



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
*FACULTAD DE INGENIERÍA*

***APLICACIÓN DE LAS TICS EN LA DIRECCIÓN DE  
OPERACIONES DEL LABORATORIO DE MECÁNICA***

TESIS PARA OBTENER EL TÍTULO DE:  
*INGENIERO EN COMPUTACIÓN*

PRESENTA:

*CÉSAR SAÚL TRUJILLO GATICA  
GASPAR CÁRDENAS GÚZMAN*

DIRECTOR DE TESIS:

*INGENIERO RAÚL ESCALANTE ROSAS*



MÉXICO, D.F., 2013



## Índice

Introducción.....	i
Antecedentes.....	i
Justificación .....	ii
Objetivos.....	iv
Alcances.....	v
1.-Fundamentos Teóricos.....	3
1.1.-Servidores.....	3
1.1.1.-Correo .....	4
1.1.2.-BD.....	5
1.1.3.-Web.....	5
1.1.4.-Seguridad .....	6
1.2.-Servidores WEB .....	6
1.2.1.-Apache .....	6
1.2.2.-IIS .....	7
1.3.-Manejadores de DB .....	7
1.3.1.-Tabla comparativa .....	7
1.4.-Arquitecturas y tipos de servidores .....	8
1.4.1.-Arquitectura Cliente-Servidor.....	8
1.4.2.- Arquitecturas Multitier (Distribuida) .....	9
1.5.-Metodologías para el desarrollo de sistemas.....	10
1.5.1.-Tabla comparativa de metodologías.....	10
1.5.2.-Análisis de metodologías .....	14
1.6.-Lenguajes de desarrollo .....	15
1.6.1.-Tabla comparativa de lenguajes de desarrollo.....	16
1.6.2.-Análisis de los lenguajes de programación .....	17
1.7.-Seguridad.....	19
2.-Análisis y diseño del sistema .....	25

2.1.-El laboratorio de mecánica .....	25
2.1.1.-Análisis del sistema de inscripciones .....	25
2.1.2.-Análisis de sistemas similares .....	28
2.2.-Selección de metodología de desarrollo .....	29
2.3.-Selección de tecnologías .....	30
2.4.-Identificación de procesos .....	31
2.5.-Diseño del sistema .....	33
2.5.1.-Diseño de los módulos del sistema .....	33
2.5.2.-Diseño del panel de Administrador .....	40
2.6.-Diseño de la base de datos .....	42
2.7.-Diseño de la interfaz de usuario.....	50
3.- Pruebas, puesta en marcha y mantenimiento .....	55
3.1.-Pruebas.....	55
3.2.-Puesta en marcha .....	62
3.3.-Requerimientos .....	62
3.4.-Mantenimiento.....	64
Conclusiones y recomendaciones.....	67
Conclusiones.....	67
Recomendaciones.....	68
Bibliografía y referencias.....	71
Apéndice A.....	75
Conceptos básicos sobre bases de datos .....	75
Fundamentos de la normalización .....	79

# INTRODUCCIÓN



# Introducción

## Antecedentes

Breve Historia del Laboratorio de Mecánica de la División de Ciencias Básicas de la Facultad de Ingeniería

A partir del segundo semestre lectivo de 1979 y dando respuesta a la propuesta del MI. Luis Ordóñez Reyna se llevaron a cabo los primeros pasos encaminados a implementar el Laboratorio de Mecánica Básica.

Se comenzó por establecer los Lineamientos Generales para la instalación, funcionamiento y mantenimiento de un Laboratorio de Mecánica Básica para la coordinación de Materias Propedéuticas de la Facultad de Ingeniería.

Posteriormente se llevó a cabo un trabajo consistente en recabar información relativa a diferentes laboratorios de mecánica instalados en algunos centros de educación superior ubicados en el D.F., en los cuales se utilizan diversos modelos ilustrativos y material filmado para complementar los cursos teóricos de diversas asignaturas del área de mecánica.

A continuación, se realizó el análisis detallado de esa información con el objeto de establecer una proposición básica para el Laboratorio de Mecánica de la Coordinación de Materias Propedéuticas, tomando en cuenta los recursos disponibles.

Como resultado de ello, en los semestres II-79 y I-80 se llegaron a implementar ocho prácticas enfocadas a diversos temas de dinámica.

Debido al interés creciente de los alumnos de Estática y Mecánica, se implementaron posteriormente otras cuatro prácticas enfocadas principalmente a algunos temas básicos de estas asignaturas.

Se elaboró un Manual de Prácticas de Laboratorio en el cual se hace la descripción en forma resumida de las mencionadas 12 prácticas de laboratorio (II-80).

Complementariamente, en el primer semestre lectivo de 1982, como parte del subprograma del Laboratorio de Mecánica, se elaboraron dos audiovisuales, uno enfocado a hablar sobre la Historia de la Mecánica y el otro consistente en una película corta sobre una práctica relacionada con el medidor de inercia.

Posteriormente se elaboraron dos prácticas adicionales relacionadas con temas básicos de Estática.

Actualmente se cuenta con doce prácticas de laboratorio, seis de Estática y seis de Cinemática y Dinámica, que abarcan diversos temas básicos de las dos asignaturas del Departamento de Ciencias Aplicadas.

### Justificación

Debido al crecimiento poblacional de la Facultad de Ingeniería, su área administrativa se vio en la necesidad de desarrollar mecanismos que le faciliten y apoyen sus variadas y cotidianas tareas, ejemplos claros son: las inscripciones, emisión de constancias para alguna acreditación, tiras de materias, etc.

En un principio, estos procesos se desarrollaban manualmente, las personas involucradas realizaban exceso de tareas, de forma constante y repetitiva, originando retrasos. Para mantener el funcionamiento veraz y oportuno de la institución se requería una mayor cantidad de personal capacitado, para ofrecer un buen tiempo de respuesta para cualquier trámite solicitado.

Es entonces que con el desarrollo de la computación y su uso como herramienta administrativa, se comenzó a adquirir confiabilidad en esta para realizar tareas repetitivas y extensas, por tanto obtuvo una mayor aceptación entre las instituciones.

Fue así como en la Facultad de Ingeniería entre otras facultades, decidió implementar un sistema computacional con el cual se tuviera un mejor control de su población estudiantil la que se incrementaba con cada ciclo escolar.

Con estos sistemas inicia la forma de revolucionar el trámite de las inscripciones, la velocidad y el menor tiempo con la que se atendía a los



alumnos mejoraba cada vez más, sin embargo a pesar del rápido tiempo de respuesta del sistema éste aún dependía de la operación de una persona que ingresaba los datos de los alumnos para su análisis, lo cual mantenía sujeto al alumno al monótono proceso de hacer una fila para ser atendido.

De esta manera las primeras aplicaciones que funcionaban en red comenzaron a surgir entre las instituciones y las corporaciones, mismas que permitían la comunicación entre las “terminales de trabajo”, (nombre que adquirieron las computadoras que se conectaban en red para realizar una tarea) conectadas a un servidor. Esto solo significaba que más personas podían realizar alguna tarea, sin embargo no eximia al alumno de hacer una fila para ser atendido, ya que aún se necesitaba a una persona que realizara esta actividad intermedia, entre el sistema y el alumno. Sin embargo, lo importante era que las aplicaciones en red estaban recibiendo más atención por parte de los desarrolladores de sistemas que veían en la red local el laboratorio perfecto para desarrollar las aplicaciones “cliente-servidor”.

Posteriormente, con el desarrollo, capacidad de difusión y accesibilidad para el usuario que adquirió el Internet, los sistemas computacionales fueron desarrollándose aprovechando el potencial que ésta les ofrecía al ser una gran red a nivel mundial.

Fue entonces, debido a que los desarrolladores de software comenzaron a explotar el campo de los sistemas en red, surgieron las aplicaciones WEB que en principio añadían al usuario el poder de interactuar con los sitios que en sus inicios solo permitían mostrar información pero sin ninguna interacción con el usuario. Este nuevo poder de interacción creó una nueva generación de páginas llamadas “ páginas dinámicas”, que tomaban datos del usuario y mostraban diferentes contenidos dependiendo de sus solicitudes a través de pequeños programas que residían en el servidor y se conectaban a la página llamados CGI's Interfaz de entrada común (en inglés *Common Gateway Interface*, abreviado CGI) que recibían las entradas de la página, procesaban y modificaban el contenido del HTML siglas de *HyperText Markup Language* (Lenguaje de Marcado de Hipertexto), lenguaje con el cual se describe la estructura y el contenido en forma de texto de la página WEB para realizar una interacción con el usuario.

Debido al éxito obtenido de la WEB para comunicarse e interactuar con las personas de manera masiva fue que se desarrollaron gran cantidad de tecnologías y lenguajes de programación, permitiendo cada vez desarrollar sitios más complejos desde el punto de vista del desarrollador, lo cual daba como resultado en una mejor usabilidad e interfaz gráfica para el usuario final.

En la actualidad la mayoría de las páginas están construidas en base a diversas tecnologías dependiendo de las necesidades del diseño, gracias a la evolución que estas han tenido, permiten desarrollar sistemas más complejos en cuanto a funcionalidad casi semejante a la de una aplicación de escritorio.

De ésta forma las instituciones como la Facultad de Ingeniería, desarrollan tantas aplicaciones como se exijan para la administración de la misma, de manera rápida y funcional; Lo cual demuestra que una vía eficiente de realizar el proceso de inscripción a la facultad es mediante la WEB, de esta forma es como el Laboratorio de Mecánica de la Facultad de Ingeniería ha decidido desarrollar su propio sistema de inscripciones, ya que al no ser éste parte curricular de la formación de los alumnos, sólo se inscriben aquellos que quieren profundizar en al conocimiento de la mecánica clásica mediante experimentos prácticos. A su vez el número de alumnos inscritos aumenta cada semestre, de ahí la importancia y necesidad de automatizar su manejo.

## Objetivos

### **Objetivo General:**

- Crear un sistema multiplataforma para la automatización de las inscripciones al laboratorio de mecánica

### **Objetivos Particulares:**

- Automatizar las inscripciones
- Registro de alumnos
- Asignación de grupo a los alumnos registrados
- Altas y cambios de grupo
- Monitor de vacantes
- Tener disponible la información en todo momento.
- Monitoreo de las inscripciones
- Generar reportes

- Generar estadísticas
- Impresión de listas de asistencia y de calificaciones

### Alcances

Desarrollar un sistema que agilice y automatice parte de las actividades administrativas que se realizan en el laboratorio de Mecánica. Permitiendo la posibilidad de ampliar las funciones de dicho sistema e incorporando nuevos módulos que vayan haciendo más robusta la aplicación.



# **CAPÍTULO 1**

## **FUNDAMENTOS TEÓRICOS**



## 1.-Fundamentos Teóricos

### Navegadores

Los navegadores WEB son programas que funcionan sobre internet, que nos permiten interpretar la información que se encuentra en la red, mostrándola de una manera entendible (decodificada) y agradable al usuario. Con el paso del tiempo, los navegadores se han convertido en programas más complejos debido a la gran cantidad de información que viaja por la red, además de interpretarla, ya que cuenta con la información que los usuarios necesitan para trabajar, comprende la información que da los vistosos formatos con los cuales se despliegan las páginas WEB en la pantalla, (colores, tipografías y posiciones de los elementos de la página, etc.). Actualmente existen una gran cantidad de navegadores WEB de diferentes empresas, pero la mayoría están sujetos a las mismas normas de funcionamiento, para ser interfaces eficientes entre el usuario y los diversos servidores que prestan sus servicios en la red.

#### 1.1.-Servidores

##### ¿Qué es un servidor?

Un servidor es un programa especializado en proporcionar uno o varios servicios a las computadoras que estén conectadas directamente o a través de una red de manera unidireccional y/o bidireccionalmente.

El servidor atiende peticiones emitidas por los clientes (otras computadoras conectadas a la red) a través de protocolos<sup>1</sup> procesándolas para generar una respuesta. El cliente a través de un navegador WEB toma esta respuesta, la interpreta para realizar cambios en la página. Existen varios tipos de servidores como:

- Servidor de correo electrónico.
- Servidor de Bases de datos.
- Servidor WEB.
- Servidor de seguridad.

---

<sup>1</sup> Andrew S, Tanenbaum 2003, Redes de computadoras.

Dependiendo del servidor que se trate y la cantidad aproximada de clientes que solicitarán sus servicios, es cómo se calcula el hardware en el que será alojado y si será un servidor dedicado a uno compartido. Los servidores dedicados se utilizan para servicios de alta demanda, ya que son servidores que se encuentran alojados en computadoras que solo se dedican a las aplicaciones de un cliente (en este caso cliente significa la persona o empresa que contrata el servicio de otra compañía dedicada al alojamiento) ya sean WEB, BD, etc. Algunas de las ventajas de este tipo de servidores son: en la contratación de este servicio, tiene acceso a todos los recursos de la computadora, por tanto puede configurarla como más le convenga, el mantenimiento y conexión a la red corren por parte de quien presta el servicio, lo cual incrementa el costo del servicio.

Otro tipo de servidores que existen son los compartidos, como su nombre lo indica, son servidores que se comparten entre varias empresas que soliciten su servicio de alojamiento. Además de ser más económico, permite tener alojadas a “n” cantidad de sitios dependiendo de las características del servidor como memoria RAM, disco duro, etc.

Cada empresa que contrate este servicio tiene acceso independiente a su sitio, así como a su panel de configuración, pero no puede variar los parámetros de la configuración del servidor más allá de lo que le permita dicho panel de control.

### *1.1.1.-Correo*

En base a la definición anterior del significado un servidor, los servidores de correos son programas que se dedican a enviar y recibir mensajes simulando un comportamiento similar al del correo postal, con la diferencia de que el correo electrónico se trata de datos que viajan por la red, por tanto no pueden ser enviados paquetes o cosas físicas como en el correo postal, sin embargo, pueden ser enviadas imágenes, videos, música y archivos digitales, a través de los datos que el servidor de correo interpreta para mostrárselos al destinatario de la misma manera en la que le fueron enviados por el remitente del correo. Actualmente, existe una gran variedad de servidores de correo electrónico con características variadas respecto a la administración de los correos, sin embargo todos conservan el mismo principio que es el de enviar y recibir datos a una dirección de correo electrónica y única.



### 1.1.2.-BD

Un servidor de bases de datos es a su vez, un programa que presta el servicio de bases de datos a los clientes (nodos conectados a la red) que lo soliciten. Es decir, que en él se encuentran almacenadas las bases de datos de uno o varios clientes, los cuales realizan peticiones de información que les es útil a través de un programa manejador de bases de datos al servidor y este les responde enviándoles la respuesta a su petición. Los servidores de BD también atienden las peticiones que le hacen otros programas que se encuentran en la red para llevar a cabo otras tareas específicas como: almacenar un correo electrónico de algún usuario, autenticar algún usuario para permitirle el acceso a algún sitio y demás datos, con el objetivo de establecer alguna relación con la empresa en cuestión. El lenguaje con el que se llevan a cabo estas consultas es SQL de sus siglas en inglés *Structured Query Language*, el cual estructura la consulta para que el servidor regrese exactamente los datos que el cliente necesita. Este lenguaje fue creado por IBM y en la norma ANSI SQL'92 se estipulan las funciones básicas que los diferentes Sistemas Manejadores de Bases de Datos deben tener<sup>2</sup>.

### 1.1.3.-Web

De igual manera los servidores WEB, son programas que corren en computadoras que se dedican a dar servicios a clientes que se conectan a ellas por medio de una red. En este caso los servidores WEB por lo general se utilizan para almacenar contenido al cual tengan acceso los nodos conectados a la red, mediante páginas WEB con el protocolo HTTP, por sus siglas en inglés *Hypertext Transfer Protocol*<sup>3</sup> de ahí que su uso principal sea el dar alojamiento a páginas WEB. Aunque la principal función de estos servidores es mostrar páginas WEB, la implementación completa del protocolo HTTP, permite otras funciones como la de enviar datos de formularios al servidor e incluso enviar archivos.

---

<sup>2</sup> <http://www.um.es/docencia/barzana/DIVULGACION/INFORMATICA>

<sup>3</sup> Andrew S, Tanenbaum 2003, Redes de computadoras.

### 1.1.4.-Seguridad

La seguridad en cuanto a cómputo, se puede aplicar tanto al *software* como al *hardware*. En lo que respecta al hardware tienen que ver diversos factores como: la ubicación del equipo, la temperatura, los ruidos eléctricos, altibajos de tensión, cortes de corriente, protección de suciedad, protección contra incendios e inundaciones y la accesibilidad, para controlar quien o quienes pueden tener acceso a él. Lo anterior está relacionado específicamente a la seguridad física que se encarga de proteger la integridad del equipo.

Los servidores de seguridad en cuestión de software pueden ser empleados para prestar el servicio de seguridad a otros servidores, por ejemplo el *firewall* es un servidor que protege los puertos del equipo para prevenir cierto tipo de ataques aplicando un conjunto de reglas especificadas por el administrador del servidor<sup>4</sup>. Es por eso que el uso más frecuente de los servidores de seguridad sea impedir que los usuarios de internet tengan acceso a redes privadas<sup>5</sup>.

## 1.2.-Servidores WEB

### 1.2.1.-Apache

Apache Tomcat es un servidor EWB con soporte para servlets y JSPs construido como una implementación de código abierto del software de las tecnologías JavaServlet y JavaServer Page.

El Servlet Java y especificaciones JavaServer Pages son desarrollados bajo Java Community Process. Este se desarrolla bajo un ambiente abierto y participativo de una comunidad de los mejores desarrolladores alrededor del mundo.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

---

<sup>4</sup> <http://www.java.com>

<sup>5</sup> <http://support.microsoft.com>

**Módulos Base:** Módulo con las funciones básicas del Apache.

**Módulos Multiproceso:** son los responsables de la unión con los puertos de la máquina, acepando las peticiones y enviando a los hijos a atender a las peticiones.

**Módulos Adicionales:** Cualquier otro módulo que le añada una funcionalidad al servidor.

### 1.2.2.-IIS

*Internet Information Services* (IIS) es un servidor web flexible, seguro y administrable para hospedar información en la Web.

### 1.3.-Manejadores de DB

El Sistema Manejador de Base de Datos (DBMS) permite crear la colección de datos interrelacionados.

