	Catálogos de moluscos del pacifico mexicano	Bivalbios
	C002	Catálogo ilustrado sobre Cephalopodos	Moluscos
	C003	Catálogo de moluscos del norte de América	Polyplacophora

Nombre de la Tabla "REF\_IMPRESAS"

Nombre de la Tabla “INVESTIGADORS”

Columna	Cment_ri	Id_ri	Fol_id_folleto	Bol_id_boletín	Cat_id_catálogo	Lib_id_libro
Tipo llave		PK	PF	PF	PF	PF
Nulo/ Único	NN	NN	NN	NN	NN	NN
Ejemplos	Sin comentario	Ri001	Fo001	B001	C001	L001
	Sin comentario	Ri002	Fo002	B002	C002	L002
	Sin comentario	Ri003	Fo003	B003	C003	L003

Nombre de la Tabla “FAMILIAS”

Columna	Grado_aca démico	drección	Nmbre_ins titución	tel	C_elect	Id_investigador	Nmbre_investigador
Tipo llave						PK	
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Doctor	Circuito escolar s/n cd.universitaria	UNAM	22563094	reguero@una m.mx	I001	Martha Reguero
	Doctor	Circuito escolar s/n cd.universitaria	UNAM	22563093	<a href="mailto:steven@unam.mx">steven@unam .mx</a>	I002	Steven Czitrom
	Biólogo	Circuito escolar s/n cd.universitaria	UNAM	22561594	raymundo@un am.mx	I003	Raymundo Rodríguez

Columna	Id_familia	Nmbre_fam
Tipo llave	PK	
Nulo/ Único	NN	NN
Ejemplos	F001	Caecidae
	F002	Turritellidae
	F003	Architecnicidae

Nombre de la Tabla “HÁBITATS”

Columna	microhábitat	ambiente	Id_hábitat	sustrato	Tipo_cuerpo
Tipo llave			PK		
Nulo/ Único	NN	NN	NN	NN	NN
Ejemplos	Patzizales	Aguas someras	H001	Arenoso	mar
	Aguas profundas	Arena	H002	Arenoso	Mar
	En la costa enterradas	Aguas someras	H002	Poco profundo	Mar

Nombre de la Tabla “FOLLETOS”

Nombre de la Tabla "CLASES"

Columna	Id_folleto	Autor_folleto	Nmbre_folleto
Tipo llave	PK		
Nulo/Único	NN	NN	NN
Ejemplos	Fo001	Carlos Domínguez R.	Algo más sobre las especies Gastrophodas
	Fo002	Enriqueta Urbina	Estudio de la familia Architectonicidae
	Fo003	Miriam Cancino	Architectonicidae

Columna	Cla_id	Nmbre_clase	Id_clase
Tipo llave	PK		
Nulo/Único	NN	NN	NN
Ejemplos	001	Bivalbia	C001
	002	Cephalopoda	C002
	003	polyplacophora	C003

Nombre de la Tabla "SUBGÉNEROS"

Columna	Nmbre_subgénero	Id_subgénero	Fcha-subgénero
Tipo llave		PK	
Nulo/Único	NN	NN	NN
Ejemplos	Acharax	Sg001	01-ene-1891
	Petrasma	Sg002	01-ene-1908
	Nucula	Sg003	01-ene-1799

Nombre de la Tabla "DONADORS"

Columna	Num_ejemplares	Id_donador	Int_dnadora	Nmbre_dnador
Tipo llave		PK		
Nulo/Único	NN	NN	NN	NN
Ejemplos	3	D001	UNAM	Dr.Carlos Ortiz V.
	15	D002	UNAM	Esteban Hernández Pinsón
	8	D003	UAM	Angélica López

Nombre de la Tabla "DET\_HÁBITATS"

Columna	Id_ch	Comntario_hábitat	Mol_id_molusco	Háb_id_hábitat
Tipo llave	PK		PF	PF
Nulo/Único	NN	NN	NN	NN
Ejemplos	Ch001	Sin comentario	M001	h001
	Ch002	Sin comentario	M002	h002
	Ch003	Sin comentario	M003	h003

## Nombre de la Tabla "ORDENS"

Columna	Nmbre_orden	Id_orden
Tipo llave		PK
Nulo/Único	NN	NN
Ejemplos	Solemya Jonson	O001
	Solemya panamesis	O002
	Solemya valvulus	O003

## Nombre de la Tabla "DET\_RECOLECS"

Columna	Cmntario_recolec	Id_rec	Mol_id_mol	Rec_id_recolector
Tipo llave		PK	PF	PF
Nulo/Único	NN	NN	NN	NN
Ejemplos	Sin comen	Dr001	M001	r001
	Sin comen	Dr002	M002	r002
	Sin comen	Dr003	M003	R003

## Nombre de la Tabla "MOLUSCOS"

Columna	Num_co nabio	Ápice	Vuelta_corporal	Edad	id_molusco	Ornamentado	Tamaño	Postulas
Tipo llave					PK			
Nulo/Único	NN	NN	NN	NN		NN	NN	NN
Ejemplos	Sin num	No descrito	Cinco o siete vueltas, vueltas convexas delgadas y resistentes	1 año	M001	No descrito	2	Postnucleares
	Sin num	Desplazad o un poco hacia el frente de la parte media	No descrito	3 meses	M002	Costillas radiales	20	no descrito
	Sin num	No descrito	No descrito	SR	M003	Ornamentada cada segunda costilla es más grande	30	No descrito

Columna	Giro_concha	Fotografia	Tipo_canal	Tipo_costilla	labios	Frma_concha	Sexo	Num_vueltas
Tipo llave								
Nulo/Único	NN	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Muestran líneas tenues axiales ampliamente esparcidas	Sin imagen	sifonal	ornamentado	Labio interno muestra un doblez	Aovada cónica	F	7
	Forma oval despirbaja	Sin imagen	No descrito	Ornamentad o	No descrito	Forma oval	F	3
	No descrito	Sin imagen	anal	No descrito	ornamnetado	Forma cónica	M	6

	Espinas	Nt_id_no m_tax
Columna	Comentario_ref	Id_r
		Ri_id_ri
		Mol_id_molusco
Nulo/ Único	NN	NN
Ejemplos	No descrito	nt001
Tipo llave	No descrito	nt002
Nulo/ Único	Sin espinas	Nt003

### Nombre de la Tabla “ESPECIES”

Columna	Fcha_especie	Dscubrdr	Id_especie	Nmbre_especie
Tipo llave			PK	
Nulo/ Único	NN	NN	NN	NN
Ejemplos	01-ene_1891	Dall	E001	Solemya Jonson
	01-ene_1908	Dall	E002	Solemya panamesis
	01-ene_1864	Carpenter	E003	Solemya valvulus

### Nombre de la Tabla “SUBCLASES”

Columna	Nmbre_subclase	Id_subclase
Tipo llave		PK
Nulo/ Único	NN	NN
Ejemplos	Mina	sc001
	Pholadina	sc002
	Euthecosomata	Sc003

### Nombre de la Tabla “NOMBRE\_TAXONÓMICOS”

Columna	Nmbre_común	Id_nom_tax	Sub3_id_s ubclase	Cla_cla_id	Gen_id_géne ro	Sup_id_superfamil ia	Sub2_id_suborde n
Tipo llave		PK	PF	PF	PF	PF	PF
Nulo/ Único	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Johnsoni	Nt001	Sc001	001	G001	Sf001	so001
	Panamesis	Nt002	Sc002	002	G002	Sf002	so002
	Valvulus	Nt003	Sc003	003	G003	Sf003	So003