#### 1.3.1.-Tabla comparativa

SGBD	Características	Hardware	Plataforma
<i>MySQL</i>	Bases de datos relacionales. Soporta procesos multihilo y multiusuario ANSISQL 99. Soporta procedimientos precargados. De uso libre.	Se considera como en consumidor medio de los recursos del sistema.	Multiplataforma
<i>Postgres SQL</i>	Bases de datos relacionales. Alta concurrencia. Provee nativamente números de precisión arbitraria. Texto de largo ilimitado De uso libre.	Consume una gran cantidad de los recursos del sistema.	Multiplataforma
<i>Microsoft SQL server</i>	Bases de datos relacionales. Estabilidad. Seguridad alta.	Consume una gran cantidad de los recursos del sistema.	Microsoft Windows

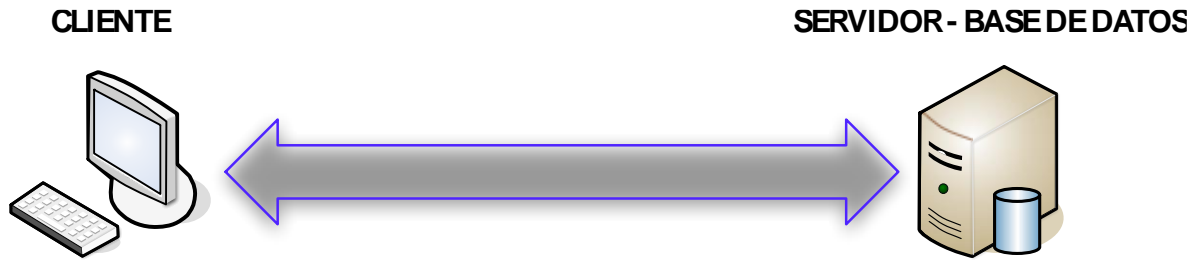
	Soporta procedimientos precargados. Arquitectura cliente-servidor.		
<i>Microsoft Access</i>	Estabilidad. Bases de datos relacionales. Interfaz gráfica. Utilerías para uso con bases de datos. Incluido en office.	Se considera como en consumidor medio de los recursos del sistema.	Microsoft Windows
<i>Oracle Database</i>	Mayor control de las redundancias de la información. Su precio es muy elevado.	Alto consumos de los recursos del sistema.	Multiplataforma
<i>DB2</i>	Bases de datos relacionales. Integra XML de forma nativa. Ejecuta PL/SQL de manera nativa. Existe una versión gratuita.	Se considera como en consumidor medio de los recursos del sistema.	Multiplataforma

#### 1.4.-Arquitecturas y tipos de servidores

##### 1.4.1.-Arquitectura Cliente-Servidor

Dentro de la informática, la arquitectura cliente servidor, tiene como principio fundamental, agrupar a todo tipo de aplicaciones que sigan este tipo de arquitectura, en dos grupos básicos, en uno de los conjuntos se encuentran las aplicaciones que se dedican a prestar un servicio a los usuarios que estén conectados a ellas, en el segundo conjunto, a todas las aplicaciones que hagan uso de esos servicios<sup>6</sup>.

<sup>6</sup> <http://www.mastermagazine.info/termino/4294.php>



Cliente. Demanda servicios.

Servidor. Provee servicios.

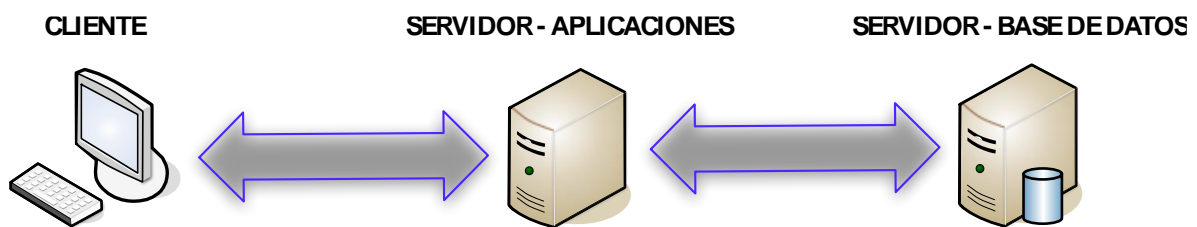
### **Ventajas**

- Reducción de costos de la plataforma.
- Favorece el uso de interfaces gráficas amigables.
- Reducción del tráfico en el ancho de banda debido a que existe información alojada en el cliente (ej. interfaz gráfica).
- Proporciona soluciones locales con información integral a nivel global.

### **Desventajas**

- Mantenimiento más complicado por la distribución del hardware y software,
- Complejidad por las diferentes plataformas.
- La seguridad por las diferentes plataformas.

### *1.4.2.- Arquitectura Multitier (Distribuida)*



El servidor de aplicaciones proporciona servicios de aplicación a los clientes y gestiona el acceso a los datos.

En esta arquitectura el servidor de aplicaciones procesa los datos del cliente y realiza la petición al servidor de base de datos, es decir que actúa como una cola.

**Ventajas**

- Mejora el balance de carga entre servidores.
- Escalabilidad de clientes y servidores.
- Fácil mantenimiento.

**Desventajas**

- Dificulta la programación.
- El cliente no dispone directamente de los recursos del servidor.

**1.5.-Metodologías para el desarrollo de sistemas**

Existen varias metodologías para el desarrollo de software que ayudan a tener una perspectiva clara de los pasos a seguir para no desperdiciar tiempo en avanzar en direcciones erróneas. A continuación se muestra una tabla comparativa que analiza las ventajas y desventajas de las diferentes metodologías de desarrollo de software.

**1.5.1.-Tabla comparativa de metodologías**

<b>Modelo</b>	<b>Enfoque</b>	<b>Ventajas/Desventajas</b>	<b>Aplicabilidad</b>
<i>Modelo en cascada</i>	Se debe esperar a que termine la etapa anterior, para el inicio de una nueva etapa.  Cualquier error de diseño detectado en la etapa de prueba, conduce al aumentando de los costos del desarrollo.	Los proyectos raras veces siguen una evolución secuencial. No todos los requisitos son expuestos, al principio, de forma explícita como requiere este modelo. El cliente debe tener paciencia, ya que la aplicación sólo estará disponible en un estado muy avanzado del proyecto.  Ampliamente criticado desde el ámbito académico y la industria.	Utilizado cuando existen especificaciones amplias de los requerimientos del cliente.

<p><i>Modelo basado en prototipos</i></p>	<p>Prototipos: No posee la funcionalidad total del sistema pero si condensa la idea principal del mismo, Paso a Paso crece su funcionalidad, alto grado de participación del usuario.</p>	<p>El cliente puede pensar el prototipo es una versión acabada.</p> <p>Pueden llegar a pasarse por alto la calidad del software global o el mantenimiento a largo plazo.</p> <p>Las herramientas elegidas pueden ser inadecuadas. La clave del éxito de este modelo consiste en definir bien, desde el principio, las reglas del juego.</p> <p>Participación del usuario.</p>	<p>Se utiliza si en el mercado no se encuentra el producto pero el cliente desea resultados inmediatos.</p> <p>Convenientemente en caso de ser necesario desarrollar módulos.</p> <p>Para sistemas interactivos pequeños o de tamaño pequeño.</p> <p>Para partes de sistemas grandes.</p> <p>Para sistemas con vida corta.</p>
<p><i>Modelo incremental o evolutivo</i></p>	<p>Modelo Lineal-Secuencial con el Modelo Basado en Prototipos.</p> <p>El sistema no se entrega de una vez, sino que se divide y se entregan incrementos. Con cada incremento se entrega la parte de la funcionalidad que se ha establecido.</p> <p>Los requisitos de un incremento son inamovibles. Sin</p>	<p>Los clientes no tienen que esperar hasta tener el sistema completo. El primer incremento satisface los requisitos más críticos.</p> <p>Los primeros incrementos sirven como prototipos y ayudan en la tarea de detectar los posteriores requisitos.</p> <p>Existe un riesgo bajo de fallar en el proyecto total.</p>	<p>Reemplazar el antiguo desarrollo con uno nuevo que satisfaga las nuevas necesidades según las redefiniciones del problema.</p> <p>Manejo de versiones.</p>

	<p>embargo estos pueden verse modificados en incrementos posteriores.</p> <p>Este proceso se repite hasta la obtención de un producto completo.</p> <p>Sin embargo el modelo incremental se centra en la entrega de un producto operativo en cada incremento.</p>	<p>Los servicios del sistema con la prioridad más alta tienden ser los más probados.</p>	
<i>Modelo espiral</i>	<p>Es una mejora del Modelo Basado en prototipos. Cada vuelta en la espiral representa una fase del proceso.</p> <p>No hay fases fijas, cada vuelta en la espiral determina las actividades a realizar.</p> <p>La dimensión radial representa el costo acumulado en la financiación de las fases.</p> <p>La dimensión angular representa el progreso hecho</p>	<p>Requiere comunicación permanente con el cliente y éste debe ser gran conocedor del sistema.</p>	<p>Utilizado para el desarrollo de aplicaciones complejas y/o específicas. (Ej. Investigación Genética).</p>



	<p>en completar cada ciclo de la espiral.</p> <p>Un ciclo a través de la espiral es simular un paso a través de un modelo en cascada.</p>		
<i>Modelo basado en componentes (orientado a objetos).</i>	<p>Es programación orientada a Objetos. Se utilizan objetos, clases y se reutilizan en diferentes partes del sistema.</p>	<p>Optimiza los tiempos de respuesta a los requerimientos del cliente y facilita la labor del programador pues hay un alto aprovechamiento del código.</p> <p>Facilita mantenimiento del software.</p>	<p>Sistemas robustos y de alta proyección.</p>
<i>Code and Fix</i>	<p>No requiere planeación y se trata de codificar y corregir. Se trabaja mediante prueba y error. Especial para desarrollos rápidos y sencillos.</p>	<p>Desarrollo Rápido.</p> <p>No garantiza calidad.</p>	<p>Desarrollo muy pequeño con claridad de objetos, requerimientos pequeños o de mantenimiento con bajo impacto.</p>
<i>Cascada con subproyectos</i>	<p>Requiere planeación</p>	<p>Plantea Organización y planeación de un gran proyecto. Se pueden realizar varias partes del proyecto al mismo tiempo por diferentes desarrolladores.</p>	<p>Adecuada para el desarrollo de proyectos complejos que estiman de 1 a 3 años de desarrollo.</p>
<i>Entrega por etapas</i>	<p>Cascada con entregas grandes en diferentes etapas del desarrollo.</p>	<p>Debe entregarse una etapa para continuar con la siguiente.</p>	<p>Desarrollos robustos. Desarrollo depende del presupuesto directamente (Ej. Ppto. Adjudicado anual)</p>

	Cascada con Evolutivo.		
--	------------------------	--	--

### *1.5.2.-Análisis de metodologías*

Comparando las diferentes metodologías de desarrollo de software se observa que el modelo en cascada no se adapta del todo al desarrollo del sistema de inscripciones, ya que según el planteamiento de dicha metodología, se debe terminar un proceso antes de iniciar otro, lo cual no es necesario en el desarrollo de éste sistema, ya que la arquitectura modular del nuevo sistema de inscripciones permite que se desarrollen los módulos por separado.

El modelo basado en prototipos aparentemente se adapta a nuestro plan de trabajo ya que propone que el sistema va creciendo su funcionalidad poco a poco, pero por otro lado, plantea la participación constante y cercana del cliente, lo cual no es del todo necesario en éste caso, porque basándonos en los requerimientos del sistema, se puede trabajar independientemente del cliente para lograr avances claros y significativos. Además dice que en todo momento el cliente puede considerar cada prototipo como un producto terminado y en éste caso con la terminación de los módulos individualmente no se puede considerar como un producto terminado.

El modelo incremental o evolutivo se basa en el modelo de prototipos, pero tiene una variante que lo hace el más adecuado para el desarrollo del sistema de inscripciones; esta variante consiste en que el sistema se divide en partes y se va desarrollando de manera creciente, con cada entrega se va completando la funcionalidad total del sistema; esto significa una gran ventaja ya que el sistema será modular y por tanto se puede adaptar muy bien a éste modelo de desarrollo. Aunque este modelo aún se basa en la entrega de productos terminados en cada incremento del desarrollo del sistema, se puede considerar a cada módulo como un producto terminado el cual formara parte de todo el sistema.

La metodología en espiral, aunque es una mejora del modelo de prototipos, ésta también requiere la participación constante del cliente durante todo el proceso de desarrollo lo cual no es necesario en éste caso, ya que solo se requiere de una hoja de requerimientos la cual se revisa al terminar un periodo

de desarrollo y se verifica que se hayan cumplido los objetivos planteados para así proseguir con la siguiente parte del programa.

El modelo basado en componentes (orientado a objetos), ayudaría parcialmente a solucionar el problema ya que se basa en clases y se aplica a desarrollos modulares como en nuestro caso, pero debido a que los módulos que estamos manejando en el diseño del nuevo sistema son relativamente diferentes no necesitamos reciclar tanto código, aunque en algunas partes si tenemos contemplado trabajar con programación orientada a objetos.

La metodología *Code and Fix*, es implementada en desarrollos rápidos en los que no se requiere una planeación muy fuerte ya que se trata de programar y corregir, lo cual no funcionaría en este caso debido a que es un sistema muy extenso y se perdería mucho tiempo si se aplicara esta metodología.

Por último las metodologías *Cascada con subproyectos* y *Entrega por etapas* son derivaciones de la metodología en cascada, la cual no es aplicable de manera óptima a éste proyecto, debido a sus limitantes en el desarrollo de un sistema que está compuesto por módulos independientes y que se pueden desarrollar en paralelo.

## 1.6.-Lenguajes de desarrollo

### ***¿Qué es un lenguaje de programación?***

Los lenguajes de programación se pueden considerar como un idioma artificial diseñado para la comunicación entre el humano y la computadora para construir secuencias de código que pueda ejecutar una computadora<sup>7</sup>. Estos lenguajes datan de las primeras computadoras digitales siendo esa época un reflejo del hardware que controlaban, es decir, eran específicos para cada hardware. Hoy en día han evolucionado para convertirse en lenguajes independientes de la plataforma computacional en que son implementados<sup>8</sup>.

Los lenguajes de programación que actualmente se usan para este tipo de aplicaciones WEB son muy variados, debido a los nuevos sistemas operativos, tendencias y necesidades de los usuarios, han surgido nuevas tecnologías,

---

<sup>7</sup> Como Programar en Java, Paul Deitel, Novena ED, 2012, Pearson.

<sup>8</sup> <http://www.tic.unam.mx>

algunas especiales para cada sistema y otras que son multiplataforma, las cuales se han popularizado debido a su versatilidad.

### 1.6.1.-Tabla comparativa de lenguajes de desarrollo

Cada una de estas tecnologías tiene un propósito dentro del desarrollo WEB, así como ventajas y desventajas como se muestra en la siguiente tabla comparativa, en la cual están citados la mayoría de los lenguajes empleados para este tipo de desarrollos.

Lenguaje	Ventajas	Desventajas
<i>PHP</i>	<ul style="list-style-type: none"> <li>*Fácil de aprender</li> <li>*Opcional a:               <ul style="list-style-type: none"> <li>-Orientado a objetos</li> <li>-No orientado a objetos</li> </ul> </li> <li>*Multiplataforma</li> <li>*Conexión con la mayoría de los manejadores de BD</li> <li>*Es de uso libre</li> <li>*Documentación disponible</li> <li>*Existe una gran comunidad de desarrollo</li> </ul>	<ul style="list-style-type: none"> <li>*El procesamiento lo realiza el servidor en el que corre la aplicación.</li> <li>*El código tiende a desordenarse al combinarlo con html.</li> </ul>
<i>JavaScript</i>	<ul style="list-style-type: none"> <li>*Seguro y confiable</li> <li>*Se ejecuta en el cliente, quitándole trabajo al servidor.</li> </ul>	<ul style="list-style-type: none"> <li>*Código visible para el usuario</li> <li>*El código debe descargarse completamente.</li> </ul>
<i>ASP.NET</i>	<ul style="list-style-type: none"> <li>*Orientado a objetos</li> <li>*Fácil de dar mantenimiento</li> <li>*Veloz</li> <li>*Seguro</li> </ul>	<ul style="list-style-type: none"> <li>*Mayor consumo de recursos</li> </ul>
<i>JSP</i>	<ul style="list-style-type: none"> <li>*Multiplataforma</li> <li>*Rápida ejecución</li> </ul>	<ul style="list-style-type: none"> <li>*Complejidad de aprendizaje</li> </ul>
<i>PYTHON</i>	<ul style="list-style-type: none"> <li>*Orientado a objetos</li> <li>*Multiplataforma</li> <li>*Código sencillo y rápido de programar</li> <li>*Libre</li> <li>*Portable</li> </ul>	<ul style="list-style-type: none"> <li>*Lento al momento de la ejecución, ya que el código es interpretado.</li> <li>*El procesamiento lo realiza el servidor en el que corre la aplicación.</li> </ul>
<i>RUBY</i>	<ul style="list-style-type: none"> <li>*Fácil de aprender</li> <li>Gran variedad de librerías disponibles</li> <li>*Desarrollo rápido</li> </ul>	<ul style="list-style-type: none"> <li>*Rendimiento comparado con PYTHON</li> <li>*No existe <i>framework</i> para desarrollo</li> </ul>

		*El procesamiento lo realiza el servidor en el que corre la aplicación.
--	--	---

### 1.6.2.-Análisis de los lenguajes de programación

En cuestión de programación es más simple descartar las posibilidades, ya que la mayoría de los lenguajes tiene muy marcado su rango de acción, por ejemplo, JavaScript es seguro y confiable pero debido a que el código es visible para el usuario no lo usaremos para la mayoría del programa, sólo para las partes en las que se tienen que validar los formularios para asegurarse que se envían los datos en el formato correcto; La prioridad del programa es ser veloz y estable al momento de atender las peticiones de los alumnos, por lo que, PYTHON y RUBY no son las mejores opciones para éste problema, en este mismo sentido entraría ASP el cual requiere un mayor consumo de recursos por parte del servidor; Resultado de éste análisis, el lenguaje óptimo para el desarrollo de éste sistema es una mezcla de PHP y JavaScript.

PHP además de ser multiplataforma al momento de su instalación en el servidor, esto es, que puede correr en una gran diversidad de sistemas operativos, es de uso libre, por ésta razón existe una gran diversidad de documentación gratuita y disponible para los desarrolladores. Una ventaja más de utilizar PHP es que el procesamiento se realiza en el servidor y este es fácil de configurar, no se requiere una gran capacidad de cómputo la cual debe ser proporcional a la cantidad de peticiones esperadas, lo que se ajusta a las necesidades de nuestro desarrollo.

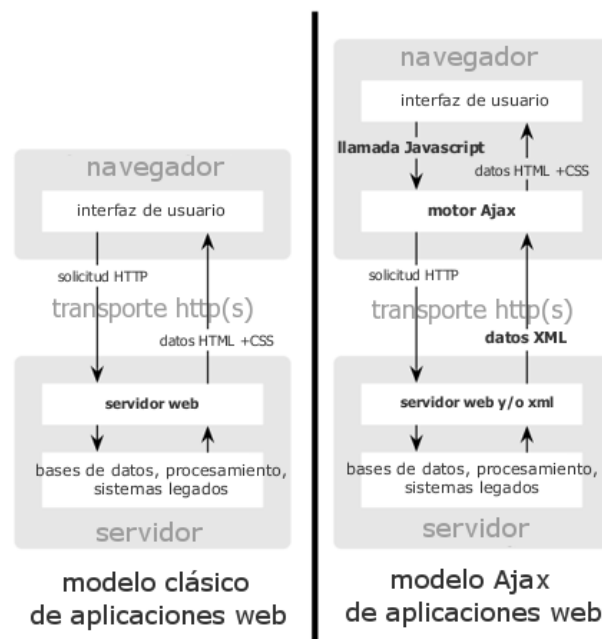
PHP por sus siglas en ingles *Hypertext Pre-processor*, además de ser código abierto compatible con todos los principales servidores WEB (principalmente Apache), tiene la ventaja de poder implementar sobre él la técnica de desarrollo AJAX especialista en crear aplicaciones interactivas de la cual se hablará más al respecto a continuación.

#### **AJAX**

AJAX no es un lenguaje de programación, sino que implementa varias tecnologías que trabajan juntas para lograr cosas más complejas como lo hace Google Maps, Gmail. AJAX, es el acrónimo para *Asynchronous JavaScript +*

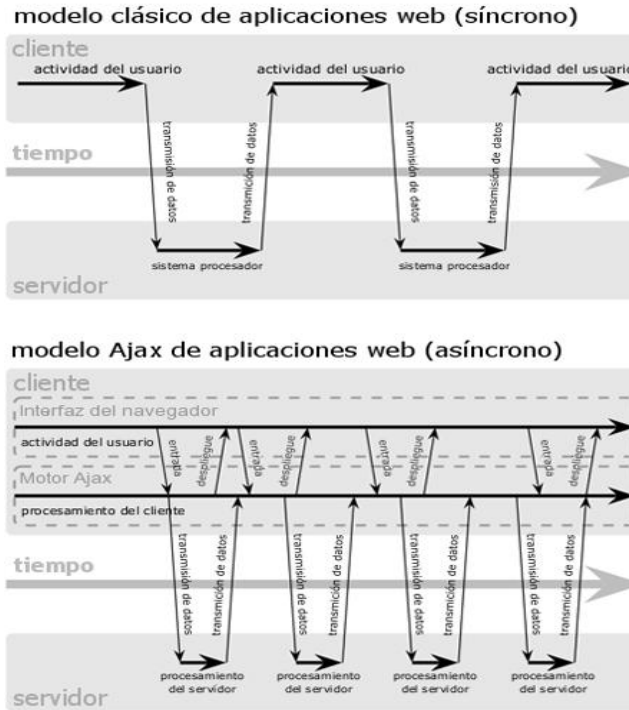
*XML*, consiste en cargar y renderizar una página, luego mantenerse en esa página mientras scripts de Java (por lo general) van al servidor buscando información solicitada por el usuario en segundo plano para actualizar la página y solo re-renderizando algunas partes de ella por medio de las etiquetas “DIV”, mostrando u ocultando partes de la misma<sup>9</sup>.

El modelo clásico de aplicaciones Web funciona de la siguiente forma: Las acciones del usuario en la interfaz gráfica generan una petición HTTP al servidor web. El servidor realiza el proceso solicitado y le devuelve el resultado en una página HTML al cliente. Como se puede ver en el siguiente diagrama:



AJAX permite que la interacción del usuario con la aplicación suceda de manera asíncrona (independientemente de la comunicación con el servidor) es decir, no se tiene que recargar toda la página, sino solo pequeñas partes de ella, por lo tanto, el usuario nunca estará mirando una ventana en blanco y el tiempo de actualización de esa parte será mucho más rápido.

<sup>9</sup> Introducción a Ajax con PHP, Lee Babin, Primera Ed, 2007, Anaya Multimedia.



## 1.7.-Seguridad

### ***Características del servidor para brindar seguridad***

#### ***Apache***

Primero, tener instalados los últimos parches de seguridad.

Ocultar la versión y otra información delicada. Hay dos directivas que necesitas agregar o corregir en el archivo httpd.conf:

```
ServerSignature Off
```

El ServerSignature aparece en la parte inferior de las páginas generadas por apache tales como los famosos errores 404.

```
ServerTokens Prod
```

La directiva ServerTokens se utiliza para determinarse lo que pondrá Apache en la cabecera de la respuesta HTTP del servidor.

Apache debe funcionar bajo su propia cuenta y grupo de usuario. Algunas versiones de Apache corren bajo el usuario nobody, esto compromete mucho su seguridad por lo tanto de quedar:

```
User apache  
Group apache
```

Deshabilitar cualquier módulo innecesario. Apache viene por defecto instalado con una serie de módulos. Revisando la documentación de Apache algunos módulos que no sean útiles en el servidor son: mod\_imap, mod\_include, mod\_info, mod\_userdir, mod\_status, mod\_cgi, mod\_autoindex.

No deseamos que se pueda acceder a los directorios que no tengan permisos para ello, supongamos que el directorio raíz para nuestras webs es /web, la configuración óptima deberá ser la siguiente:

```
Order Deny,Allow  
Deny from all  
Options None  
AllowOverride None
```

```
Order Allow,Deny  
Allow from all
```

Desactivar las opciones para explorar directorios con las opciones de directiva dentro de la etiqueta directorio donde se tienen dos posibles valores none o indexes.

```
Options -Indexes
```

Desactivar la ayuda para los archivos .htaccess.

```
AllowOverride None
```

Otra opción interesante sería bloquear la descarga de todos los archivos que comiencen con .ht por ejemplo, se haría de la siguiente manera:

```
AccessFileName .httpdoverride  
Order allow,deny  
Deny from all  
Satisfy All
```



Disminuir el valor máximo de tiempo de espera. Por el defecto el tiempo de espera es de 300 segundos. Disminuirlo por seguridad para prevenir ataques de esta manera:

```
Timeout 45
```

Se evitó el uso de variables globales, a pesar de que resultan muy cómodas de utilizar representan un problema de seguridad. Por el contrario se utilizaron variables de sesión.

### ***Base de datos***

Creación de usuarios de MySQL con acceso restringido a la base de datos del sistema y con los privilegios necesarios.

Almacenamiento de contraseñas cifradas para acceso de usuarios al sistema.



# CAPÍTULO 2

## ANÁLISIS Y DISEÑO DEL SISTEMA



## 2.-Análisis y diseño del sistema

En este apartado de la tesis se realizará un análisis del sistema de inscripciones actual del laboratorio, en base a éste análisis, se procederá a realizar el diseño y construcción del nuevo sistema.

### 2.1.-El laboratorio de mecánica

El laboratorio de mecánica es un aula de la Facultad de Ingeniería donde se imparten las partes prácticas de las asignaturas de Estática y de Cinemática y Dinámica. Está ubicado en el edificio G del Anexo de la Facultad de Ingeniería. Éste laboratorio cuenta con dos salas (A y B) en las cuales se imparten las prácticas correspondientes a las asignaturas de Estática y Cinemática y Dinámica.

#### *2.1.1.-Análisis del sistema de inscripciones*

Cada semestre se realiza la planeación de la distribución de los grupos de laboratorio de las dos asignaturas, basándose en la demanda de semestres anteriores y la cantidad de alumnos inscritos a las respectivas clases de teoría.

Una vez realizada la planeación se obtiene una relación de grupos de laboratorio con sus respectivos horarios y cupo con que se les va a abrir, éste valor puede variar al momento de realizar los cambios de grupo ya que algunos grupos se saturan rápidamente, siendo necesario abrir más vacantes para dar mejor disponibilidad de horario a los alumnos, pero este valor no puede rebasar el límite de 25 alumnos, por cuestiones de seguridad.

Como el proceso de inscripciones del laboratorio es posterior a las asignaturas curriculares, hay que tener tiempo de solicitar al administrador del sistema de inscripciones de las asignaturas curriculares, nos proporcione la información de los alumnos como: número de cuenta, el número de sorteo de los alumnos que solicitaron las asignaturas para las cuales se imparten éstos laboratorios, con estos datos, es el sistema de inscripciones del laboratorio autentica a los alumnos para asignarle grupo en el laboratorio de mecánica.

La inscripción al Laboratorio de Mecánica para los alumnos, se realiza actualmente en seis etapas principales, consistentes en:

- Etapa de consulta de horarios.
- Registro de opciones.
- Asignación de grupo de laboratorio.
- Publicación de resultados.
- Altas/cambios.
- Información de fechas importantes, como inicio de semestre.

La primera etapa del proceso es simplemente informativa, ésta consiste en publicar las listas con los grupos y horarios ofertados por el laboratorio para el semestre que está por iniciar.

En la segunda “registro de alumnos”, se les entrega un formulario en el laboratorio que debe ser llenado a mano con sus datos personales y una serie de cuatro opciones en orden de preferencia del total de grupos que el laboratorio ofrece en el ciclo escolar vigente, así como, su número de reinscripción. Posteriormente ellos entregan de nuevo la papeleta en el laboratorio para la asignación de grupo, la cual se hace basándose en el número de reinscripción a las asignaturas curriculares. De esta forma el alumno queda registrado.

Existe una sub-etapa, la cual es transparente para el alumno, la cual consiste en la asignación del grupo de laboratorio, esto se hace ordenando a mano las papeletas que los alumnos entregaron con su número de reinscripción; Éste proceso es bastante tardado ya que la matrícula promedio del laboratorio de mecánica es de aproximadamente 2900 alumnos, esto significa que varias personas tienen que dedicar un mínimo dos días para completar el ordenamiento, una vez que esto se realiza se procede a la asignación de grupo de laboratorio. La persona que está realizando la tarea de asignar los grupos, tiene a la mano una lista de todos los grupos de laboratorio ofertados, en la cual, va anotando la cantidad de alumnos que cada grupo ya tiene inscritos cada que procesa una de las papeletas, entonces, se toma la primera papeleta, la del número de reinscripción más bajo, se verifica que la primera opción de laboratorio del alumno no esté llena, o sea que tenga vacantes, de ser así, se captura la primera opción de laboratorio del alumno, en caso de que la

primera opción esté llena se procede a verificar la segunda opción, así sucesivamente, cuando se le asigne un grupo se actualiza la cantidad de alumnos inscritos en ese grupo, en el caso de que las cuatro opciones que el alumno escogió estén llenas se le deja sin grupo de laboratorio para que después en el periodo de altas y cambios el alumno escoja uno de los grupos que le convenga en horario que tenga vacantes, pero siendo esta la primera papeleta es seguro que el alumno tendrá su primera opción de laboratorio. Este proceso se repite para los “n” registros por lo que es una tarea aún más tardada que la de ordenar las papeletas, ya que además de capturar las opciones de laboratorio de los alumnos se realiza ésta pequeña validación por parte de la persona encargada de la inscripción, entonces, este proceso es mucho más tardado que el anterior.

El resultado de ésta etapa es generar una lista con todos los alumnos que solicitaron grupo de laboratorio, junto con el grupo de laboratorio que les fue asignado, la que posteriormente es publicada para que los alumnos se tengan conocimiento del grupo que les fue asignado y se preparen para el inicio de clases, a su vez, generar las listas de asistencia para los profesores.

La etapa de altas/cambios, a los alumnos que no se les asignó grupo de laboratorio debido a que las opciones que querían estaban llenas, se les da un periodo de una semana para que asistan al laboratorio, escojan un grupo de laboratorio en el cual existan vacantes y se adapte a sus horarios. Ésta selección por parte del alumno consiste en que el encargado por parte del laboratorio de ésta etapa le muestre al alumno una lista en la cual están indicadas las cantidades de los alumnos inscritos a cada grupo, entonces, el alumno basándose en su horario de clases teóricas analiza la lista de vacantes y decide en que horario va a tomar la clase de laboratorio; El encargado del laboratorio después de que el alumno seleccionó un grupo, resta una vacante del grupo seleccionado y este proceso se repite para todos los alumnos que quedaron fuera de la asignación de grupo.

Posteriormente en la última etapa que es meramente informativa, son publicadas las fechas de inicio y fin de las prácticas del laboratorio, calendario de actividades.

### *2.1.2.-Análisis de sistemas similares*

El programa más cercano al que se pretende construir es el sistema de inscripciones de la Facultad de Ingeniería. Debido a que los alumnos de la facultad ya tienen experiencia en manejar los sistemas WEB ya sea en su correo electrónico o simplemente navegando en internet, es que la facultad emplea éste medio para realizar sus inscripciones y prestar sus servicios a la comunidad estudiantil. En éste sistema los alumnos ingresan sus datos de autenticación, tienen acceso a una sesión en la cual se despliega un menú que les permite seleccionar las asignaturas de su interés y los laboratorios que dichos alumnos pretenden cursar. El sistema reconoce al alumno creándole una sesión independiente con duración de 45 minutos, basándose en su número de reinscripción el cual se calcula en base al rendimiento del alumno durante su estancia en la facultad, el cual le da al alumno el intervalo dicho de 45 minutos para que realice su inscripción, mostrándole su avance de créditos, su promedio y las asignaturas que ya cursó, para que con esta información el alumno analice su situación académica y decida s asignaturas cursar en el próximo periodo escolar.

En éste sistema, el alumno puede monitorear las vacantes de los diferentes grupos de las asignaturas ofertadas por la facultad para el siguiente periodo escolar, así el alumno tiene conocimiento sobre el cupo en el grupo de su elección y pueda proceder a su inscripción; Si el grupo ya no tiene cupo, el sistema le indica al alumno que no es posible la inscripción, de igual forma si el grupo se traslapa con alguna otra asignatura. De ésta manera el alumno continúa ingresando las asignaturas hasta llegar a un máximo de siete y mínimo de una (en el caso específico su respectivo laboratorio); A lo largo del proceso si un alumno desea realizar una modificación, lo puede hacer presionando el botón de cambio y así, puede sustituir una asignatura que ya había inscrito por otra o simplemente cambiar el grupo en el que la había inscrito, de tal manera que al quedar satisfecho el alumno con su inscripción presiona el botón de finalizar, confirmando su inscripción e imprimiendo el comprobante de las asignaturas a cursar.

Una vez iniciado el ciclo escolar y transcurrió una semana de clases el alumno tiene un periodo de ajustes, en el cual puede cambiar de grupo, darse de baja o darse de alta de una o varias asignaturas, conservando el mismo criterio, tener inscritas máximo siete asignaturas y mínimo una. Ya finalizado éste



periodo, el alumno queda definitivamente inscrito a las asignaturas seleccionadas y el ciclo escolar da comienzo formalmente.

Al final del ciclo escolar se publican nuevamente las listas de las vacantes de próximo periodo para que el alumno seleccione las materias a inscribir, repitiendo este proceso nuevamente.

Debido a la facilidad que implica manejar grandes cantidades de alumnos con sistemas computacionales, varias facultades han optado por utilizarlos para su administración, en particular para sus respectivos procesos de inscripción, tan así que todas las facultades utilizan un sistema en el cual los alumnos realicen su inscripción vía WEB.

El sistema de inscripciones de la facultad de ingeniería es un sistema muy complejo, ya que necesita una gran estabilidad para poder dar servicio a más de 13000 alumnos cada semestre.

Ya que el laboratorio de mecánica solo atiende a los alumnos que estén cursando las asignaturas de Estática y Cinemática y Dinámica el número promedio aproximado que llega a atender por semestre es de 1700 alumnos de los aproximadamente 2400 alumnos inscritos formalmente a las dos asignaturas, lo que permite un poco más de tiempo para el proceso de inscripción y una mejor respuesta por parte del sistema, ya que dedicaremos un periodo a la etapa de pre registro, la cual evita cuellos de botella y disminuye el estrés al servidor de la facultad donde correrá el sistema. El periodo extra de tiempo del que hablamos se refiere a la etapa de pre-inscripción en la cual los alumnos escogen sus opciones de laboratorio, dando la posibilidad de colocarlos en las vacantes de los grupos, para que después ellos entren al sistema nuevamente y verifiquen que están en la opción que deseaban disminuyendo la cantidad de cambios en la siguiente etapa (altas, bajas y cambios).

## 2.2.-Selección de metodología de desarrollo

Para el desarrollo de este tipo de sistemas informáticos hay que tener en cuenta ciertos elementos de diseño e implementación para lograr que la automatización del proceso nos lleve al aumento en la eficiencia.

Debido a esto, el diseño es la etapa fundamental del desarrollo de un sistema, ya que de esta depende su eficiencia y su usabilidad. En ésta etapa, la mayoría de las metodologías de desarrollo están de acuerdo en que se debe invertir un gran esfuerzo para minimizar tiempo de desarrollo o simplemente hacerlo más fluido.

Un análisis de las necesidades del usuario y un diseño adecuado de las funciones necesarias para el sistema, así como, la tecnología empleada son igual indispensables como la metodología de desarrollo. Basándonos en las necesidades del proyecto se decidió utilizar la metodología “Modelo incremental o evolutivo” ya que la primera necesidad a cubrir fue la de ordenar las papeletas de registro de los alumnos, posteriormente se implementara el módulo de asignación automática de grupo, el de cambios y altas, el de revalidaciones, el de publicación de las listas de asistencia y los demás módulos que se necesiten basándonos en la etapa de diseño del sistema. Debido a que cada módulo puede funcionar por separado y el desarrollo puede realizarse de manera independiente para generar un producto funcional que forme parte del sistema total de inscripciones, es que se decidió utilizar ésta metodología.

### 2.3.-Selección de tecnologías

Debido a que este servicio será utilizado por una gran cantidad de alumnos de la Facultad de Ingeniería, debe tener una rápida velocidad de respuesta a las peticiones, es por eso que seleccionamos el lenguaje PHP, el cual se ejecuta del lado del servidor y es código oculto al usuario, brindando seguridad al sitio.

Ya que el diseño de este sistema se basa en el lenguaje multiplataforma PHP el cual corre del lado del servidor procesando las peticiones de los usuarios y el resultado se muestra con HTML, es que, el usuario no necesita características especiales para poder entrar al sistema y realizar sus inscripciones, más que tener actualizado su navegador de internet a la versión más actual. Otra de las razones principales por la cual se decidió utilizar el lenguaje PHP se debe a que el servidor en el que se pretende alojar el sistema, ya está configurado para esta tecnología, lo cual facilitaría la puesta en marcha del sistema. Es por esta razón que la construcción de la base de datos que se necesitará para el sistema, se va a realizar en MySQL; Ya que se tiene que almacenar toda la información de los usuarios, en este caso los más de 2900 alumnos, se necesita un sistema manejador de bases de datos eficiente,

aunque una ventaja significativa de utilizar MySQL, es que es de uso libre y tiene una conexión fácil de implementar con PHP.

#### 2.4.-Identificación de procesos

En este caso, desde el punto de vista del laboratorio, el proceso de inscripciones actualmente consta de varias etapas (las cuales se pretenden automatizar con el desarrollo del sistema de inscripciones vía WEB), dichas se enumeran a continuación para su análisis y para identificar los principales procesos que se necesitan automatizar:

- a. Publicación de listas de horarios para el próximo semestre en la vitrina del laboratorio.
- b. Emisión de papeletas para que los alumnos, proporcionen sus opciones de grupo de laboratorio de estática o de cinemática y dinámica.
- c. Recepción y ordenamiento de las papeletas según el número de reinscripción de los solicitantes.
- d. Carga de la tabla que contiene la información de los alumnos como:
  - Número de cuenta
  - Nombre
  - Número de reinscripción
  - Asignaturas a las que se encuentra inscrito
- e. Asignación manual de grupo de laboratorio a cada alumno.
- f. Publicación de resultados de la asignación de grupos por medio de listas.
- g. Realizar los cambios de grupo y altas a alumnos que no se les asigno el grupo de laboratorio.
- h. Impresión de listas de asistencia para los profesores.
- i. Dar información importante a los alumnos para que la tengan en cuenta.