Columna	Sub_id_subgénero	Fam_id_fam	Esp_id_especie	Ord_id_orden	Sub1_id_subfamilia
Tipo llave	PF	PF	PF	PF	PF
Nulo/ Único	NN	NN	NN	NN	NN
Ejemplos	Sg001	F001	E001	O001	Subf001
	Sg002	F002	E002	O002	Subf002
	Sg003	F003	E003	O003	Sub003

### Nombre de la Tabla “DET\_REFS”

Tipo llave		PK	PF	PF
Nulo/ Único	NN	NN	NN	NN
Ejemplos	Sin comentario	Dr001	Ri001	m001
	Sin comentario	Dr002	Ri002	m002
	Sin comentario	Dr003	Ri003	m003
Columna	Nmbre_superfamilia	Id_superfamilia		

### Nombre de la Tabla "DET\_DNADORS"

Columna	Id_cd	Comntario_dnador	Don_id_donador	Mol_id_molusco
Tipo llave	PK		PF	PF
Nulo/ Único	NN	NN	NN	NN
Ejemplos	Dd001	Sin comentario	D001	m001
	Dd002	Sin comentario	D002	m002
	Dd003	Sin comentario	D003	M003

### Nombre de la Tabla "BOLETÍNS"

Columna	Inst_boletín	Fecha_pub	Nombre_boletín	Autor_boletín	Id_boletín
Tipo llave					PK
Nulo/ Único	NN	NN	NN	NN	NN
Ejemplos	UNAM	16-ene-200	Nuevas especies	Dr. Artemio Gallegos	b001
	IPN	28-feb-1999	Moluscos en aguas profundas	Ana Laura Flores Torres	b002
	UNAM	116-abril-1988	Últimos gastropodos descubiertos	Dra. Andrea Rivas Gallardo	b003

### Nombre de la Tabla "SUBORDENS"

Columna	Nmbre_suborden	Id_suborden
Tipo llave		PK
Nulo/ Único	NN	NN
Ejemplos	Myina	so001
	Pholadina	so002
	Euthecosomata	So003

### Nombre de la Tabla "GÉNEROS"

Columna	Fcha_género	Dscubrdor_género	Id_género	Nmbre_género
Tipo llave			PK	
Nulo/ Único	NN	NN	NN	NN
Ejemplos	01-ene-1818	Lamarck	G001	Solemya
	10-ene-1799	Lemark	G002	Nucula
	01-ene-1807	Link	G003	Nuculana

### Nombre de la Tabla "SUPERFAMILIAS"

Tipo llave		PK
Nulo/Único	NN	NN
Ejemplos	Solemyacea	sf001
	Nucolacea	sf002
	Nuculonacea	sf003

Nombre de la Tabla “REGIÓN”

Columna	Nmbre_región	latitud	Clave_región	longitud	profundidad	Id_región	salinidad
Tipo llave						PK	
Nulo/Único	NN	NN	NN	NN	NN	NN	NN
Ejemplos	Campeche	15m	xx	20m	150m	R001	35
	Manzanillo	8m	Xx	20m	10m	R002	35
	Isla Mujeres	45m	xx	54m	32m	R003	35

Columna	hora	Ambiente_terrestre	Altitud	Dr_id_cr	Paí_id_pais
Tipo llave				PF	PF
Nulo/Único	NN	NN	NN	NN	NN
Ejemplos	3:00pm	Lagunas costeras	20m	Dr001	p001
	11:00am	Arenoso	21m	Dr002	p002
	8:00pm	rocas	20m	Dr003	P003

Nombre de la Tabla

“PAÍSES”

Columna	Id_pais	Ciudad_pais	Nombre_pais	Estadodistrito
Tipo llave	PK			
Nulo/Único	NN	NN	NN	NN
Ejemplos	P001	Sin dato	México	Guerrero
	P002	Sin dato	México	Colima
	P003	Sin dato	México	La paz

Nombre de la Tabla “DET\_REGIÓN”

Columna	Id_cr	comntario	Mol_id_molusco
Tipo llave			PF
Nulo/Único	NN	NN	NN
Ejemplos	Dr001	Sin comentario	m001
	Dr002	Sin comentario	m002
	Dr003	Sin comentario	m003

Nombre de la Tabla “SUBFAMILIAS”

Columna	Id_subfamilia	nmbre_subfamilia
Tipo llave	PK	
Nulo/Único	NN	NN
Ejemplos	Subf001	Monodontinae
	Subf002	Patellinae
	Subf003	Fssurellinae



Nombre de la Tabla "DET\_INVS"

Columna	Comentario_inv	Id_ci	Mol_id_molusco	Inv_id_investigador
Tipo llave		PK	PF	PF
Nulo/ Único	NN	NN	NN	NN
Ejemplos	Sin com	Di001	M001	i001
	Sin com	Di002	M002	i002
	Sin com	Di003	M003	I003