Basándonos en éstas etapas podemos saber las funciones básicas que tendrá el nuevo sistema vía WEB.

Se lleva a cabo la planeación del semestre, después se genera una lista para que los alumnos conozcan los grupos disponibles, por tanto se necesita un módulo de información que se actualice automáticamente desde la base de datos, tomando la información precargada de los grupos.

Posteriormente el alumno se registrará con su número de cuenta y número de reinscripción previamente cargados en la tabla que contendrá la información del alumno para autenticarlo y pueda ingresar sus opciones de laboratorio, quedando en espera del resultado de la asignación automática de grupo. Éste periodo de registro tendrá una duración de una semana, en la cual los alumnos que deseen revalidar el laboratorio, tendrán la posibilidad de realizarlo, claro está, siempre y cuando la calificación que obtuvieron haya sido aprobatoria; Este proceso se va implementar a través de un formulario en el cual los alumnos colocarán el semestre en el cual cursaron el laboratorio, el grupo, así como su número de cuenta y será independiente del resto del proceso de inscripciones como se detallará más adelante.

El siguiente proceso será transparente para el alumno desde el punto de vista que el alumno no lo realizará; Éste proceso se va ejecutar automáticamente una vez finalizada la etapa de registro asignando automáticamente el grupo a los solicitantes, tomando en cuenta las restricciones en que se basa este proceso, es decir que no puede haber más de 25 alumnos por grupo, respetando el orden del número de inscripción de los alumnos de menor a mayor, así como, el orden de sus opciones.

Después se publicará la lista resultados de la asignación automática y durante la siguiente semana el alumno podrá consultar su resultado y realizar los cambios que necesite para quedar en el horario más conveniente para él. Así mismo, a los alumnos que no se les asignó grupo de laboratorio en el proceso anterior, se les mostrará una lista actualizada en tiempo real de las vacantes disponibles en los diferentes grupos, brindándole la libertad de escoger un grupo de laboratorio, que se adapte a sus necesidades.

Por lo tanto, el resto del proceso de inscripciones será el imprimir las listas de asistencia para los profesores, dando comienzo el siguiente ciclo escolar.

## 2.5.-Diseño del sistema

Con el análisis anterior, podemos proceder al diseño en diagrama de flujo de cada uno de los módulos del sistema. En éste proceso de diseño del sistema se analizará la viabilidad incluir un módulo de validación de la inscripción el cual tiene el objetivo de verificar la información de los alumnos, ya que se ha detectado que al final del semestre los alumnos no reciben su calificación, debido a que al inicio del ciclo escolar, realizan cambios de grupo de teoría sin avisar al laboratorio de dichos cambios, provocando que su calificación sea enviada a otro profesor y que el alumno tenga que tramitar una constancia de que curso el laboratorio.

### *2.5.1.-Diseño de los módulos del sistema*

En base a los requerimientos hemos identificado los siguientes módulos o procesos principales que conformarán el sistema de inscripciones del Laboratorio de Mecánica.

**Módulo 1:** Consulta de grupos y horarios programados para el semestre

**Módulo 2:** Registro a Laboratorios y Revalidación de calificación

**Módulo 3:** Publicación de asignación de grupos

**Módulo 4:** Cambio y alta de laboratorio

**Módulo 5:** Informativo de fechas importantes como el inicio de clases

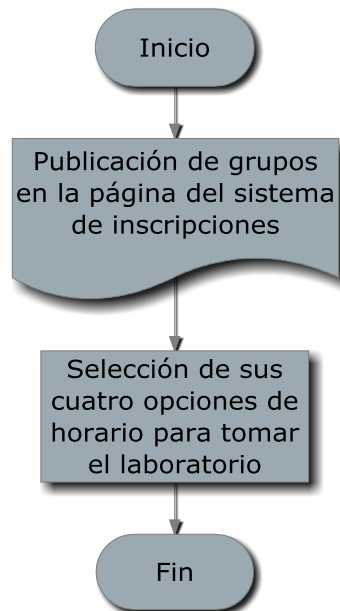
**Módulo 6:** Validación de inscripción

Con ellos se cumplen los objetivos de las necesidades del alumno como usuario final, pero se necesita un módulo extra para el rol del administrador del sistema, esto es, la persona encargada de solicitar al sistema los reportes, las listas de asistencia, vacantes, monitoreo de los grupos, estadísticas y en general su administración. A continuación se desarrollarán los diagramas de flujo de los módulos mencionados anteriormente.

#### ***Módulo 1: Consulta de grupos y horarios programados para el semestre.***

El objetivo de éste módulo es meramente informativo. El sistema se conecta a la base de datos del laboratorio cargando automáticamente los grupos con su respectiva información. El papel de los alumnos en éste módulo es el de entrar al sistema para consultar los horarios y seleccionar sus opciones de

laboratorio, para después pasar al módulo de registro. Estos datos de los diferentes grupos de laboratorio estarán precargados por el administrador del sistema a quien se los proporciona el coordinador una vez que ha realizado la planeación del próximo semestre.



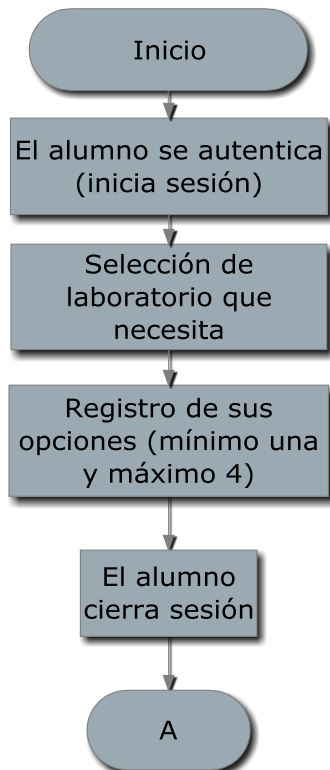
### ***Módulo 2: Registro a Laboratorios y Revalidación de calificación***

En este módulo los alumnos entran a realizar su registro de opciones para la asignación automática de grupo. Antes de que se habilite este módulo ya están cargados en la base de datos del laboratorio, los datos del alumno como su número de cuenta, su número de reinscripción, las asignaturas a las que está inscrito sólo en el caso de que se haya inscrito a estático o a cinemática y dinámica y el grupo respectivo al que esté inscrito. Estos datos son solicitados al administrador del sistema de inscripciones de la Facultad de Ingeniería.

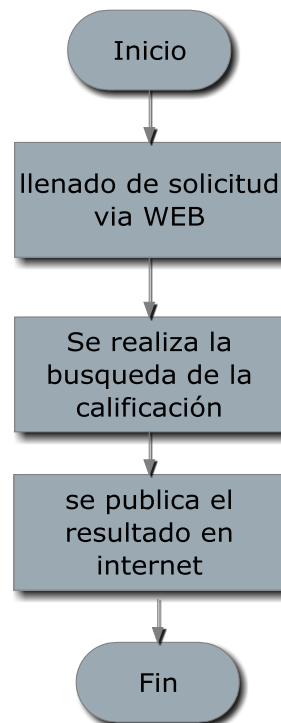
Después de consultar los grupos disponibles en el módulo anterior el alumno entrará al sistema por medio de una autenticación la cual generará una sesión personalizada para cada alumno y le permite ver sólo la información referente a sus trámites dentro del laboratorio. El sistema realizará las validaciones de todos los formularios que se muestran en la pantalla y en este caso que el número de cuenta sea sólo numérico así como las opciones de grupos. En el transcurso del semestre en que se inició con el proyecto se aprobó una propuesta de revalidación del laboratorio, la cual implica que los alumnos que

aprobaron el laboratorio en trimestres pasados (no más de dos trimestres atrás) pueden solicitar que se les revalide su calificación para no tener que volver a cursar el laboratorio. En el siguiente diagrama de flujo se muestra el proceso que sigue el alumno en ésta etapa.

#### Solicitud de laboratorio (Alumno)



#### Revalidación de laboratorio (Alumno)



El siguiente proceso, es el que asigna los grupos de laboratorio a los alumnos registrados, siendo transparente para el alumno ya que solo ve el resultado del mismo. El proceso de asignación de grupos en el sistema actual sirve como mecanismo para llenar los grupos de manera homogénea, ya que primero se ordenan las papeletas de menor a mayor número de inscripción, después se toma las opciones de cada papeleta hasta encontrar una con cupo y se comienza a llenar la tabla de alumnos inscritos a cada laboratorio; A grandes rasgos el proceso es eficiente ya que permite controlar el llenado de los grupos de manera más homogénea, la única desventaja es que es un proceso muy largo para realizarlo a mano y es por la misma razón que se decidió

simplemente automatizar este proceso. El siguiente diagrama muestra el algoritmo con el cual se asignan los grupos de laboratorio a los alumnos.

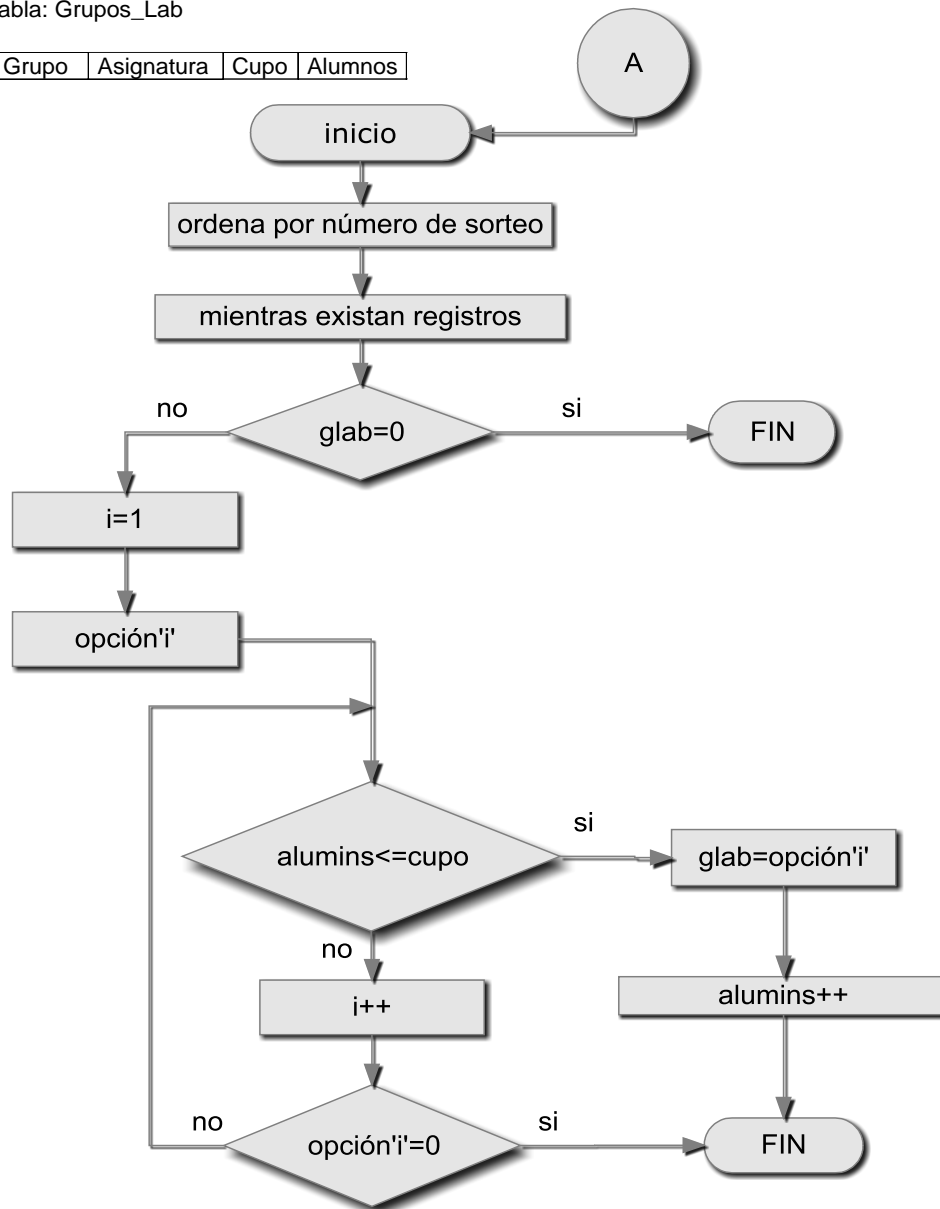
### Proceso de asignación de grupos

Tabla: Mecánica\_dcb

# Cta.	Asignatura	opción1	opción1	opción2	opción3	opción4
--------	------------	---------	---------	---------	---------	---------

Tabla: Grupos\_Lab

# Grupo	Asignatura	Cupo	Alumnos
---------	------------	------	---------





### ***Módulo 3: Publicación de asignación de grupos.***

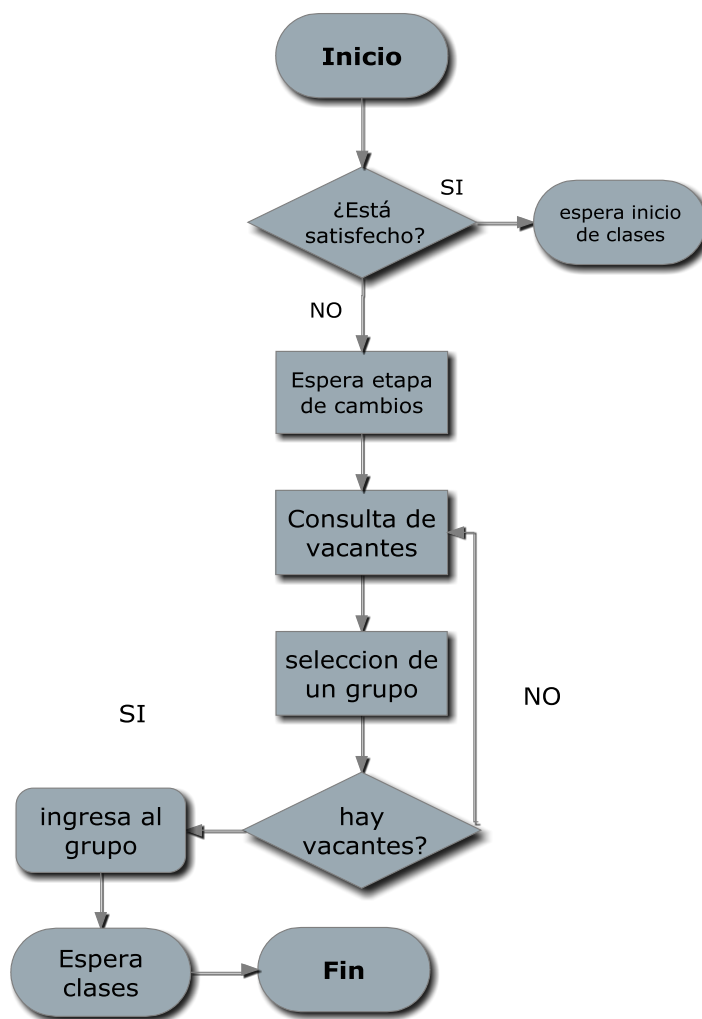
En este módulo, el alumno entra al sistema autenticándose únicamente con su número de cuenta, para consultar el resultado de la asignación automática de grupo; En éste caso no puede modificar el registro por tanto no se necesita tanta seguridad, es por eso que no se les pide su número de reinscripción para la autenticación. En caso de que no esté satisfecho con el grupo que le fue asignado entrará al módulo siguiente para realizar los cambios convenientes, se le solicita ingresar número de cuenta y su número de reinscripción.



### ***Módulo 4: Cambio y alta de laboratorio***

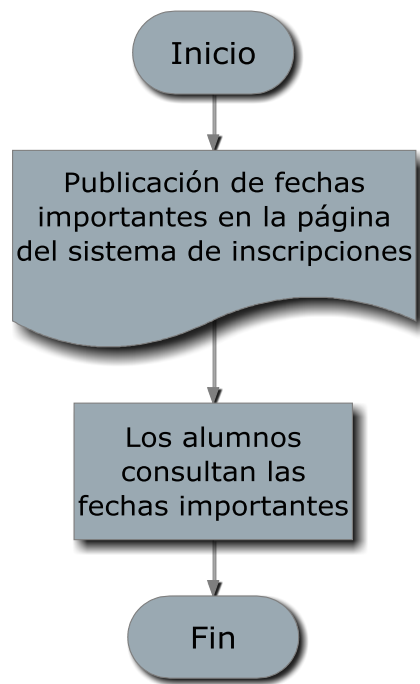
Éste módulo será el encargado de hacer cambios en los grupos asignados, si algún alumno no pudo registrarse a tiempo en consecuencia no tiene grupo de laboratorio, lo pueda obtener registrándose y seleccionando uno de los grupos que tenga vacantes. En éste caso el sistema atenderá a los alumnos conforme vayan llegando durante el periodo de cambios y altas que dura una semana; A diferencia del módulo de asignación de grupo que se basa en el número de reinscripción para atender las peticiones de los alumnos, en éste el alumno entrará en cualquier momento de la semana para realizar el cambio o alta del laboratorio. Una vez que el alumno se registre, tendrá acceso a las

listas de vacantes en tiempo real de todos los grupos del laboratorio en cuestión lo cual le permitirá estar pendiente de algún grupo en particular para realizar su alta o su cambio. En el proceso de inscripciones no está de manera explícita la opción de baja de laboratorio, pero en caso de que un alumno necesite darse de baja por alguna causa, puede acudir al laboratorio con el coordinador para explicar las razones por las que necesita darse de baja del mismo, a consideración de éste, se procede a dar de baja al alumno por parte del administrador del sistema.

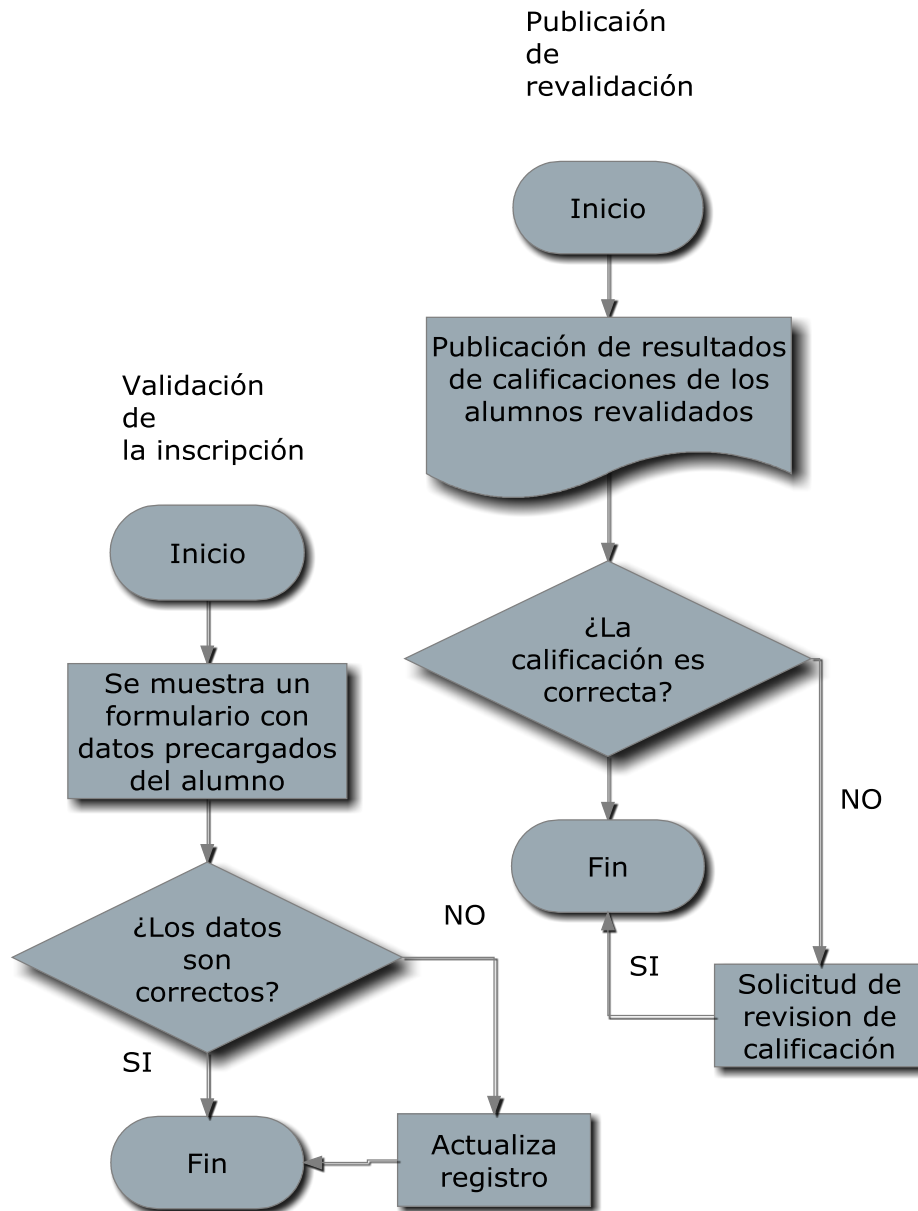


### ***Módulo 5: Informativo de fechas importantes como el inicio de clases***

El módulo 5, se encargará de dar a los alumnos la información más importante referente al laboratorio, como el inicio y fin de clases. Ésta información estará cargada previamente en la base de datos del sistema.



### Módulo 6: Validación de inscripción



#### 2.5.2.-Diseño del panel de Administrador

El administrador del sistema será en encargado de:

##### **Planeación:**

Dar de alta en el sistema los grupos de laboratorio disponibles, resultado de la planeación por parte del coordinador del laboratorio.

**Configuración:**

Coloca las fechas de inicio y fin de las diferentes etapas del sistema.

Coloca el semestre en curso.

Configura los valores predeterminados del sistema.

Activa la asignación automática de grupos.

**Monitorear:**

Monitorea el llenado de los grupos y analiza la posibilidad de abrir más vacantes en ciertos grupos.

**Imprimir:**

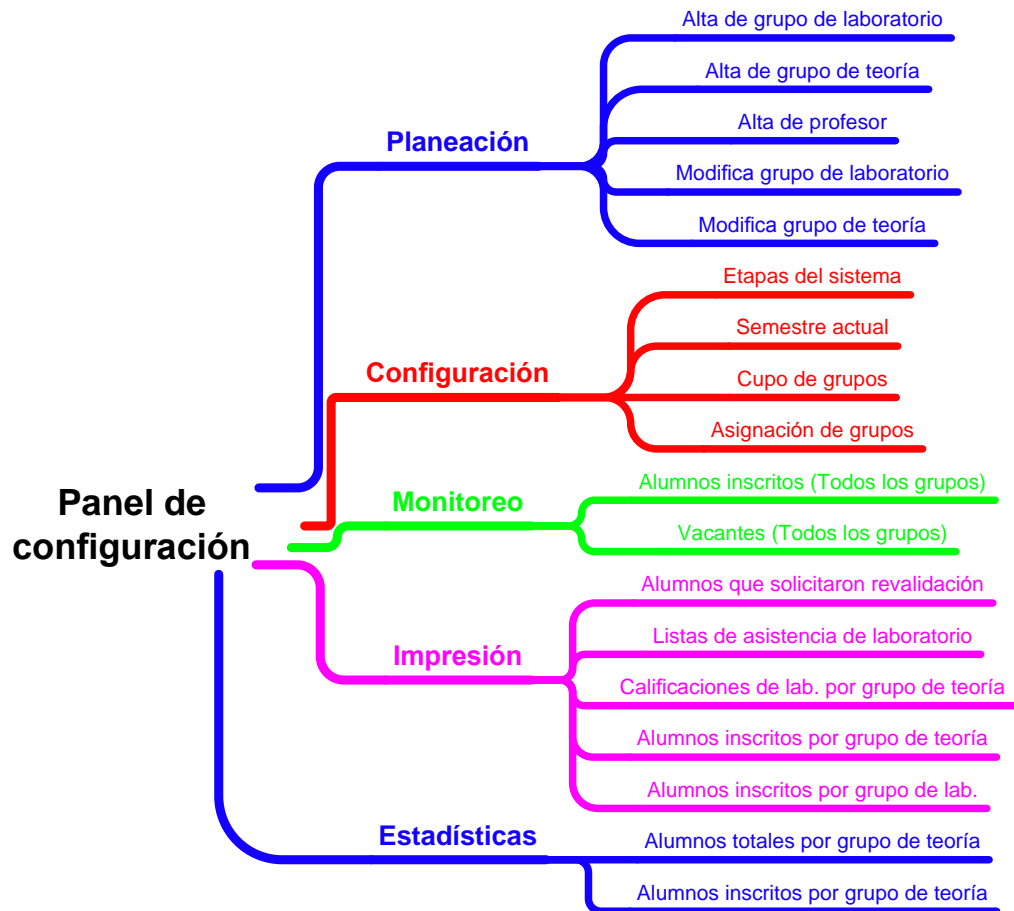
Imprime las listas de asistencia para los profesores.

Imprime los reportes de la cantidad de alumnos inscritos por grupo de laboratorio.

**Estadísticas:**

Genera los reportes que el coordinador necesite para analizar el comportamiento del periodo de inscripciones.

A continuación se presenta un diagrama de las principales funciones del administrador del sistema:



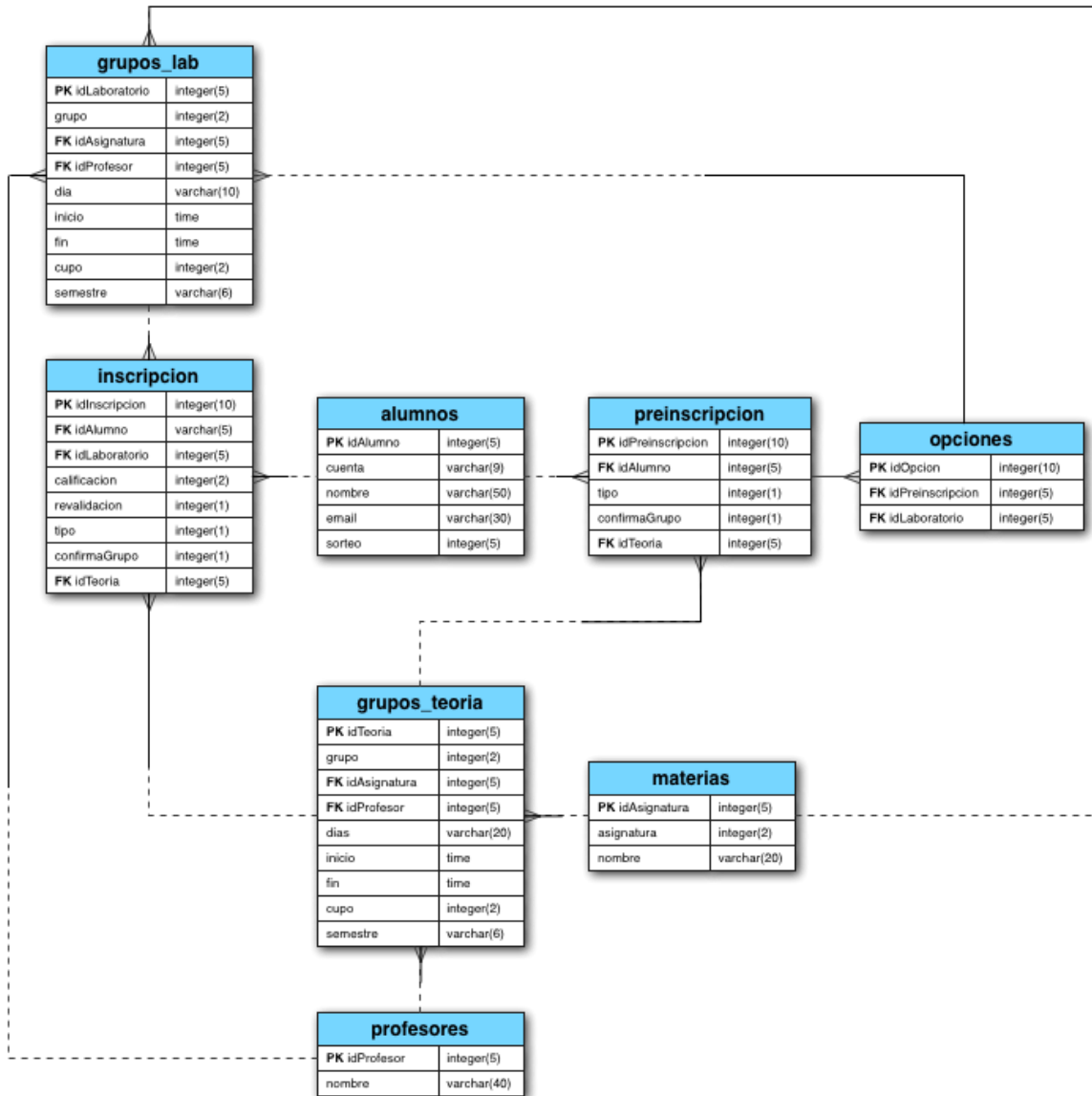
## 2.6.-Diseño de la base de datos

### Diagrama E-R

Basándonos en los requerimientos del sistema que recolectamos por parte del coordinador del laboratorio de mecánica y con los cuales diseñamos el sistema de inscripciones, también trabajamos para el diseño del diagrama entidad relación de la base de datos que se encargara de almacenar la información.

usuarios	
PK idUsuario	integer(4)
usuario	varchar(10)
contrasena	varchar(40)
nombre	varchar(10)
aPaterno	varchar(10)
aMaterno	varchar(10)
email	varchar(30)
tipo	integer(1)

configuracion	
PK id	integer(4)
semestre	varchar(6)
etapa	varchar(1)
cupoPredLab	integer(2)



Los datos de la tabla alumnos son solicitados al administrador del sistema de inscripciones de la facultad para facilitar la autenticación de los alumnos haciendo más ágil el proceso de registro, al menos para los alumnos regulares, ya que los alumnos oyentes y alumnos que realizaron cambios o altas en su inscripción curricular, se les pide que llenen un formulario para tener esta información y realizar su registro en la tabla de alumnos, es por esa misma razón que el laboratorio les asigna un número de reinscripción local (solo funciona para el laboratorio) para el proceso de asignación de grupo. Éste número se asigna calculando cuál es el número de reinscripción más alto y a partir de ahí se les asigna dicho número conforme van entrando los alumnos a llenar el formulario; Esto es para dar prioridad a los alumnos regulares.

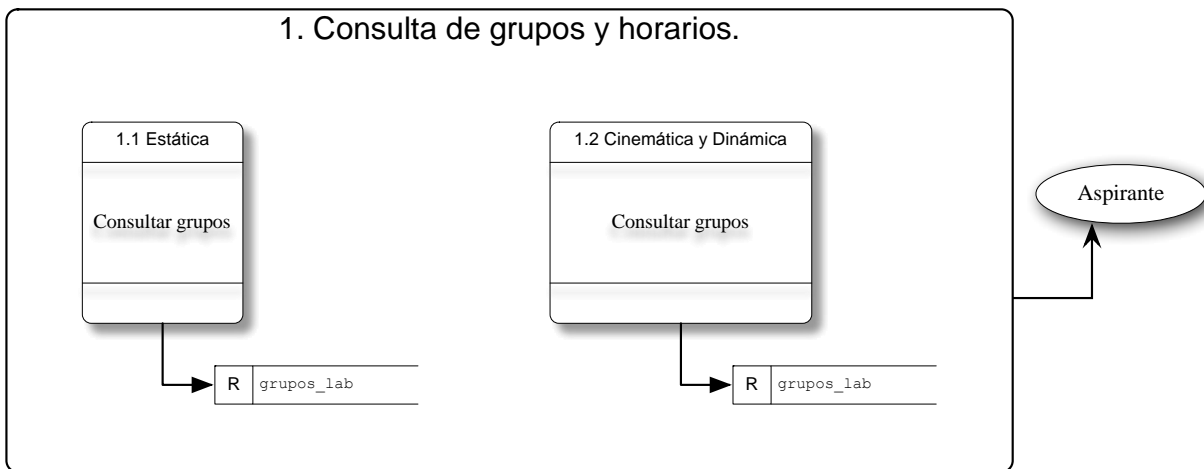
### Etapas del sistema de inscripción



En los siguientes diagramas se muestran las consultas que se realizan a la base de datos en los diferentes módulos del sistema:

### Módulo 1

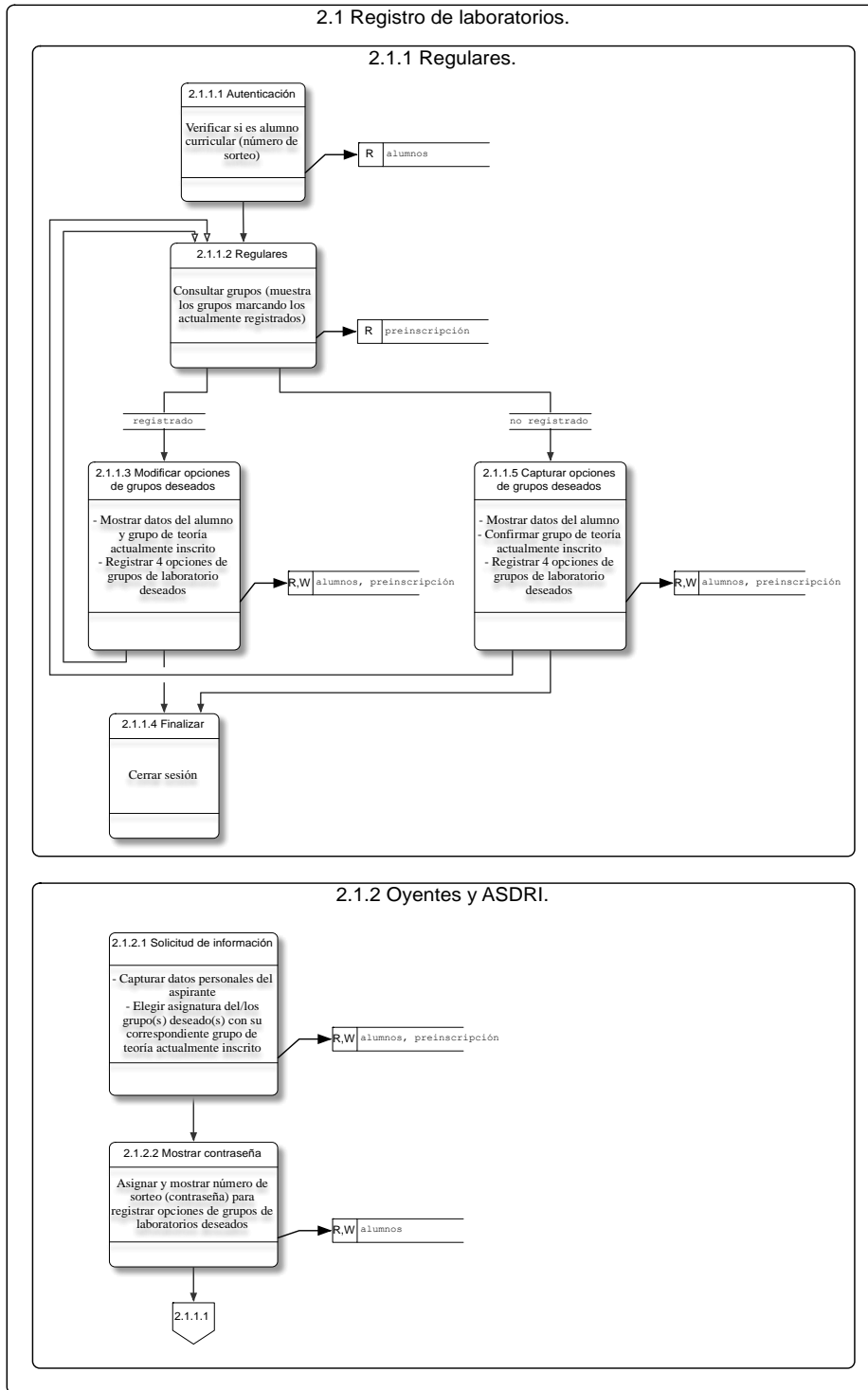
Consulta de grupos y horarios Estática  
Cinemática

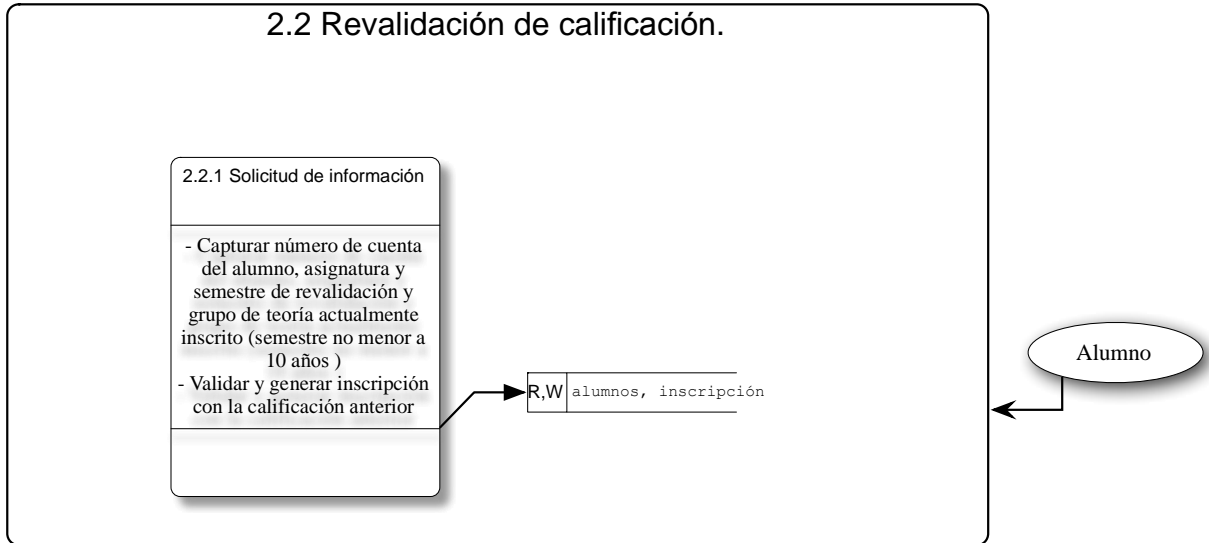


### Módulo 2

Registro de laboratorios Regulares  
Oyentes y ASDRI  
 Revalidación de calificación Revalidar