## Creación de Tablas

```
PROMPT Creating Table 'RECOLECTORS'
CREATE TABLE RECOLECTORS
(ID_RECOLECTOR VARCHAR2(20) NOT NULL
,CORREO_E VARCHAR2(50) NOT NULL
,CD_INST VARCHAR2(50) NOT NULL
,NUM_EJEMPLARES INTEGER NOT NULL
,NMBRE_INST VARCHAR2(80) NOT NULL
,TELÉFONO VARCHAR2(15) NOT NULL
,GRADO_CURADOR VARCHAR2(20) NOT NULL
,NMBRE_CURADOR VARCHAR2(50) NOT NULL
)
/
```

```
PROMPT Creating Table 'LIBROS'
CREATE TABLE LIBROS
(URL VARCHAR2(100) NOT NULL
,FECHA_CONSLTA DATE NOT NULL
,EDITORIAL VARCHAR2(20) NOT NULL
,AUTOR_LIBRO VARCHAR2(80) NOT NULL
,TITULO VARCHAR2(50) NOT NULL
,AÑO_PUB DATE NOT NULL
,GRADO_LIB VARCHAR2(20) NOT NULL
,ID_LIBRO VARCHAR2(20) NOT NULL
,ISBN VARCHAR2(50) NOT NULL
,OBSERVACIONES VARCHAR2(256) NOT NULL
,LUGAR VARCHAR2(30) NOT NULL
,ISSN VARCHAR2(50) NOT NULL
)
/
```

```
PROMPT Creating Table 'CATÁLOGOS'
CREATE TABLE CATÁLOGOS
(ID_CATÁLOGO VARCHAR2(20) NOT NULL
,NMBRE_CATÁLOGO VARCHAR2(70) NOT NULL
,NMBRE_COLEC VARCHAR2(70) NOT NULL
)
/
```

```
PROMPT Creating Table 'REF_IMPRESAS'
CREATE TABLE REF_IMPRESAS
(CMENT_RI VARCHAR2(20) NOT NULL
,ID_RI VARCHAR2(20) NOT NULL
,FOL_ID_FOLLETO VARCHAR2(20) NOT NULL
,BOL_ID_BOLETÍN VARCHAR2(20) NOT NULL
,CAT_ID_CATÁLOGO VARCHAR2(20) NOT NULL
,LIB_ID_LIBRO VARCHAR2(20) NOT NULL
)
/
```

```
PROMPT Creating Table 'INVESTIGADORS'
CREATE TABLE INVESTIGADORS
(GRADO_ACADÉMICO VARCHAR2(30) NOT NULL
,DIRECCIÓN VARCHAR2(150) NOT NULL
,NMBRE_INSTITUCIÓN VARCHAR2(80) NOT NULL
,TEL INTEGER NOT NULL
,C_ELECT VARCHAR2(30) NOT NULL
,ID_INVESTIGADOR VARCHAR2(20) NOT NULL
,NMBRE_INVES VARCHAR2(50) NOT NULL
)
/
```

```
PROMPT Creating Table 'FAMILIAS'
CREATE TABLE FAMILIAS
(ID_FAMILIA VARCHAR2(20) NOT NULL
,NMBRE_FAM VARCHAR2(20) NOT NULL
)
/
```

```
PROMPT Creating Table 'HÁBITATS'  
CREATE TABLE HÁBITATS  
(MICRHÁBITAT VARCHAR2(256) NOT NULL  
,AMBIENTE VARCHAR2(256) NOT NULL  
,ID_HÁBITAT VARCHAR2(20) NOT NULL  
,SUSTRATO VARCHAR2(256) NOT NULL  
,TIPO_CUERPO VARCHAR2(256) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'FOLLETOS'  
CREATE TABLE FOLLETOS  
(ID_FOLLETO VARCHAR2(20) NOT NULL  
,AUTOR_FOLLETO VARCHAR2(50) NOT NULL  
,NMBRE_FOLLETO VARCHAR2(50) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'CLASES'  
CREATE TABLE CLASES  
(CLA_ID NUMBER(10,0) NOT NULL  
,NMBRE_CLASE VARCHAR2(240) NOT NULL  
,ID_CLASE VARCHAR2(240) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'SUBGÉNEROS'  
CREATE TABLE SUBGÉNEROS  
(NMBRE_SUBGÉNERO VARCHAR2(20) NOT NULL  
,ID_SUBGÉENRO VARCHAR2(20) NOT NULL  
,FCHA_SUBGÉNERO DATE NOT NULL  
)  
/
```

```
PROMPT Creating Table 'DONADORS'  
CREATE TABLE DONADORS  
(NUM_EJEMPLARES INTEGER NOT NULL  
,ID_DONADOR VARCHAR2(20) NOT NULL  
,INT_DNADORA VARCHAR2(80) NOT NULL  
,NMBRE_DNADOR VARCHAR2(60) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'DET_HÁBITATS'  
CREATE TABLE DET_HÁBITATS  
(ID_CH VARCHAR2(20) NOT NULL  
,COMNTARIO_HÁBITAT VARCHAR2(100) NOT NULL  
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL  
,HÁB_ID_HÁBITAT VARCHAR2(20) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'ORDENS'  
CREATE TABLE ORDENS  
(NMBRE_ORDEN VARCHAR2(20) NOT NULL  
,ID_ORDEN VARCHAR2(20) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'DET_RECOLECS'  
CREATE TABLE DET_RECOLECS  
(COMNTARIO_RECOLEC VARCHAR2(100) NOT NULL  
,ID_REC VARCHAR2(20) NOT NULL  
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL  
,REC_ID_RECOLECTOR VARCHAR2(20) NOT NULL  
)  
/
```

```
PROMPT Creating Table 'MOLUSCOS'  
CREATE TABLE MOLUSCOS  
(NUM_CANABIO VARCHAR2(10) NOT NULL  
,ÁPICE VARCHAR2(256) NOT NULL  
,VUELTA_CORPORAL VARCHAR2(256) NOT NULL  
,EDAD VARCHAR2(7) NOT NULL  
,ID_MOLUSCO VARCHAR2(20) NOT NULL  
,ORNAMENTADO VARCHAR2(256) NOT NULL
```

```

,TAMAÑO NUMBER NOT NULL
,POSTULAS VARCHAR2(256) NOT NULL
,GIRO_CONCHA VARCHAR2(256) NOT NULL
,FOTOGRAFIA VARCHAR2(20) NOT NULL
,TIPO_CANAL VARCHAR2(256) NOT NULL
,TIPO_COSTILLA VARCHAR2(256) NOT NULL
,LABIOS VARCHAR2(256) NOT NULL
,FRMA_CONCHA VARCHAR2(20) NOT NULL
,SEXO VARCHAR2(2) NOT NULL
,NUM_VUELTAS VARCHAR2(256) NOT NULL
,ESPINAS VARCHAR2(256) NOT NULL
,NT_ID_NOM_TAX VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'ESPECIES'
CREATE TABLE ESPECIES
(FCHA_ESPECIE DATE NOT NULL
,DSCUBRDOR_ESPECIE VARCHAR2(50) NOT NULL
,ID_ESPECIE VARCHAR2(20) NOT NULL
,NMBRE_ESPECIE VARCHAR2(40) NOT NULL
)
/

```

```

PROMPT Creating Table 'SUBCLASES'
CREATE TABLE SUBCLASES
(NMBRE_CUBCLASE VARCHAR2(20) NOT NULL
,ID_SUBCLASE VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'NOMBRE_TAXONÓMICOS'
CREATE TABLE NOMBRE_TAXONÓMICOS
(NMBRE_COMÚN VARCHAR2(20) NOT NULL
,ID_NOM_TAX VARCHAR2(20) NOT NULL
,SUB3_ID_SUBCLASE VARCHAR2(20) NOT NULL
,CLA_CLA_ID NUMBER(10,0) NOT NULL
,GEN_ID_GÉNERO VARCHAR2(20) NOT NULL
,SUP_ID_SUPERFAMILIA VARCHAR2(20) NOT NULL
,SUB2_ID_SUBORDEN VARCHAR2(20) NOT NULL
,SUB_ID_SUBGÉENRO VARCHAR2(20) NOT NULL
,FAM_ID_FAMILIA VARCHAR2(20) NOT NULL
,ESP_ID_ESPECIE VARCHAR2(20) NOT NULL
,ORD_ID_ORDEN VARCHAR2(20) NOT NULL
,SUB1_ID_SUBFAMILIA VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'DET_REFS'
CREATE TABLE DET_REFS
(COMNTARIO_REF VARCHAR2(100) NOT NULL
,ID_R VARCHAR2(20) NOT NULL
,RI_ID_RI VARCHAR2(20) NOT NULL
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'DET_DNADORS'
CREATE TABLE DET_DNADORS
(ID_CD VARCHAR2(20) NOT NULL
,COMNTARIO_DNADOR VARCHAR2(100) NOT NULL
,DON_ID_DONADOR VARCHAR2(20) NOT NULL
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'BOLETÍNS'
CREATE TABLE BOLETÍNS
(INST_BOLETÍN VARCHAR2(80) NOT NULL
,FECHA_PUB DATE NOT NULL
,NOMBRE_BOLETÍN VARCHAR2(50) NOT NULL
,AUTOR_BOLETÍN VARCHAR2(50) NOT NULL
,ID_BOLETÍN VARCHAR2(20) NOT NULL
)
/

```

```

PROMPT Creating Table 'SUBORDENS'

```

```

CREATE TABLE SUBORDENS
(NMBRE_SUBORDEN VARCHAR2(20) NOT NULL
,ID_SUBORDEN VARCHAR2(20) NOT NULL
)
/

PROMPT Creating Table 'GÉNEROS'
CREATE TABLE GÉNEROS
(FCHA_GÉNERO DATE NOT NULL
,DSCUBRDOR_GÉNERO VARCHAR2(50) NOT NULL
,ID_GÉNERO VARCHAR2(20) NOT NULL
,NMBRE_GÉNERO VARCHAR2(15) NOT NULL
)
/

PROMPT Creating Table 'SUPERFAMILIAS'
CREATE TABLE SUPERFAMILIAS
(NMBRE_SUPERFAMILIA VARCHAR2(20) NOT NULL
,ID_SUPERFAMILIA VARCHAR2(20) NOT NULL
)
/

PROMPT Creating Table 'REGIÓN'S'
CREATE TABLE REGIÓN'S
(NMBRE_REGIÓN VARCHAR2(50) NOT NULL
,LATITUD VARCHAR2(6) NOT NULL
,CLAVE_REGIÓN VARCHAR2(12) NOT NULL
,LONGITUD VARCHAR2(6) NOT NULL
,PROFUNDIDAD VARCHAR2(10) NOT NULL
,ID_REGIÓN VARCHAR2(20) NOT NULL
,SALINIDAD VARCHAR2(10) NOT NULL
,HORA DATE NOT NULL
,AMBIENTE__TERRESTRE VARCHAR2(256) NOT NULL
,ALTITUD VARCHAR2(6) NOT NULL
,DR_ID_CR VARCHAR2(20) NOT NULL
,PAÍ_ID_PAÍS VARCHAR2(20) NOT NULL
)
/

PROMPT Creating Table 'PAÍSES'
CREATE TABLE PAÍSES
(ID_PAÍS VARCHAR2(20) NOT NULL
,CIUDAD_PAÍS VARCHAR2(256) NOT NULL
,NOMBRE_PAÍS VARCHAR2(30) NOT NULL
,ESTADODISTRITO VARCHAR2(30) NOT NULL
)
/

PROMPT Creating Table 'DET_REGIÓN'S'
CREATE TABLE DET_REGIÓN'S
(ID_CR VARCHAR2(20) NOT NULL
,COMNTARIO_REGIÓN VARCHAR2(100) NOT NULL
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL
)
/

PROMPT Creating Table 'SUBFAMILIAS'
CREATE TABLE SUBFAMILIAS
(ID_SUBFAMILIA VARCHAR2(20) NOT NULL
,NMBRE_SUBFAMILIA VARCHAR2(20) NOT NULL
)
/

PROMPT Creating Table 'DET_INVS'
CREATE TABLE DET_INVS
(COMNTARIO_INV VARCHAR2(100) NOT NULL
,ID_CI VARCHAR2(20) NOT NULL
,MOL_ID_MOLUSCO VARCHAR2(20) NOT NULL
,INV_ID_INVESTIGADOR VARCHAR2(20) NOT NULL
)
/

```

## Creación de Llaves Primarias y Foráneas

```
PROMPT Creating Primary Key on 'RECOLECTORS'  
ALTER TABLE RECOLECTORS  
ADD (CONSTRAINT REC_PK PRIMARY KEY  
(ID_RECOLECTOR))  
/
```

```
PROMPT Creating Primary Key on 'LIBROS'  
ALTER TABLE LIBROS  
ADD (CONSTRAINT LIB_PK PRIMARY KEY  
(ID_LIBRO))  
/
```

```
PROMPT Creating Primary Key on 'CATÁLOGOS'  
ALTER TABLE CATÁLOGOS  
ADD (CONSTRAINT CAT_PK PRIMARY KEY  
(ID_CATÁLOGO))  
/
```

```
PROMPT Creating Primary Key on 'REF_IMPRESAS'  
ALTER TABLE REF_IMPRESAS  
ADD (CONSTRAINT RI_PK PRIMARY KEY  
(ID_RI))  
/
```

```
PROMPT Creating Primary Key on 'INVESTIGADORES'  
ALTER TABLE INVESTIGADORES  
ADD (CONSTRAINT INV_PK PRIMARY KEY  
(ID_INVESTIGADOR))  
/
```

```
PROMPT Creating Primary Key on 'FAMILIAS'  
ALTER TABLE FAMILIAS  
ADD (CONSTRAINT FAM_PK PRIMARY KEY  
(ID_FAMILIA))  
/
```

```
PROMPT Creating Primary Key on 'HÁBITATS'  
ALTER TABLE HÁBITATS  
ADD (CONSTRAINT HÁB_PK PRIMARY KEY  
(ID_HÁBITAT))  
/
```

```
PROMPT Creating Primary Key on 'FOLLETOS'  
ALTER TABLE FOLLETOS  
ADD (CONSTRAINT FOL_PK PRIMARY KEY  
(ID_FOLLETO))  
/
```

```
PROMPT Creating Primary Key on 'CLASES'  
ALTER TABLE CLASES  
ADD (CONSTRAINT CLA_PK PRIMARY KEY  
(CLA_ID))  
/
```

```
PROMPT Creating Primary Key on 'SUBGÉNEROS'  
ALTER TABLE SUBGÉNEROS  
ADD (CONSTRAINT SUB_PK PRIMARY KEY  
(ID_SUBGÉNERO))  
/
```

```
PROMPT Creating Primary Key on 'DONADORES'  
ALTER TABLE DONADORES  
ADD (CONSTRAINT DON_PK PRIMARY KEY  
(ID_DONADOR))  
/
```

```
PROMPT Creating Primary Key on 'DET_HÁBITATS'  
ALTER TABLE DET_HÁBITATS  
ADD (CONSTRAINT DH_PK PRIMARY KEY  
(ID_CH))  
/
```

```
PROMPT Creating Primary Key on 'ORDENS'  
ALTER TABLE ORDENS  
ADD (CONSTRAINT ORD_PK PRIMARY KEY  
(ID_ORDEN))  
/
```

```
PROMPT Creating Primary Key on 'DET_RECOLECS'  
ALTER TABLE DET_RECOLECS  
ADD (CONSTRAINT DR2_PK PRIMARY KEY  
(ID_REC))  
/
```

```
PROMPT Creating Primary Key on 'MOLUSCOS'  
ALTER TABLE MOLUSCOS  
ADD (CONSTRAINT MOL_PK PRIMARY KEY  
(ID_MOLUSCO))  
/
```

```
PROMPT Creating Primary Key on 'ESPECIES'  
ALTER TABLE ESPECIES  
ADD (CONSTRAINT ESP_PK PRIMARY KEY  
(ID_ESPECIE))  
/
```

```
PROMPT Creating Primary Key on 'SUBCLASES'  
ALTER TABLE SUBCLASES  
ADD (CONSTRAINT SUB3_PK PRIMARY KEY  
(ID_SUBCLASE))  
/
```

```
PROMPT Creating Primary Key on 'NOMBRE_TAXONÓMICOS'  
ALTER TABLE NOMBRE_TAXONÓMICOS  
ADD (CONSTRAINT NT_PK PRIMARY KEY  
(ID_NOM_TAX))  
/
```

```
PROMPT Creating Primary Key on 'DET_REFS'  
ALTER TABLE DET_REFS  
ADD (CONSTRAINT DR1_PK PRIMARY KEY  
(ID_R))  
/
```

```
PROMPT Creating Primary Key on 'DET_DNADORS'  
ALTER TABLE DET_DNADORS  
ADD (CONSTRAINT DD_PK PRIMARY KEY  
(ID_CD))  
/
```

```
PROMPT Creating Primary Key on 'BOLETÍNS'  
ALTER TABLE BOLETÍNS  
ADD (CONSTRAINT BOL_PK PRIMARY KEY  
(ID_BOLETÍN))  
/
```

```
PROMPT Creating Primary Key on 'SUBORDENS'  
ALTER TABLE SUBORDENS  
ADD (CONSTRAINT SUB2_PK PRIMARY KEY  
(ID_SUBORDEN))  
/
```

```
PROMPT Creating Primary Key on 'GÉNEROS'  
ALTER TABLE GÉNEROS  
ADD (CONSTRAINT GEN_PK PRIMARY KEY  
(ID_GÉNERO))  
/
```

```
PROMPT Creating Primary Key on 'SUPERFAMILIAS'  
ALTER TABLE SUPERFAMILIAS  
ADD (CONSTRAINT SUP_PK PRIMARY KEY  
(ID_SUPERFAMILIA))  
/
```

```
PROMPT Creating Primary Key on 'REGIÓN'S'  
ALTER TABLE REGIÓN'S  
ADD (CONSTRAINT REG_PK PRIMARY KEY  
(ID_REGIÓN))
```

```

/

PROMPT Creating Primary Key on 'PAÍSES'
ALTER TABLE PAÍSES
ADD (CONSTRAINT PAÍ_PK PRIMARY KEY
(ID_PAÍS))
/

PROMPT Creating Primary Key on 'DET_REGIÓN'S'
ALTER TABLE DET_REGIÓN'S
ADD (CONSTRAINT DR_PK PRIMARY KEY
(ID_CR))
/

PROMPT Creating Primary Key on 'SUBFAMILIAS'
ALTER TABLE SUBFAMILIAS
ADD (CONSTRAINT SUB1_PK PRIMARY KEY
(ID_SUBFAMILIA))
/

PROMPT Creating Primary Key on 'DET_INVS'
ALTER TABLE DET_INVS
ADD (CONSTRAINT DI_PK PRIMARY KEY
(ID_CI))
/

PROMPT Creating Foreign Key on 'REF_IMPRESAS'
ALTER TABLE REF_IMPRESAS ADD (CONSTRAINT
RI_CAT_FK FOREIGN KEY
(CAT_ID_CATÁLOGO) REFERENCES CATÁLOGOS
(ID_CATÁLOGO))
/

PROMPT Creating Foreign Key on 'REF_IMPRESAS'
ALTER TABLE REF_IMPRESAS ADD (CONSTRAINT
RI_BOL_FK FOREIGN KEY
(BOL_ID_BOLETÍN) REFERENCES BOLETÍNS
(ID_BOLETÍN))
/

PROMPT Creating Foreign Key on 'REF_IMPRESAS'
ALTER TABLE REF_IMPRESAS ADD (CONSTRAINT
RI_LIB_FK FOREIGN KEY
(LIB_ID_LIBRO) REFERENCES LIBROS
(ID_LIBRO))
/

PROMPT Creating Foreign Key on 'REF_IMPRESAS'
ALTER TABLE REF_IMPRESAS ADD (CONSTRAINT
RI_FOL_FK FOREIGN KEY
(FOL_ID_FOLLETO) REFERENCES FOLLETOS
(ID_FOLLETO))
/

PROMPT Creating Foreign Key on 'DET_HÁBITATS'
ALTER TABLE DET_HÁBITATS ADD (CONSTRAINT
DH_MOL_FK FOREIGN KEY
(MOL_ID_MOLUSCO) REFERENCES MOLUSCOS
(ID_MOLUSCO))
/

PROMPT Creating Foreign Key on 'DET_HÁBITATS'
ALTER TABLE DET_HÁBITATS ADD (CONSTRAINT
DH_HÁB_FK FOREIGN KEY
(HÁB_ID_HÁBITAT) REFERENCES HÁBITATS
(ID_HÁBITAT))
/

PROMPT Creating Foreign Key on 'DET_RECOLECS'
ALTER TABLE DET_RECOLECS ADD (CONSTRAINT
DR2_MOL_FK FOREIGN KEY
(MOL_ID_MOLUSCO) REFERENCES MOLUSCOS
(ID_MOLUSCO))
/