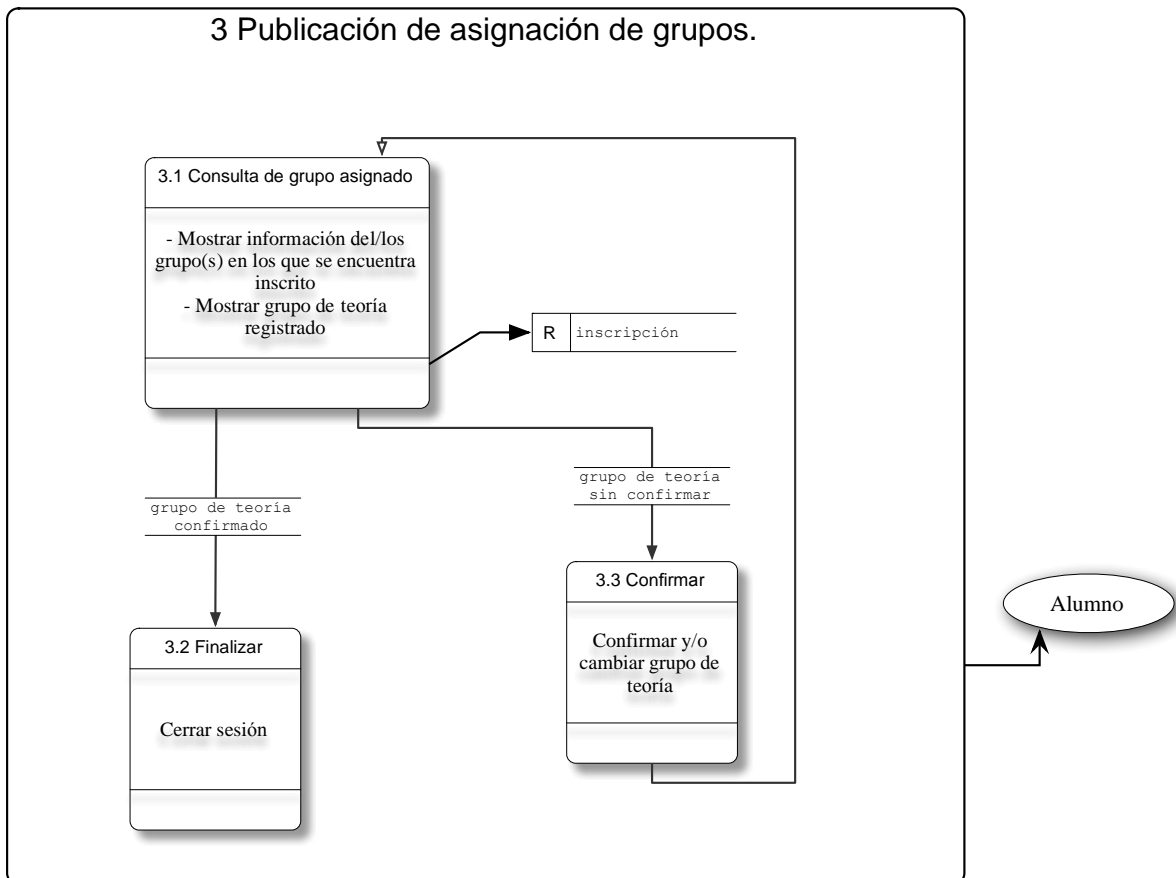




### Módulo 3

#### Publicación de asignación de grupos

#### Consulta de grupo asignado

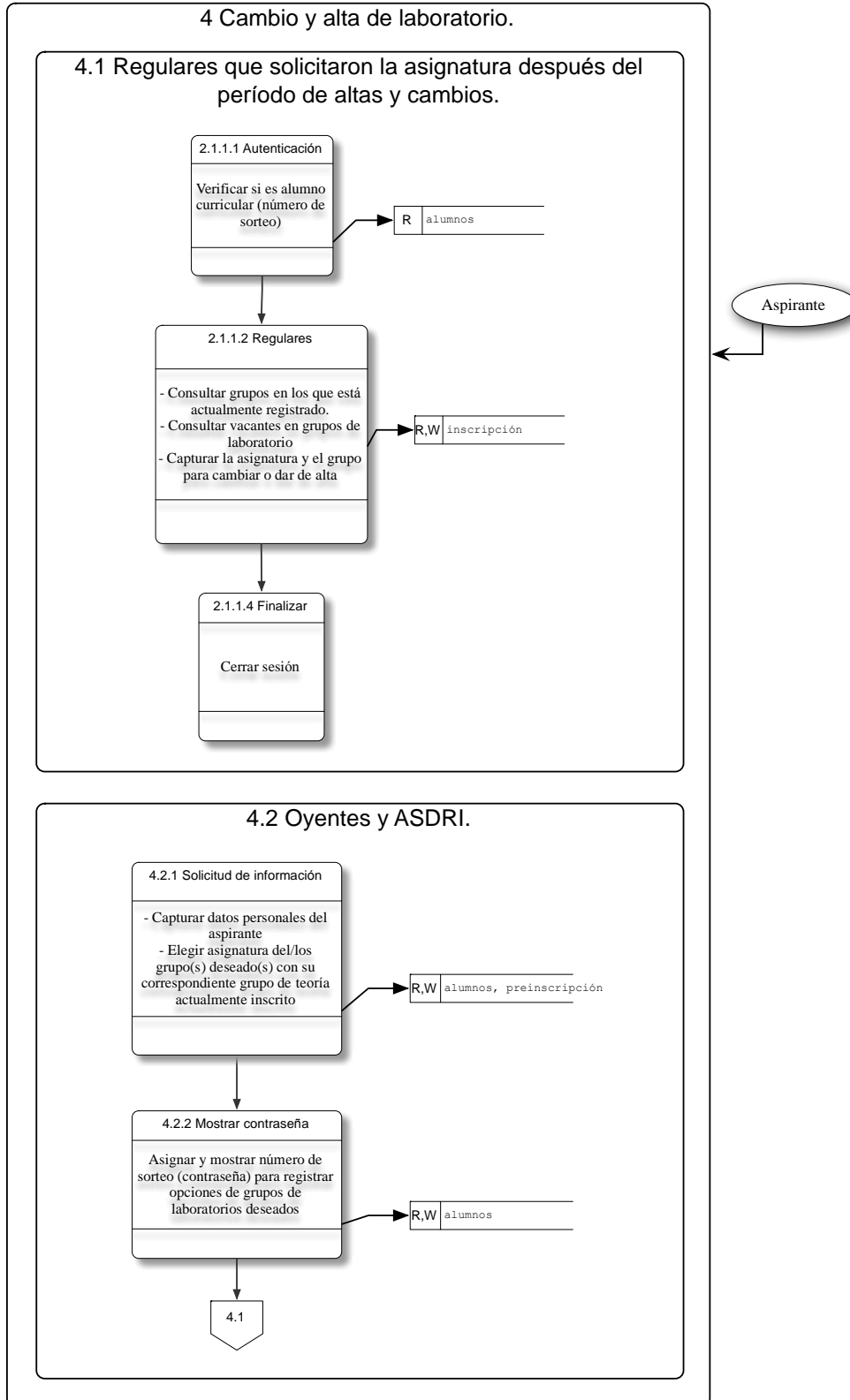


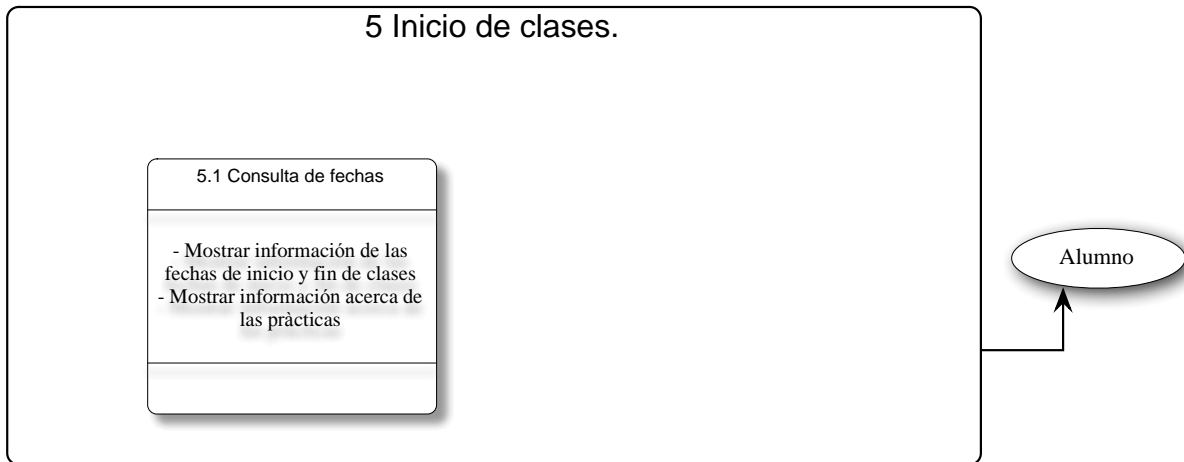
**Módulo 4**

**Cambio y alta de laboratorio**

Regulares que solicitaron la asignatura después del periodo de cambios

Oyentes y ASDRI



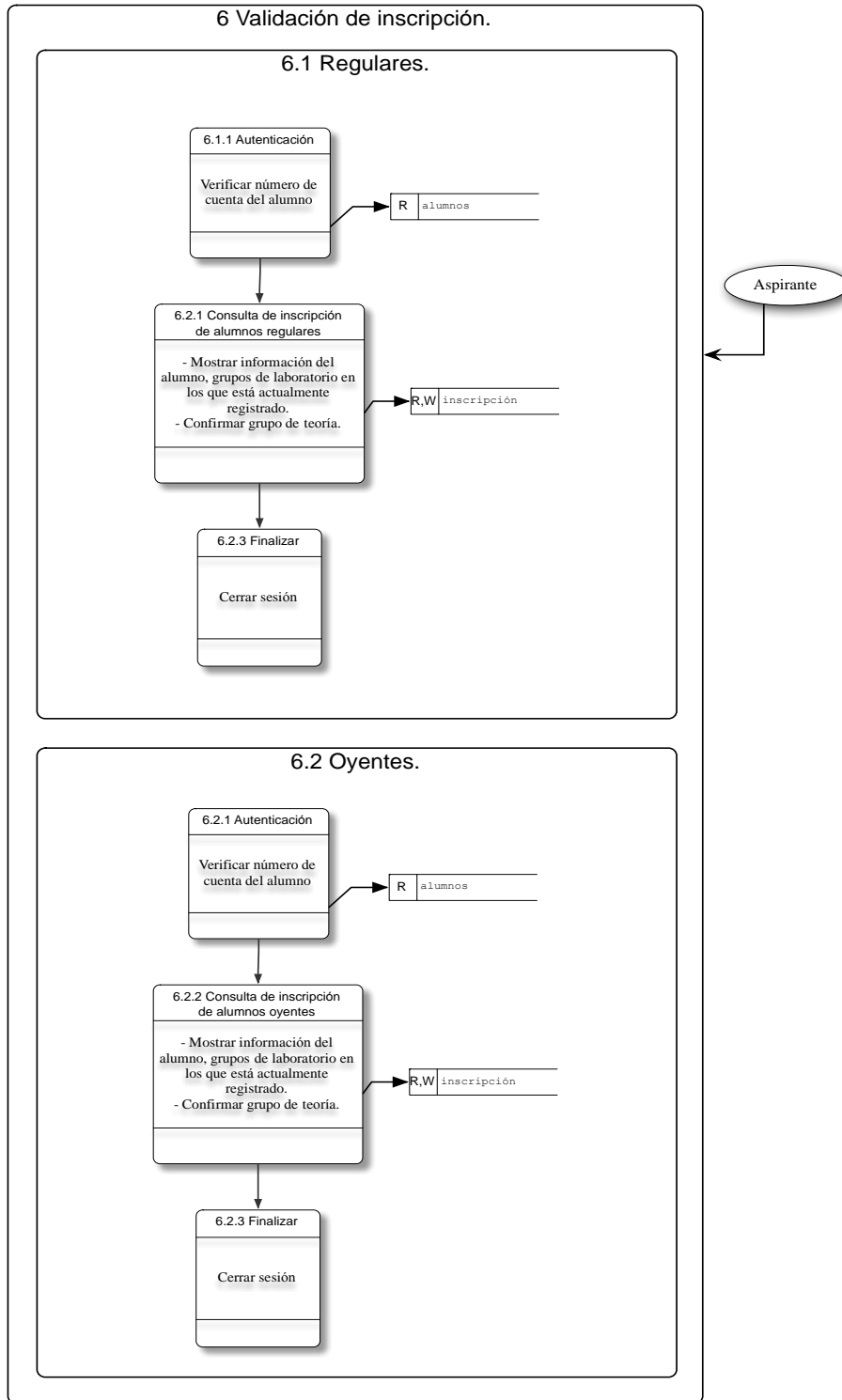
**Módulo 5****Inicio de clases**[Sitio web del laboratorio de mecánica](#)

**Módulo 6**

Valida tu inscripción (alumnos regulares y oyentes)

**Validación de inscripción**

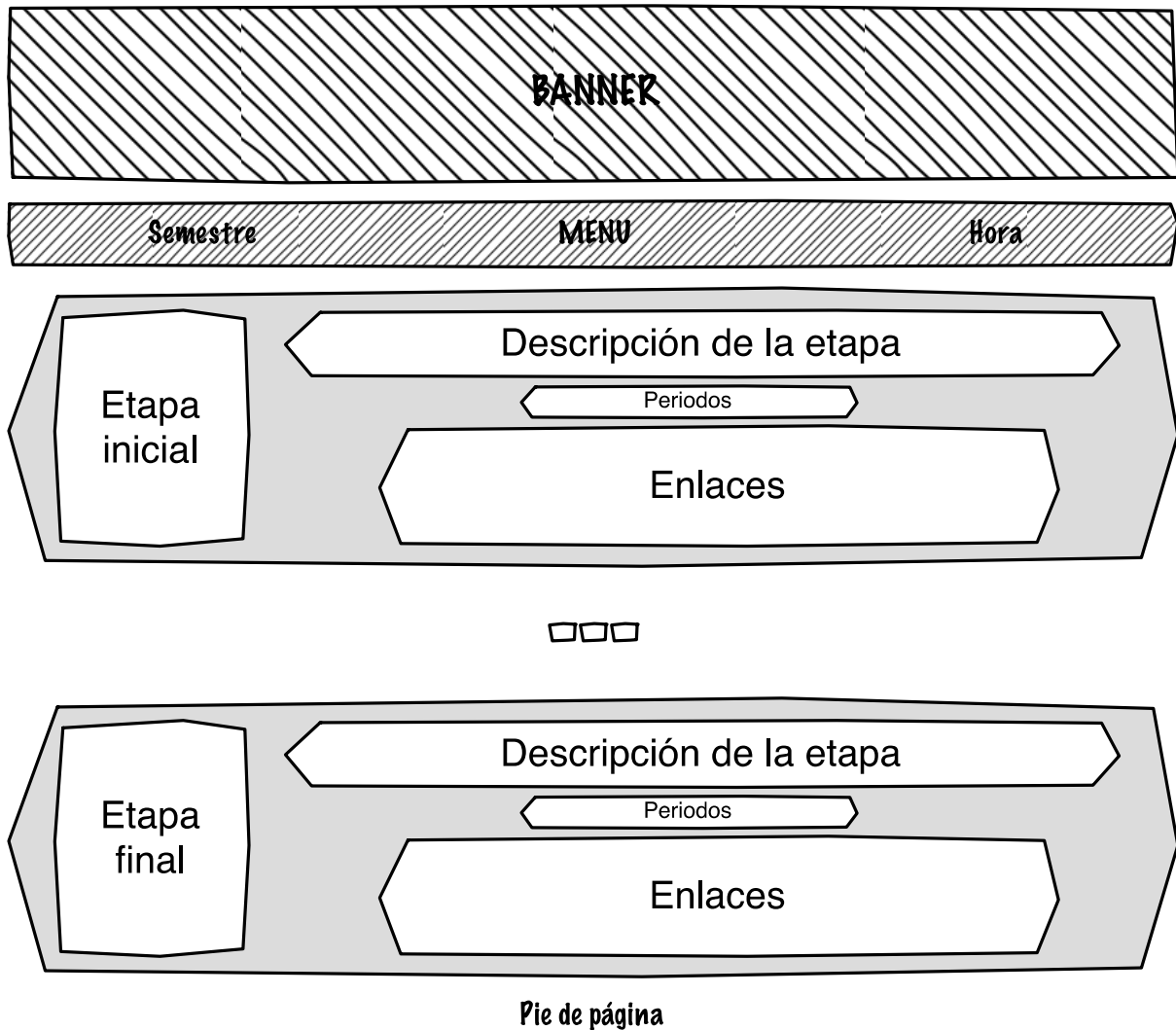
Valida tu inscripción (alumnos que revalidaron)



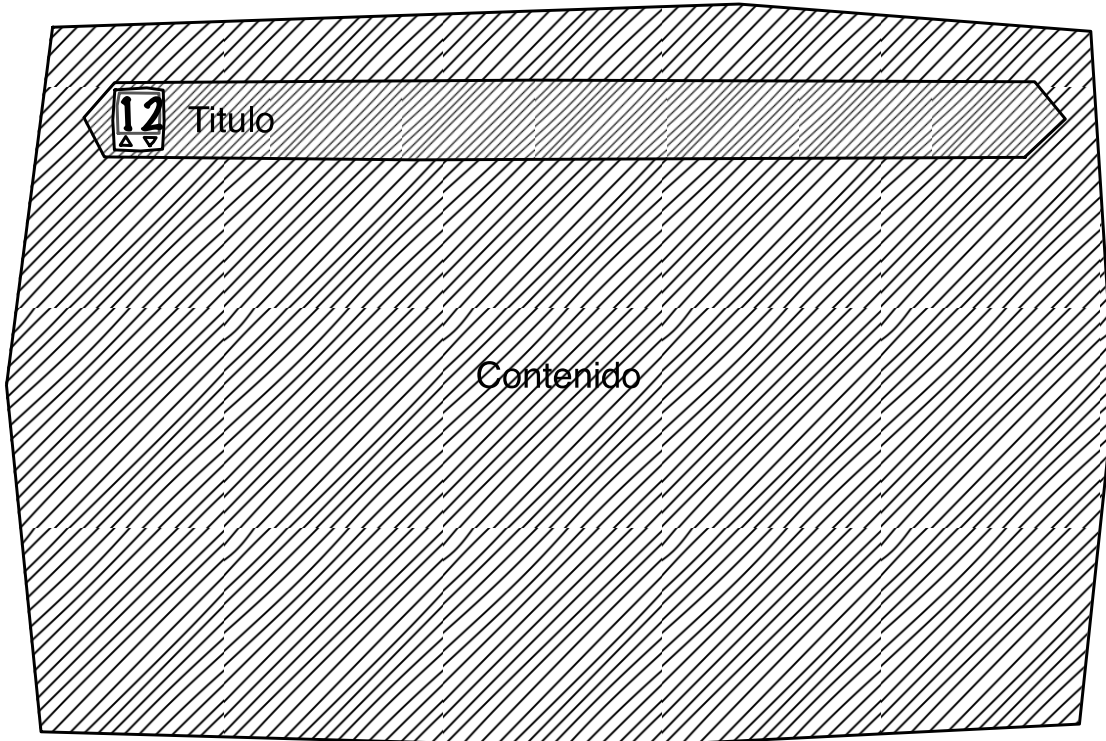
## 2.7.-Diseño de la interfaz de usuario

### ***Página principal***

El diseño de la interfaz de usuario, está basada en un diseño similar al de los correos electrónicos, esto se refiere a que cuenta con una distribución de los botones y ligas importantes de una manera similar a la ya vista en su correo electrónico, simplificando y haciendo más intuitivo el sistema para los alumnos.