```



```
PROMPT Creating Foreign Key on 'DET_RECOLECS'  
ALTER TABLE DET_RECOLECS ADD (CONSTRAINT  
DR2_REC_FK FOREIGN KEY  
(REC_ID_RECOLECTOR) REFERENCES RECOLECTORS  
(ID_RECOLECTOR))  
/
```

```
PROMPT Creating Foreign Key on 'MOLUSCOS'  
ALTER TABLE MOLUSCOS ADD (CONSTRAINT  
MOL_NT_FK FOREIGN KEY  
(NT_ID_NOM_TAX) REFERENCES NOMBRE_TAXONOMICOS  
(ID_NOM_TAX))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_ESP_FK FOREIGN KEY  
(ESP_ID_ESPECIE) REFERENCES ESPECIES  
(ID_ESPECIE))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_SUB1_FK FOREIGN KEY  
(SUB1_ID_SUBFAMILIA) REFERENCES SUBFAMILIAS  
(ID_SUBFAMILIA))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_ORD_FK FOREIGN KEY  
(ORD_ID_ORDEN) REFERENCES ORDENS  
(ID_ORDEN))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_SUB3_FK FOREIGN KEY  
(SUB3_ID_SUBCLASE) REFERENCES SUBCLASES  
(ID_SUBCLASE))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_GEN_FK FOREIGN KEY  
(GEN_ID_GENERO) REFERENCES GENEROS  
(ID_GENERO))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_CLA_FK FOREIGN KEY  
(CLA_CLA_ID) REFERENCES CLASES  
(CLA_ID))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_FAM_FK FOREIGN KEY  
(FAM_ID_FAMILIA) REFERENCES FAMILIAS  
(ID_FAMILIA))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_SUB2_FK FOREIGN KEY  
(SUB2_ID_SUBORDEN) REFERENCES SUBORDENS  
(ID_SUBORDEN))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONOMICOS'  
ALTER TABLE NOMBRE_TAXONOMICOS ADD (CONSTRAINT  
NT_SUB_FK FOREIGN KEY  
(SUB_ID_SUBGENERO) REFERENCES SUBGENEROS  
(ID_SUBGENERO))  
/
```

```
PROMPT Creating Foreign Key on 'NOMBRE_TAXONÓMICOS'  
ALTER TABLE NOMBRE_TAXONÓMICOS ADD (CONSTRAINT  
NT_SUP_FK FOREIGN KEY  
(SUP_ID_SUPERFAMILIA) REFERENCES SUPERFAMILIAS  
(ID_SUPERFAMILIA))  
/
```

```
PROMPT Creating Foreign Key on 'DET_REFS'  
ALTER TABLE DET_REFS ADD (CONSTRAINT  
DR1_MOL_FK FOREIGN KEY  
(MOL_ID_MOLUSCO) REFERENCES MOLUSCOS  
(ID_MOLUSCO))  
/
```

```
PROMPT Creating Foreign Key on 'DET_REFS'  
ALTER TABLE DET_REFS ADD (CONSTRAINT  
DR1_RI_FK FOREIGN KEY  
(RI_ID_RI) REFERENCES REF_IMPRESAS  
(ID_RI))  
/
```

```
PROMPT Creating Foreign Key on 'DET_DNADORS'  
ALTER TABLE DET_DNADORS ADD (CONSTRAINT  
DD_MOL_FK FOREIGN KEY  
(MOL_ID_MOLUSCO) REFERENCES MOLUSCOS  
(ID_MOLUSCO))  
/
```

```
PROMPT Creating Foreign Key on 'DET_DNADORS'  
ALTER TABLE DET_DNADORS ADD (CONSTRAINT  
DD_DON_FK FOREIGN KEY  
(DON_ID_DONADOR) REFERENCES DONADORS  
(ID_DONADOR))  
/
```

```
PROMPT Creating Foreign Key on 'REGIÓN'S'  
ALTER TABLE REGIÓN'S ADD (CONSTRAINT  
REG_PAÍ_FK FOREIGN KEY  
(PAÍ_ID_PAÍS) REFERENCES PAÍSES  
(ID_PAÍS))  
/
```

```
PROMPT Creating Foreign Key on 'REGIÓN'S'  
ALTER TABLE REGIÓN'S ADD (CONSTRAINT  
REG_DR_FK FOREIGN KEY  
(DR_ID_CR) REFERENCES DET_REGIÓN'S  
(ID_CR))  
/
```

```
PROMPT Creating Foreign Key on 'DET_REGIÓN'S'  
ALTER TABLE DET_REGIÓN'S ADD (CONSTRAINT  
DR_MOL_FK FOREIGN KEY  
(MOL_ID_MOLUSCO) REFERENCES MOLUSCOS  
(ID_MOLUSCO))  
/
```

```
PROMPT Creating Foreign Key on 'DET_INVS'  
ALTER TABLE DET_INVS ADD (CONSTRAINT  
DI_INV_FK FOREIGN KEY  
(INV_ID_INVESTIGADOR) REFERENCES INVESTIGADORS  
(ID_INVESTIGADOR))  
/
```