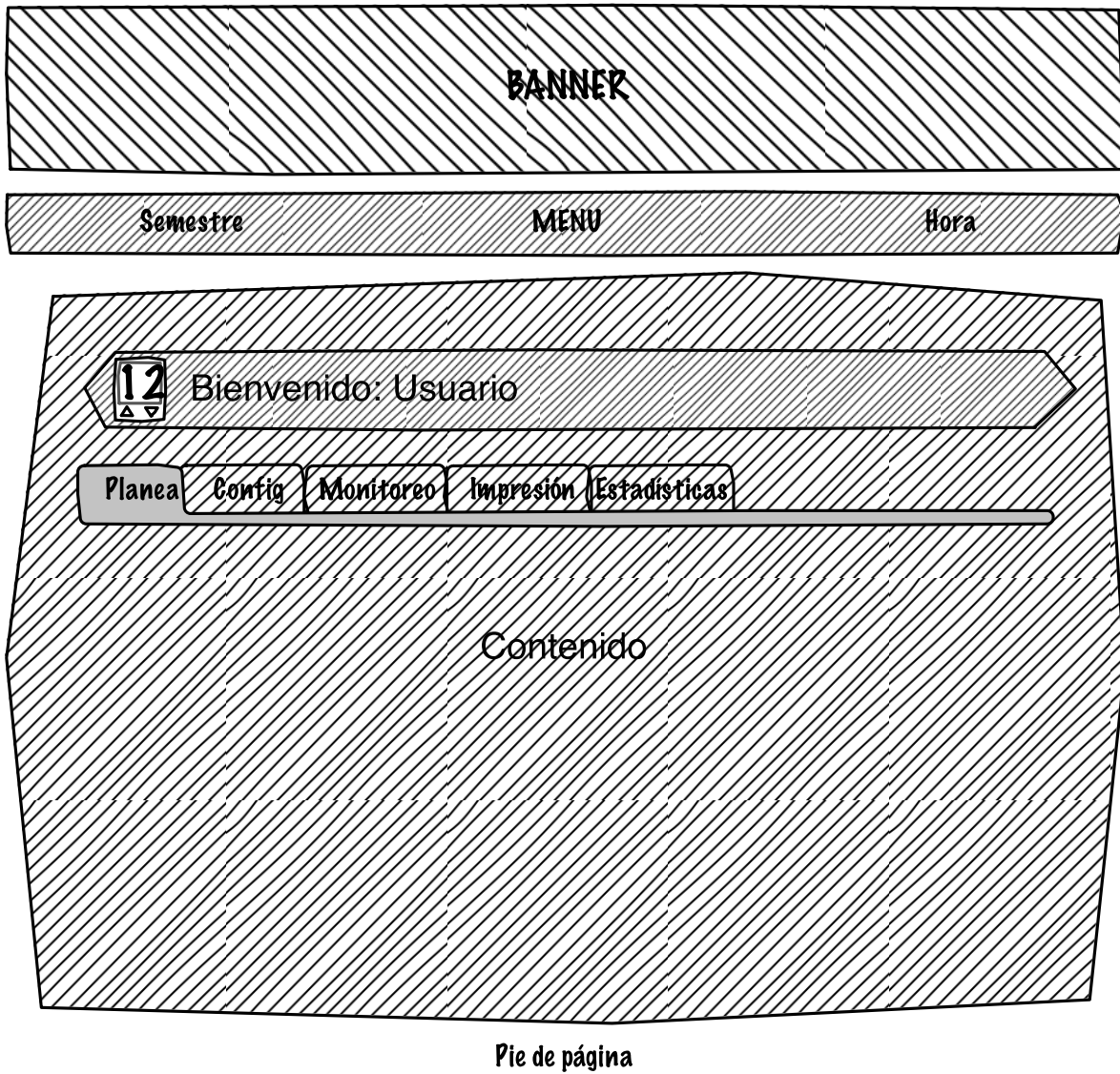


**Páginas de contenido (usuario)**



**Pie de página**

### *Página de administración*





# **CAPÍTULO 3**

## **PRUEBAS, PUESTA EN MARCHA Y MANTENIMIENTO**



### 3.- Pruebas, puesta en marcha y mantenimiento

#### 3.1.-Pruebas

Las pruebas se realizaron localmente con una muestra de 100 alumnos seleccionados conforme llegaban a pedir informes sobre las inscripciones al laboratorio de mecánica. Éstas pruebas nos ayudaron a localizar las fallas en el flujo atreves de las diferentes etapas del proceso. Durante las pruebas se presentaron algunas fallas que serán corregidas para la puesta en marcha. A continuación se explicará cómo se hizo la implementación y las pruebas del nuevo sistema.

Las pruebas del nuevo sistema se hicieron durante el periodo de inscripciones 2013-1 en agosto de 2012, aunque las capturas de pantalla utilizadas son del sistema ya en funcionamiento para ilustrar la versión final del sistema de inscripciones; Éstas consistieron que a los alumnos que llegaban a pedir informes sobre los grupos de laboratorio, se les pedía que realizaran su registro en las computadoras del laboratorio bajo la observación del coordinador del laboratorio y los desarrolladores del sistema; La sola observación del proceso hizo evidentes algunas fallas dentro del flujo, las cuales se explicaran a continuación. Cuando el alumno entró a la página del Sistema de inscripciones, esta página es la que apareció en su navegador<sup>1</sup>:

---

<sup>1</sup><http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/CienciasAplicadas/LaboratorioMecanica/alumnos/index1.php>



Universidad Nacional  
Autónoma de México



Sistema de Inscripción del  
Laboratorio de Mecánica

SEMESTRE: 2013-2    INICIO    PLANEACIÓN    FI    DCB    15:44:45

1

Consulta de grupos y horarios programados para el semestre 2013-2

Estática
Cinemática y Dinámica

2

**Registro a Laboratorios**  
Del: 28 de enero de 2013  
al: 1º de febrero de 2013

ALUMNOS:

- Regulares
- Regulares que solicitaron la asignatura después del periodo de altas y cambios
- Oyentes y ASDRI

---

**Revalidación de calificación**  
Del: 28 de enero de 2013  
al: 8 de febrero de 2013

- Revalidar

3

**Publicación de asignación de grupos**  
5 de febrero de 2013

- Consulta de grupo asignado

4

**Cambio y alta de laboratorio**  
Del: 5 de febrero de 2013  
al: 8 de febrero de 2013

- Regulares
- Regulares que solicitaron la asignatura después del periodo de altas y cambios
- Oyentes y ASDRI

5

**Inicio de clases**  
Primera semana: 11 de febrero de 2013  
Segunda semana: 18 de febrero de 2013

según el grupo que se trate

En el sitio [web](#) del laboratorio podrán consultar o descargar los formatos de las prácticas

6

**Validación de inscripción**  
Del: 1º de marzo de 2013  
al: 1º de junio de 2013

- Valida tu inscripción (regulares y oyentes)
- Valida tu inscripción (alumnos que revalidaron)

Dudas: [laboratoriomecanica@gmail.com](mailto:laboratoriomecanica@gmail.com) o Acuda al Laboratorio de Mecánica.  
 Copyright © 2011 Facultad de Ingeniería. All Rights Reserved.

Posteriormente el alumno durante la primera etapa entró a la opción número uno:



Dependiendo del laboratorio que necesitaba el alumno, podía entrar individualmente a consultar las listas de grupos de Estática o Cinemática y dinámica, para después proceder a su registro de opciones dependiendo de la situación curricular del alumno en la materia:



En el caso de los alumnos regulares, son re direccionados directamente al registro de las opciones, pero en el caso de los alumnos oyentes y los extemporáneos se les muestra un formulario en el cual se capturan sus datos personales que no se tenían debido a que no estaban en la lista que se solicitó

al administrador del sistema de inscripciones de la facultad y se les asigna un número de reinscripción local para como ya se explicó en el diseño de la base de datos.

En general y en cada etapa, después de transcurrida la participación de cada alumno, se le preguntaba ¿Cuál era su experiencia usando el sistema de inscripciones?, se anotaban todas las observaciones del alumno. Una observación tomada, fue que la letra de los textos de las indicaciones era muy pequeña, por lo que se procedió al aumento de su tamaño en toda la página. En el transcurso de la semana de registro, se les pidió a 100 alumnos que utilizaran el sistema para analizar el flujo del proceso; Al finalizar cada intervención se le indicaba al alumno que tenía que regresar al laboratorio para continuar con él proceso. Una vez concluida ésta semana se procedió a la asignación automática de grupos de laboratorio, la cual es ejecutada con un botón en el panel de administración:

Universidad Nacional Autónoma de México  
UNAM

Sistema de Inscripción del Laboratorio de Mecánica

SEMESTRE: 2013-2 INICIO PLANEACIÓN FI DCB 17:21:14 CERRAR SESIÓN

2013 03 Mar Bienvenido(a): raul

PLANEACIÓN CONFIGURACIÓN MONITOREO IMPRESIÓN ESTADÍSTICAS

Etapa actual del proceso: Validación

Semestre: 2013-2

Cupo predeterminado para los laboratorios: 15

Guardar configuración

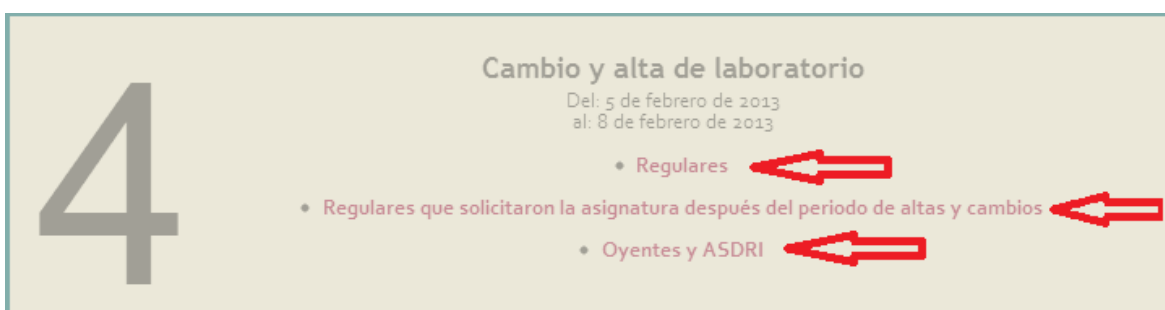
Ejecutar la asignación de grupos Ingresa la clave para ejecutar Ejecutar

Una vez realizado el proceso de asignación, se activó el periodo publicación de resultados y el de altas y cambios.

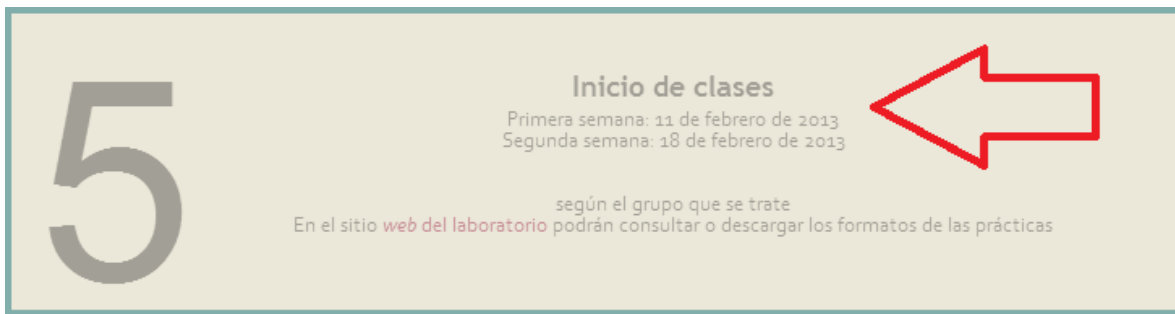
En el módulo de publicación de resultados, el alumno ingresó su número de cuenta, para después ver el resultado de la asignación de grupo. En este caso el alumno tiene dos opciones, puede estar de acuerdo con el grupo que le fue asignado, el cual seguramente está entre las cuatro opciones que él eligió y esperar el inicio de las clases, en caso contrario, ya sea que algún cambio en su horario curricular ocasione que el horario que le tocó ya no le sea viable o que no se le haya asignado grupo de laboratorio, puede pasar al módulo siguiente para realizar cambios o altas de grupo.



En este caso se puede optar por tres opciones como en el módulo 2, dependiendo de la situación del alumno en cuestión. Los alumnos seleccionados, llegaron al laboratorio para continuar con el proceso de inscripción, cada uno dependiendo de su caso entro a la opción que le correspondía, los alumnos regulares a los que no se les asignó grupo, entraron a la primera opción en la cual se les pidió su número de cuenta y su número de reinscripción y un vez dentro realizaron el cambio o la alta según el caso:



Una vez satisfechos los alumnos con sus grupos de laboratorio, sólo les queda esperar el inicio de clases y seguir consultando periódicamente la página del laboratorio de mecánica para bajar los formatos de las prácticas, llenar encuestas de calidad y simplemente estar atentos a cualquier anuncio oficial emitido por el laboratorio.



**5**

**Inicio de clases**  
Primera semana: 11 de febrero de 2013  
Segunda semana: 18 de febrero de 2013

según el grupo que se trate  
En el sitio [web del laboratorio](#) podrán consultar o descargar los formatos de las prácticas

En el módulo seis se trata de validar la inscripción, esto se refiere a que los alumnos frecuentemente realizan cambios de grupo de teoría sin avisar al laboratorio, lo cual genera una serie de peticiones de corrección de grupo al final del semestre, debido a que una vez finalizado el periodo escolar las calificaciones son enviadas a los profesores de teoría para que ellos las usen como más les convenga y si un alumno ingresó un grupo de teoría y después se cambió, sin avisar al laboratorio, su calificación le llegará al profesor del primer grupo que se tenía registrado, entonces éste acude al laboratorio a pedir una constancia de que cursó el laboratorio y la calificación que obtuvo. Es por eso que se implementó el módulo de validación de calificación. Una vez dentro, se le muestra al alumno los datos que se tiene de él, haciéndole énfasis en que revise su grupo de teoría.



**6**

**Validación de inscripción**  
Del: 1º de marzo de 2013  
al: 1º de junio de 2013

- Valida tu inscripción (regulares y oyentes)
- Valida tu inscripción (alumnos que revalidaron)

Así mismo los alumnos que solicitaron revalidación de calificación de laboratorio, entran a este módulo para consultar el resultado de su petición, en este caso por ser la primera vez que se realiza, se busca la calificación del alumno en las bases de datos del laboratorio de semestres anteriores y se coloca la calificación en su registro para que la consulten en las fechas indicados, si existe alguna discrepancia en la calificación el alumno puede acudir al laboratorio a solicitar cualquier aclaración.



Este fue el proceso de pruebas que en general encontramos satisfactorio como desarrolladores, ya que la mayoría de las observaciones de los alumnos fueron acerca del aspecto visual del sistema, más no de la funcionalidad, nos sugirieron aumentar el tamaño en general de la letra y en algunos casos como en la pantalla donde se le muestra al alumno oyente su número de reinscripción temporal.

Un comentario mas hecho durante las pruebas, fue que se colocaran pequeños textos que dieran alguna instrucción respecto a la opción que el alumno estaba por seleccionar.

A continuación se presenta una tabla que indica cuales fueron los textos que se colocaron, así como, el módulo y la liga a la que están asociados.

**Tabla de letreros de ayuda**

Módulo	Liga	Texto de ayuda
2 y 4	Regulares	Si vas a revalidar no debes inscribirte como alumno regular, elige la opción "Revalidar". En caso de haber revalidado, no podrás inscribirte a un grupo de laboratorio
	Regulares que solicitaron la asignatura durante el periodo de altas y cambios	
	Oyentes y ASDRI	
2	Revalidar	Si ya te inscribiste en un grupo de laboratorio, no podrás revalidar calificaciones de semestres pasados.

Considerando las observaciones hechas en la etapa de pruebas, se procedió a su corrección y poner en marcha formalmente el sistema de inscripciones en el periodo 2013-2.

**Tabla resumen de las fallas y sus acciones correctivas**

Falla	Corrección
Las fechas de los módulos no se actualizan correctamente	Se implementó un control alternativo para cambiar el estado del proceso en cualquier momento

### 3.2.-Puesta en marcha

La puesta en marcha del nuevo sistema de inscripciones se realizó en el periodo 2013-2, el cual comenzó el día 28 de enero de 2013, ése mismo día comenzó la etapa de publicación de grupos y de registro al laboratorio.

En el apéndice de esta tesis se encuentra el manual de usuario para cada rol del proceso de inscripción, el de administrador y el del alumno. Después de las pruebas y corrección de fallas del formato del sistema, la puesta en marcha no presentó problemas de flujo con excepción de algunos alumnos que llegaron al laboratorio con reportes de fallas que estaban fuera de nuestro control, como el hecho de que la liga a la página no funcionaba pero esto se debía a que no tenían una buena conexión a internet y las paginas tardaban mucho tiempo en cargar o en otros casos no recordaban su número de reinscripción, en este último, para solucionar esta eventualidad se implementó colocar una liga a la página de la Facultad de Ingeniería, en la que semestralmente se publica dicho número para que los alumnos lo consulten vía WEB.

### 3.3.-Requerimientos

Para garantizar una operación constante y buen funcionamiento del sistema se requiere que la infraestructura en donde se ejecute sea segura. Esto quiere decir que no debemos limitarnos con brindar seguridad a la página web exclusivamente, ya que la seguridad comprende todos los componentes utilizados para el funcionamiento del sistema, desde la computadora o dispositivo de un usuario viajando por la red de datos hasta el servidor de manera bidireccional.