## Creación de Índices

```
PROMPT Creating Index 'RI_BOL_FK_I'  
CREATE INDEX RI_BOL_FK_I ON REF_IMPRESAS  
(BOL_ID_BOLETÍN)  
/
```

```

PROMPT Creating Index 'RI_LIB_FK_I'
CREATE INDEX RI_LIB_FK_I ON REF_IMPRESAS
(LIB_ID_LIBRO)
/

PROMPT Creating Index 'RI_FOL_FK_I'
CREATE INDEX RI_FOL_FK_I ON REF_IMPRESAS
(FOL_ID_FOLLETO)
/

PROMPT Creating Index 'RI_CAT_FK_I'
CREATE INDEX RI_CAT_FK_I ON REF_IMPRESAS
(CAT_ID_CATALOGO)
/

PROMPT Creating Index 'DH_HAB_FK_I'
CREATE INDEX DH_HAB_FK_I ON DET_HABITATS
(HAB_ID_HABITAT)
/

PROMPT Creating Index 'DH_MOL_FK_I'
CREATE INDEX DH_MOL_FK_I ON DET_HABITATS
(MOL_ID_MOLUSCO)
/

PROMPT Creating Index 'DR2_MOL_FK_I'
CREATE INDEX DR2_MOL_FK_I ON DET_RECOLECS
(MOL_ID_MOLUSCO)
/

PROMPT Creating Index 'DR2_REC_FK_I'
CREATE INDEX DR2_REC_FK_I ON DET_RECOLECS
(REC_ID_RECOLECTOR)
/

PROMPT Creating Index 'MOL_NT_FK_I'
CREATE INDEX MOL_NT_FK_I ON MOLUSCOS
(NT_ID_NOM_TAX)
/

PROMPT Creating Index 'NT_SUB2_FK_I'
CREATE INDEX NT_SUB2_FK_I ON NOMBRE_TAXONOMICOS
(SUB2_ID_SUBORDEN)
/

PROMPT Creating Index 'NT_GEN_FK_I'
CREATE INDEX NT_GEN_FK_I ON NOMBRE_TAXONOMICOS
(GEN_ID_GENERO)
/

PROMPT Creating Index 'NT_SUP_FK_I'
CREATE INDEX NT_SUP_FK_I ON NOMBRE_TAXONOMICOS
(SUP_ID_SUPERFAMILIA)
/

PROMPT Creating Index 'NT_SUB1_FK_I'
CREATE INDEX NT_SUB1_FK_I ON NOMBRE_TAXONOMICOS
(SUB1_ID_SUBFAMILIA)
/

PROMPT Creating Index 'NT_CLA_FK_I'
CREATE INDEX NT_CLA_FK_I ON NOMBRE_TAXONOMICOS
(CLA_CLA_ID)
/

PROMPT Creating Index 'NT_SUB_FK_I'
CREATE INDEX NT_SUB_FK_I ON NOMBRE_TAXONOMICOS
(SUB_ID_SUBGENERO)
/

PROMPT Creating Index 'NT_FAM_FK_I'
CREATE INDEX NT_FAM_FK_I ON NOMBRE_TAXONOMICOS
(FAM_ID_FAMILIA)
/

PROMPT Creating Index 'NT_ESP_FK_I'
CREATE INDEX NT_ESP_FK_I ON NOMBRE_TAXONOMICOS

```

```
(ESP_ID_ESPECIE)
```

```
/
```

```
PROMPT Creating Index 'NT_SUB3_FK_I'  
CREATE INDEX NT_SUB3_FK_I ON NOMBRE_TAXONÓMICOS  
(SUB3_ID_SUBCLASE)
```

```
/
```

```
PROMPT Creating Index 'NT_ORD_FK_I'  
CREATE INDEX NT_ORD_FK_I ON NOMBRE_TAXONÓMICOS  
(ORD_ID_ORDEN)
```

```
/
```

```
PROMPT Creating Index 'DR1_MOL_FK_I'  
CREATE INDEX DR1_MOL_FK_I ON DET_REFS  
(MOL_ID_MOLUSCO)
```

```
/
```

```
PROMPT Creating Index 'DR1_RI_FK_I'  
CREATE INDEX DR1_RI_FK_I ON DET_REFS  
(RI_ID_RI)
```

```
/
```

```
PROMPT Creating Index 'DD_DON_FK_I'  
CREATE INDEX DD_DON_FK_I ON DET_DNADORS  
(DON_ID_DONADOR)
```

```
/
```

```
PROMPT Creating Index 'DD_MOL_FK_I'  
CREATE INDEX DD_MOL_FK_I ON DET_DNADORS  
(MOL_ID_MOLUSCO)
```

```
/
```

```
PROMPT Creating Index 'REG_PAÍ_FK_I'  
CREATE INDEX REG_PAÍ_FK_I ON REGIÓNES  
(PAÍ_ID_PAÍS)
```

```
/
```

```
PROMPT Creating Index 'REG_DR_FK_I'  
CREATE INDEX REG_DR_FK_I ON REGIÓNES  
(DR_ID_CR)
```

```
/
```

```
PROMPT Creating Index 'DR_MOL_FK_I'  
CREATE INDEX DR_MOL_FK_I ON DET_REGIONES  
(MOL_ID_MOLUSCO)
```

```
/
```

```
PROMPT Creating Index 'DI_INV_FK_I'  
CREATE INDEX DI_INV_FK_I ON DET_INVS  
(INV_ID_INVESTIGADOR)
```

```
/
```

```
PROMPT Creating Index 'DI_MOL_FK_I'  
CREATE INDEX DI_MOL_FK_I ON DET_INVS  
(MOL_ID_MOLUSCO)
```

```
/
```

## Inserción de Información

```
PROMPT RECOLECTORS
```

```
insert into recolectors values(  
'r001','ajus@yahoo.com','México',15,'UNAM','54 15 20 33','Técnico','Ajustín Rojas Paredes'  
);
```

```
insert into recolectors values(  
'r002','jacob36@hotmail.com','México',5,'UAM','15 23 44 78','Doctor','Jacobó Bolaños Ríos'  
);
```

```
insert into recolectors values(  
'r003','angela55@yahoo.com.mx','México',30,'UNAM','52 22 13 55','Bióloga','Angeles Prado Uribe'  
);
```

--LIBROS

```
insert into LIBROS values(
'sin url','12-sep-1992','Prentice Hall','Antonio García Cubas','Gastropodos y más','01-ene-1990','maestria','I001','xx','sin observaciones','
México D.F','xx'
);
```

```
insert into LIBROS values(
'sin url','25-mar-2005','Grijalbo','Esteban González Flores ','Cephalopodos','01-ene-2004','licenciatura','I002','xx','sin observaciones',
'México D.F','xx'
);
```

```
insert into LIBROS values(
'SIN URL','08-nov-2005','UNAM','Martha Reguero','Catálogo de Gasterópodos del golfo de México','01-mar-2004','doctorado','I003','xx','sin
observaciones','
México D.F','xx'
);
```

--CATÁLOGOS

```
insert into catálogos values(
'c001','Catálogo de moluscos del pacifico mexicano','Bivalbios'
);
```

```
insert into catálogos values(
'c002','Catálogo ilustrado sobre Cephalopodos','moluscos'
);
```

```
insert into catálogos values(
'c003','Catálogo de moluscos del norte de América','polyplacophora'
);
```

--REF\_IMPRESAS

```
insert into ref_impresas values(
'sin comentario','ri001','fo001','b001','c001','I001'
);
```

```
insert into ref_impresas values(
'sin comentario','ri002','fo002','b002','c002','I002'
);
```

```
insert into ref_impresas values(
'sin comentario','ri003','fo003','b003','c003','I003'
);
```

--INVESTIGADORES

```
insert into investigadores values(
'Doctor','circuito escolar s/n cd. universitaria','UNAM','22563094','reguero@unam.mx','i001','Martha Reguero'
);
```

```
insert into investigadores values(
'Doctor','circuito escolar s/n cd. universitaria','UNAM','22563093','steven@unam.mx','i002','Steven Czitrom'
);
```

```
insert into investigadores values(
'biologo','circuito escolar s/n cd. universitaria','UNAM','22561594','raymundo@unam.mx','i003','Raymundo Rodriguez'
);
```

--FAMILIAS

```
insert into familias values(
'f001','Caecidae'
);
```

```
insert into familias values(
'f002','Turritellidae'
);
```

);

```
insert into familias values(  
'f003','Architeconicidae'  
);
```

#### --HÁBITATS

```
insert into hábitats values(  
'paztisiales','aguas someras','h001','arenoso','mar'  
);
```

```
insert into hábitats values(  
'aguas profundas','arena','h002','arenoso','mar'  
);
```

```
insert into hábitats values(  
'en la costa enterradas','aguas someras','h003','poco profundo','mar'  
);
```

#### --FOLLETOS

```
insert into folletos values(  
'fo001','Carlos Dominguez R.','Algo más sobre las especies Gastropodas'  
);
```

```
insert into folletos values(  
'fo002','Enriqueta Urbina','Estudio de la familia Architectonicidae'  
);
```

```
insert into folletos values(  
'fo003','Miriam Cancino','Architectonica'  
);
```

#### --CLASES

```
insert into clases values(  
'001','Bivalvia','c001'  
);
```

```
insert into clases values(  
'002','Cephalopoda','c002'  
);
```

```
insert into clases values(  
'003','Polyplacophora','c003'  
);
```

#### --SUBGENEROS

```
insert into subgéneros values(  
'Acharax','sg001','01-ene-1891'  
);
```

```
insert into subgéneros values(  
'Petrasma','sg002','01-ene-1908'  
);
```

```
insert into subgéneros values(  
'Nucula','sg003','01-ene-1799'  
);
```

#### --DONADORS

```
insert into donadors values(  
'3','d001','UNAM','Dr. Carlos Ortíz V.'  
);
```

```
insert into donadors values(  
'15','d002','UNAM','Esteban Hernández Pinsón'  
);
```

```
insert into donadors values(
'8','d003','UAM','Angelica López'
);
```

#### --DET\_HÁBITATS

```
insert into det_habitats values(
'ch001','sin comentario','m001','h001'
);
```

```
insert into det_habitats values(
'ch002','sin comentario','m002','h002'
);
```

```
insert into det_habitats values(
'ch003','sin comentario','m003','h003'
);
```

#### --ORDENS

```
insert into ordens values(
'Solemya johnsoni','o001'
);
```

```
insert into ordens values(
'Solemya panamesis','o002'
);
```

```
insert into ordens values(
'Solemya valvulus','o003'
);
```

#### --DET\_RECOLECS

```
insert into det_recolect values(
'sin comen','dr001','m001','r001'
);
```

```
insert into det_recolect values(
'sin comen','dr002','m002','r002'
);
```

```
insert into det_recolect values(
'sin comen','dr003','m003','r03'
);
```

#### --MOLUSCOS

```
insert into moluscos values(
'sin num','no descrito','cinco osiete vueltas vueltas convexas delgadas y resistentes','1 año','m001','no descrito','2','postnucleares','muestran
lineas tenues axiales ampliamente esparcidas','sin imagen','sifonal','ornamentado','labio interno muestra un doblez','aovada cónica','f','7','no
descrito','nt001'
);
```

```
insert into moluscos values(
'sin num','desplazado un poco hacia el frente de la parte media','no descrito','3 meses','m002','costillas radiales','20 ','no descrito','forma oval
despirbaja','sin imagen','no descrito','ornamentado','no descrito','forma oval','f','3','no descrito','nt002'
);
```

```
insert into moluscos values(
'sin num','no descrito','no descrito','sr','m003','ornamentada cada segunda costilla es más grande','30','no descrito','no descrito','sin
imagen','anal','no descrito','ornamentado','forma cónica','m','6','sin espinas','nt003'
);
```

#### --ESPECIES

```
insert into especies values(
'01-ene-1891','Dall','e001','Solemya johnsoni'
);
```

```
insert into especies values(
'01-ene-1908','Dall','e002','Solemya panamesis'
);
```

```

);

insert into especies values(
'01-ene-1864','Carpenter','e003','Solemya valvulus'
);

--SUBCLASES

insert into subclasses values(
'Myina','sc001'
);

insert into subclasses values(
'Pholadina','sc002'
);

insert into subclasses values(
'Euthecosomata','sc003'
);

--NOMBRE_TAXONÓMICOS

insert into nombre_taxonómicos values(
'Johnsoni','nt001','sc001','001','g001','sf001','so001','sg001','f001','e001','o001','subf001'
);

insert into nombre_taxonómicos values(
'Panamensis','nt002','sc002','002','g002','sf002','so002','sg002','f002','e002','o002','subf002'
);

insert into nombre_taxonómicos values(
'Valvulus','nt003','sc003','003','g003','sf003','so003','sg003','f003','e003','o003','subf003'
);

--DET_REFS

insert into det_refs values(
'sin comentario','dr001','ri001','m001'
);

insert into det_refs values(
'sin comentario','dr002','ri002','m002'
);

insert into det_refs values(
'sin comentario','dr003','ri003','m003'
);

--DET_DNADORS

insert into det_dnadors values(
'dd001','sin comentario','d001','m001'
);

insert into det_dnadors values(
'dd002','sin comentario','d002','m002'
);

insert into det_dnadors values(
'dd003','sin comentario','d003','m003'
);

--BOLETÍNS

insert into boletíns values(
'UNAM','16-ene-2000','Nuevas especies','Dr. Artemio Gallegos','b001'
);

insert into boletíns values(
'IPN','28-feb-1999','Moluscos en aguas profundas','Ana Laura Flores Torres','b002'
);

insert into boletíns values(

```



```
'UNAM','16-abril-1988','Últimos gastropodos descubiertos','Dra. Andrea Rivas Gallardo','b003'  
);
```

#### --SUBORDENS

```
insert into subordens values(  
'Myina','so001'  
);
```

```
insert into subordens values(  
'Pholadina','so002'  
);
```

```
insert into subordens values(  
'Euthecosomata','so003'  
);
```

#### --GENÉROS

```
insert into géneros values(  
'01-ene-1818','Lamarck','g001','Solemya'  
);
```

```
insert into géneros values(  
'10-ene-1799','Lemark','g002','Nucula'  
);
```

```
insert into géneros values(  
'01-ene-1807','Link','g003','Nuculana'  
);
```