#### ***Recomendaciones para el usuario***

El software antivirus puede identificar y bloquear una gran cantidad de virus antes de que infecten un equipo de cómputo. Una vez que se instala un software antivirus, es importante mantenerlo actualizado.

Tener configurado firewall personal y bloquear todos los puertos que no se utilicen, regularmente los sistemas operativos cuentan con firewall por software y por defecto bloquean todos los puertos, se van desbloqueando con autorización del usuario cuando alguna aplicación así lo requiera.

No abrir archivos de extraña procedencia. Una gran diversidad de virus en el Internet se propaga a través del correo electrónico, adjuntando a estos un archivo infectado con lo cual se puedan seguir propagando.

Establecer contraseñas complejas, cambiarlas periódicamente y no almacenarlas en equipos públicos.

No revelar tu contraseña, ya que contiene información confidencial. La gran mayoría de las veces esta es la causa del robo de información y usurpación de identidad.

### **Software**

Por parte del usuario:

Los requerimientos por parte de software son mínimos ya que el sistema se ejecutará vía WEB en un servidor de la Facultad de Ingeniería, por parte del usuario, sólo se necesita contar con una computadora con acceso a Internet y algún explorador de internet (internet Explorer, Mozilla Firefox, Google Chrome) actualizado.

Por parte del servidor:

Apache.- servidor WEB HTTP de código abierto y multiplataforma.

PHP.- Lenguaje de programación que es interpretado.

MySQL .- Sistema manejador de bases de datos.

Con sus respectivas configuraciones.

### **Hardware**

En la actualidad existen tantos tipos de sistemas operativos, con diferentes requerimientos de hardware que en este caso en particular no se necesita un hardware específico en cuanto a procesador, memoria RAM y capacidad en disco duro para ejecutar esta aplicación siempre y cuando el equipo cuente con un sistema operativo capaz de ejecutar algún explorador de internet y claro está, que esté conectado a internet.

### 3.4.-Mantenimiento

El mantenimiento del sistema es realizado por el administrador del sistema quien es el encargado de monitorear todo el proceso para evitar cualquier tipo de falla.

## CONCLUSIONES Y RECOMENDACIONES



## Conclusiones y recomendaciones

### Conclusiones

Este capítulo está dedicado a exponer las conclusiones y recomendaciones a las cuales se llegaron durante el desarrollo de este trabajo de tesis, con el fin de abrir la posibilidad del seguimiento al proyecto.

Con la construcción de estos sistemas computacionales, se puede lograr una automatización óptima para los procesos en cuestión, así como, un aumento en su eficiencia, como en el caso del laboratorio de mecánica de la Facultad de Ingeniería, en el cual se logró automatizar el proceso de inscripciones, generando una nueva línea de trabajo para este proceso, en la que se ven involucrados únicamente el coordinador del laboratorio y el administrador del sistema, a diferencia de antes que se necesitaba la intervención de todos los que laboran en el laboratorio de mecánica durante todo el periodo de inscripciones. Es aquí donde la automatización juega un papel fundamental ya que con la implementación del sistema de inscripciones en línea, se configura y se monitorea su correcto funcionamiento en tiempo real y al finalizar el periodo de inscripciones automáticamente se dan a conocer los resultados y se generan las listas de asistencia para los profesores, así como, las estadísticas de los grupos más solicitados.

En lo que respecta al proceso de desarrollo del sistema, estamos de acuerdo en que la etapa de diseño es la más importante, ya que si se realiza de manera adecuada, se puede ahorrar mucho tiempo de desarrollo reduciendo las fallas a corregir. Es por esto que el modelo evolutivo nos permitió avanzar de manera clara y segura durante el desarrollo permitiéndonos planear cada módulo en base a los requerimientos del sistema, de manera que la implementación era directa y nos enfocábamos en probarlo para analizar su correcto funcionamiento.

Así mismo la elección de un lenguaje como PHP, nos permitió avanzar de manera independiente y así optimizar el tiempo de desarrollo, mediante el uso de clases en virtud de que es un lenguaje orientado a objetos que además

funciona de manera muy estable en la conexión con MySQL, nuestro manejador de bases de datos.

Este trabajo de tesis además de ser un reto como ingenieros, implica el apoyo directo a la Facultad de Ingeniería, para nosotros fue un verdadero placer desarrollar y poder retribuir el conocimiento que la Facultad nos brindó durante nuestra estancia en ella.

### Recomendaciones

Nuestra recomendación más importante está enfocada en que se tenga en cuenta que el óptimo funcionamiento de cualquier sistema a largo plazo está en su mantenimiento y actualización, ya que con el avance de la tecnología, los sistemas necesitan una constante revisión y optimización, de lo contrario, se puede tener un sistema obsoleto con el paso del tiempo si no se atiende cuidadosamente esta recomendación.

También es importante aclarar, que en un futuro se tenga en cuenta la posibilidad de integrar más asignaturas, ya que el sistema es apto para una expansión de este tipo, por el diseño de la base de datos, sin embargo no se llegó a dicha expansión ya que el objetivo de esta tesis es el de cumplir con los requerimientos del laboratorio de mecánica, los cuales quedan satisfechos en base a las asignaturas que están bajo su responsabilidad, Estática y Cinemática y dinámica.



## BIBLIOGRAFÍA Y REFERENCIAS



## Bibliografía

- **Andrew S. Tanenbaum**, Redes de computadoras, Cuarta Ed, 2003, Pearson.
- **Lee Babin**, Introducción a Ajax con PHP, Primera Ed, 2007, Anaya Multimedia-Anaya Interactiva.
- **Paul Deitel**, Como Programar en Java, Novena ED, 2012, Pearson.
- **Steve Suehring**, La biblia de PHP, Tim Converse y Joyce Park, Primera Ed, 2009, Anaya Multimedia-Anaya Interactiva

## Referencias

- <http://www.mastermagazine.info/termino/4294.php>, consultado el día 15 de marzo de 2012.
- <http://www.apache.org/>, consultado en día 27 de abril de 2012.
- [http://www.intergraph.com/global/mx/about\\_us/history\\_90s.aspx](http://www.intergraph.com/global/mx/about_us/history_90s.aspx), consultado el día 22 de febrero de 2012.
- <http://www.php.net/manual/es/>, consultado el día 18 de junio de 2012.
- <http://www.mysql.com/why-mysql/>, consultado el día 18 de junio de 2012.
- <http://manuales.dgsca.unam.mx/javascript/EvHisLen.html>, consultado el día 17 de abril de 2012.
- <http://www.um.es/docencia/barzana/DIVULGACION/INFORMATICA>, consultado el día 29 de abril de 2012.
- <http://www.java.com/es/download/faq/firewall.xml>, consultado el día 07 de agosto de 2012.
- <http://support.microsoft.com/kb/321050/es>, consultado el día 10 de septiembre de 2012.



## APÉNDICE A



## Apéndice A

### Conceptos básicos sobre bases de datos

Base de datos. Es una colección de datos relacionados.

RDBMS. Relational Database Management System o Sistema de gestión de base de datos relacional.

DDL. Data Definition Language o Lenguaje de definición de datos.

DML. Data Manipulation Language o Lenguaje de manipulación de datos.

DCL. Data Control Language o Lenguaje de control de datos. Controla acceso a la base de datos.

SQL es el lenguaje de las bases de datos relacionales.

### Roles y actores

- Administrador de datos y bases de datos (DBA)
- Diseñadores de bases de datos
- Desarrolladores de aplicaciones
- Usuarios finales

### Administrador de datos y bases de datos (DBA)

- Administración de datos
  - Planificación de bases de datos
  - Desarrollo de estándares, políticas y procedimientos
  - Diseño lógico de la base de datos
- Administración de bases de datos
  - Diseño e implementación de la base de datos
  - Acceso y control (usuarios y contraseñas)
  - Mantenimiento
  - Respaldo

### Diseñador de bases de datos

- Diseño lógico de la base de datos
- ¿Qué datos deben ser almacenados?
  - Definir vista parcial
- ¿Cómo deben organizarse los datos?
- ¿Cuáles son los usos típicos de los datos?

#### Programador de aplicaciones

- Programas que usen las bases de datos
- Interfaces a otros sistemas
  - Web
  - Correo electrónico
  - Sistemas de ficheros
  - Etc.

#### Usuarios finales

- Paramétricos o simples
  - Usuarios de las aplicaciones finales
  - Consultas pre-fijadas
- Expertos
  - Reportes y análisis

#### Ventajas de las bases de datos

- Control de la redundancia
- Evitar inconsistencia
- Restricción de accesos
- Controlar concurrencia
- Relaciones complejas entre datos
- Restricciones de integridad
- Los datos son independientes de los programas
- Copias de seguridad y recuperación

#### Desventajas de las bases de datos

- La seguridad y la integridad pueden ser contraproducentes sin buenos controles



## Definición de modelo de datos

Es el resultado del análisis de la información y consiste en la representación conceptual de esta.

## Tipos de modelos de datos

- Jerárquico. Se puede especificar relaciones de un registro padre y múltiples registros hijos.
- De red. Se puede especificar relaciones entre múltiples registros padre y múltiples registros hijos.
- Relacional. Se representa lógicamente de B.D. mediante entidades y sus relaciones entre sí.

## Modelo de datos Entidad-Relación

Se basa en la percepción del mundo real, que consiste en un conjunto de objetos llamados entidades y las relaciones entre ellas. Representa la estructura lógica general de la base de datos gráficamente.

Entidad. Es un objeto concreto como una persona o abstracto como un día festivo. Siempre son sustantivos tangibles o intangibles. Está formada por un conjunto de atributos (características propias).

P.e. ALUMNO

Para cada atributo existe un rango de valores permitidos llamado dominio del atributo.

P.e. El dominio del atributo Nombre pueden ser el conjunto de todos los alumnos inscritos a una asignatura.

## Definición de atributos

Describen una entidad para calificar, identificar, clasificar, cuantificar o expresar el estado de una entidad.

P.e. Algunos de los atributos de la entidad ALUMNO pueden ser:

- Número de cuenta que identifica a un ALUMNO
- Nombre y apellidos califican a un ALUMNO
- Correo electrónico que consulta un ALUMNO

Conjunto de entidades. Es un grupo de entidades del mismo tipo.

Índice. Es una estructura de datos definida sobre una columna de tabla (o varias) y que permite localizar de forma rápida las filas de la tabla en base a su contenido en la columna indexada además de permitir recuperar las filas de la tabla ordenadas por esa misma columna. Los índices en las bases de datos al igual que en los libros sirven para encontrar más rápido lo que buscamos, por tanto extrapolando en las bases de datos sirven para agilizar las consultas a las tablas.

Ventajas

- Se evita la revisión completa de una tabla.
- Se evita la sobrecarga de CPU, sobrecarga de disco y concurrencia.
- Se evitan las lecturas secuenciales.
- Permiten mayor rapidez en la ejecución de las consultas.

Desventajas

- Para tablas con operaciones de escritura (insert, delete y update) frecuente se actualizan los índices cada que se modifica una columna.
- En tablas demasiado pequeñas no es aconsejable puesto que no necesitamos ganar tiempo.
- No es aconsejable para tablas en la que se aplica devuelva una gran cantidad de datos en cada consulta.
- Los índices ocupan espacio y en algunas ocasiones más espacio que los propios datos.

Terminología relacional equivalente

Relación = tabla o archivo

Registro = registro, fila, renglón o tupla

Atributo = columna o campo

Clave = llave o código de identificación

Clave Candidata = superclave mínima

Clave Primaria = clave candidata elegida

Clave Ajena (o foránea) = clave externa o clave foránea

Clave Alternativa = clave secundaria

Dependencia Multivaluada = dependencia multivalor

1FN = Significa, *Primera Forma Normal* o 1NF del inglés *First Normal Form*.

## Fundamentos de la normalización

La normalización es el proceso de simplificar la estructura de datos de una base de datos de forma organizada para evitar redundancia e incoherencias.

Los datos redundantes afectan en el espacio de disco y consumo de datos y crean problemas de mantenimiento, por ejemplo si deseamos actualizar los datos de un alumno resulta muy rápido cambiarlos si sólo se almacenan en un catálogo de alumno.

Existen reglas en la normalización de una base de datos. Cada regla se denomina una "forma normal" y básicamente son tres. Si se cumple la primera regla, se dice que la base de datos está en la "*Primera Forma Normal (1FN)*". Si se cumplen las tres primeras reglas, la base de datos se considera que está en la "*Tercera Forma Normal (3FN)*". Aunque son posibles otros niveles de normalización, la Tercera Forma Normal se considera el máximo nivel necesario para la mayor parte de las aplicaciones.

Al igual que con otras muchas reglas y especificaciones formales, en los escenarios reales no siempre se cumplen los estándares de forma perfecta.

Regla	Descripción
<i>Primera Forma Normal (1FN)</i>	Incluye la eliminación de todas las columnas repetidas y colocarse en tablas separadas.
<i>Segunda Forma Normal (2FN)</i>	Asegura que todas las columnas que no son llave sean completamente dependientes de la llave primaria (PK)
<i>Tercera Forma Normal (3FN)</i>	Elimina cualquier dependencia transitiva*.

\*Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

Reconocer datos sin normalizar

Un dato sin normalizar no cumple con ninguna regla de normalización.

Por ejemplo.

idInscripcion	idAlumno	cuenta	nombre	email	idLaboratorio	grupo	asignatura	semestre
1	1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com	1	7	Estática	2011-2
2	1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com	2	13	Cinemática y Dinámica	2011-2
3	2	300124872	Mario Romero Islas	mromero@gmail.com	2	13	Cinemática y Dinámica	2011-2
4	3	300829133	José Luis Torres Infante	jtorres@hotmail.com	3	2	Cinemática y Dinámica	2011-2
5	4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com	4	12	Estática	2011-2
6	5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx	5	10	Cinemática y Dinámica	2011-2

Primera Forma Normal (1FN)

ALUMNOS

idAlumno	cuenta	nombre	email1	email2
1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com	ltapia@gmail.com
2	300124872	Mario Romero Islas	mromero@gmail.com	
3	300829133	José Luis Torres Infante	jtorres@hotmail.com	jtorres@gmail.com
4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com	rlorenzo@yahoo.com.mx
5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx	

Pasos para convertir a la 1FN:

- a. Eliminar los grupos de repetición.
- b. Duplicar los valores con registros repetidos o separar el atributo que viola 1FN en otra tabla.

ALUMNOS

idAlumno	cuenta	nombre	email
1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com
1	300276692	Lorena Tapia Arrellano	ltapia@gmail.com
2	300124872	Mario Romero Islas	mromero@gmail.com
3	300829133	José Luis Torres Infante	jtorres@hotmail.com

3	300829133	José Luis Torres Infante	jtorres@gmail.com
4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com
4	302939991	Ricardo Lorenzo Zepeda	rlorenzo@yahoo.com.mx
5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx

ó

## ALUMNOS

idAlumno	cuenta	nombre
1	300276692	Lorena Tapia Arrellano
2	300124872	Mario Romero Islas
3	300829133	José Luis Torres Infante
4	302939991	Ricardo Lorenzo Zepeda
5	300123988	Miguel Martell Álvarez

## EMAIL

idAlumno	email
1	lorena@hotmail.com
1	ltapia@gmail.com
2	mromero@gmail.com
3	jtorres@hotmail.com
3	jtorres@gmail.com
4	lorenzo@hotmail.com
4	rlorenzo@yahoo.com.mx
5	mmartell@yahoo.com.mx

## Segunda Forma Normal (2FN)

## ALUMNOS

idAlumno	cuenta	nombre	email
1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com
1	300276692	Lorena Tapia Arrellano	ltapia@gmail.com
2	300124872	Mario Romero Islas	mromero@gmail.com
3	300829133	José Luis Torres Infante	jtorres@hotmail.com
3	300829133	José Luis Torres Infante	jtorres@gmail.com
4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com
4	302939991	Ricardo Lorenzo Zepeda	rlorenzo@yahoo.com.mx
5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx

Pasos para convertir a la 2FN:

Determinar cuáles columnas que no son llave no dependen de la llave primaria completa de la tabla.

Eliminar esas columnas de la tabla base.

Crear una segunda tabla con esas columnas y la(s) columna(s) de la PK de la cual dependen.

Como podemos observar la llave primaria en la tabla ALUMNOS se conforma por las columnas idAlumno y email y las columnas no llave cuenta y nombre no dependen de la llave primaria completa, únicamente de id Alumno, por tanto creamos una segunda tabla como sigue:

#### ALUMNOS

idAlumno	cuenta	nombre
1	300276692	Lorena Tapia Arrellano
2	300124872	Mario Romero Islas
3	300829133	José Luis Torres Infante
4	302939991	Ricardo Lorenzo Zepeda
5	300123988	Miguel Martell Álvarez

#### EMAIL

idAlumno	email
1	lorena@hotmail.com
1	ltapia@gmail.com
2	mromero@gmail.com
3	jtorres@hotmail.com
3	jtorres@gmail.com
4	lorenzo@hotmail.com
4	rlorenzo@yahoo.com.mx
5	mmartell@yahoo.com.mx

## Tercera Forma Normal (3FN)

Pasos para convertir a la 3FN:

## INSCRIPCION

idInscripcion	idAlumno	cuenta	nombre	email	idLaboratorio	grupo	asignatura	semestre
1	1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com	1	7	Estática	2011-2
2	1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com	2	13	Cinemática y Dinámica	2011-2
3	2	300124872	Mario Romero Islas	mromero@gmail.com	2	13	Cinemática y Dinámica	2011-2
4	3	300829133	José Luis Torres Infante	jtorres@hotmail.com	3	2	Cinemática y Dinámica	2011-2
5	4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com	4	12	Estática	2011-2
6	5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx	5	10	Cinemática y Dinámica	2011-2

- Determinar que columnas son dependientes de otra columna no llave.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y con la columna no llave de la cual son dependientes.

## INSCRIPCION

idInscripcion	idAlumno	idLaboratorio
1	1	1
2	1	2
3	2	2
4	3	3
5	4	4
6	5	5

## ALUMNOS

idAlumno	cuenta	nombre	email
1	300276692	Lorena Tapia Arrellano	lorena@hotmail.com
2	300124872	Mario Romero Islas	mromero@gmail.com
3	300829133	José Luis Torres Infante	jtorres@hotmail.com
4	302939991	Ricardo Lorenzo Zepeda	lorenzo@hotmail.com
5	300123988	Miguel Martell Álvarez	mmartell@yahoo.com.mx

## GRUPOS\_LAB

<b>idLaboratorio</b>	<b>grupo</b>	<b>asignatura</b>	<b>semestre</b>
1	7	Estática	2011-2
2	13	Cinemática y Dinámica	2011-2
3	2	Cinemática y Dinámica	2011-2
4	12	Cinemática y Dinámica	2011-2
5	10	Estática	2011-2

## ASIGNATURAS

<b>idAsignatura</b>	<b>asignatura</b>
1	Estática
2	Cinemática y Dinámica