#### --SUPERFAMILIAS

```
insert into superfamilias values(  
'Solemyacea','sf001'  
);
```

```
insert into superfamilias values(  
'Nucolacea','sf002'  
);
```

```
insert into superfamilias values(  
'Nuculonacea','sf003'  
);
```

#### --REGIÓN

```
insert into regiones values(  
'Campeche','15m','xx','20m','150m','r001','35',to_date('2000/05/31:03:00:00PM','yyyy/mm/dd:hh:mi:ssam'),'lagunas  
costeras','20m','dr001','p001'  
);
```

```
insert into regiones values(  
'Manzanillo','8m','xx','20m','10m','r002','35',to_date('1998/05/18:11:00:00AM','yyyy/mm/dd:hh:mi:ssam'),'arenoso','21m','dr002','p002'  
);
```

```
insert into regiones values(  
'Isla Mujeres','45m','xx','54m','32m','r003','35',to_date('2005/01/16:08:00:00PM','yyyy/mm/dd:hh:mi:ssam'),'rocas','20m','dr003','p003'  
);
```

#### --PAÍSES

```
insert into países values(  
'p001','sin dato','México','Guerrero'  
);
```

```
insert into países values(  
'p002','sin dato','México','Colima'  
);
```

```
insert into países values(  
'p003','sin dato','México','La Paz'  
);
```

--DET\_REGIÓNS

```
insert into det_regiões values(  
'dr001','sin comentario','m001'  
);
```

```
insert into det_regiões values(  
'dr002','sin comentario','m002'  
);
```

```
insert into det_regiões values(  
'dr003','sin comentario','m003'  
);
```

--SUBFAMILIAS

```
insert into subfamilias values(  
'subf001','Monodontinae'  
);
```

```
insert into subfamilias values(  
'subf002','Patellinae'  
);
```

```
insert into subfamilias values(  
'subf003','Fissurellinae'  
);
```

--DET\_INVS

```
insert into det_invs values(  
'sin com','di001','m001','i001'  
);
```

```
insert into det_invs values(  
'sin com','di002','m002','i002'  
);
```

```
insert into det_invs values(  
'sin com','di003','m003','i003'  
);
```

### BIBLIOGRAFIA

- [Ackeman y Fielding, 1995] Ackerman, M.S., y Fielding, R.T. 1995, "Collection Maintenance in the Digital Library" Digital Libraries'95, The Second Annual Conference on the Theory and Practice of Digital Libraries, Austin, Texas.
- [Argumedo, 1985] Argumedo Lamothe, R., Lázaro-Chávez Mancilla, E. & Meave Gallegos, O., 1985, "Colección Helmintológica del Instituto de Biología", Instituto de Biología, UNAM, México.
- [Batín, 1994] Batín Carlo, Ceri Stefano, Navathe Shamkant., 1994, "Diseño Conceptual de Bases de Datos", Addison-Wesley Iberoamericana Wilmington, Delaware, E.U.A.
- [Billy, 2004] Billy Reynoso Carlos, 2004, "Introducción a la Arquitectura de Software", Versión 1.0, Argentina.
- [Booch, 1999] Booch Grady, James Rumbaugh e Ivar Jacobson, 1999, "The Unified Modeling Language User Guide", Rational Software Corporation. Addison-Wesley, E.U.A.
- [Braude, 2003] Braude Eric J., 2003, "Ingeniería de Software", Boston University, Alfaomega, México.
- [Cota, 1994] Cota A., 1994, "Ingeniería de Software". Soluciones Avanzadas, México.
- [De Miguel, 1999] De Miguel Castaño Adoración y Piattini Velthuis Mario G., 1999 "Fundamentos y Modelos de bases de datos", 2ª.ed., Alfaomega, México.
- [Del campo y Sánchez, 1984] Del Campo y Sánchez Martín, 1984, "Malacología Mexicana Antigua", Univer. Aut. Baja Cal. Sur.
- [Hansen, 1998] Hansen Gary W y James V., 1998, "Diseño y Administración de Base de Datos", Prentice Hall, México.
- [Hawryszkiewycz, 1994] I.T. Hawryszkiewycz, 1994, "Análisis y Diseño de Bases de Datos", Noriega Editores, México.
- [Hernandez, 2003] Hernández Domingo, Escuela de Ingeniería de Sistemas, Departamento de Computación.  
[http://www.magma.com.ni/~jorge/upoli\\_uml/refs/Introduccion\\_UML.pdf](http://www.magma.com.ni/~jorge/upoli_uml/refs/Introduccion_UML.pdf)
- [Hurtado, 2003] Hurtado Sandra Victoria Gil, Representación de la Arquitectura de Software usando uml". [http://www.icesi.edu.co/es/publicaciones/publicaciones/contenidos/sistemas\\_telematica/1/shurtado\\_repres-uml.pdf](http://www.icesi.edu.co/es/publicaciones/publicaciones/contenidos/sistemas_telematica/1/shurtado_repres-uml.pdf)
- [Luque y Nieto, 2002] Luque Ruiz Irene y Nieto-Gómez Miguel Angel, 2002, "Base de Datos", Alfaomega, México.
- [Parrilla, 1997] Parrilla Juan Carlos, Juan José Rubio, 1997, "Sistemas Gestores de Bases de Datos", Síntesis, España.
- [Martín, 1995] Martín James, 1995. "Organización de Bases de Datos", Prentice-Hall, México.
- [Rumbaugh et al, 1998] Rumbaugh James, Ivar Jacobson & Grady Booch, 1998, "The Unified Modeling Language reference Manual", Edit Addison – Wesley.

## Bibliografía

---

- [Scott,1996] Scott Annette, 1996, "Data Modeling and Relational Database Design", Oracle University, E.U.A.
- [Tramullas, 2003] Tramullas Saz, J. 2003, "Análisis preliminar de bibliotecas digitales en las universidades españolas." Actas de las VIII Jornadas Españolas de Documentación, Barcelona: FESABID.
- [Tsai, 1990] Tsai Alice Y.H., 1990, "Sistemas de Bases de Datos, Administración y Uso", Prentice-Hall, México.
- [Ullman, 1999] Ullman Jeffrey D. y Jennifer Widom, 1999, "Introducción a los Sistemas de Base de Datos", Prentice-Hall, México.
- [Willey, 1990] Willey John & Sons, 1990, "Software Engineering", E.U.A.

<http://www.monografias.com/trabajos5/insof/insof.shtml>

[www.rational.com/uml/](http://www.rational.com/uml/)

[http://bibliodgsca.unam.mx/tesis/tes7cllg/sec\\_10.htm](http://bibliodgsca.unam.mx/tesis/tes7cllg/sec_10.htm)

[http://www.vico.org/pages/Sitios\\_www/vico\\_sitios\\_UML.html](http://www.vico.org/pages/Sitios_www/vico_sitios_UML.html)

<http://usuarios.lycos.es/ooopere/uml.htm>

[http://mitotech.epublish.cl/articles/epublish\\_20030630094047.html](http://mitotech.epublish.cl/articles/epublish_20030630094047.html)

[http://www.microsoft.com/spanish/msdn/arquitectura/roadmap\\_arg/arquitectura\\_soft.asp](http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arg/arquitectura_soft.asp)

<http://www.unalmed.edu.co/~mstabare/Dbms.htm>

<http://www.ibiologia.unam.mx/pdf/sistematica>

[http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec\\_13.htm](http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec_13.htm)

<http://www.bibnal.edu.ar/paginas/recursobiblio.htm>

<http://www.microsoft.com/latam/technet/articulos/200201/art08/>