



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE CIENCIAS**

**“Aplicación de TSPi para un sitio WEB en CONACULTA”**

**PROYECTO**

**QUE PARA OBTENER EL TÍTULO DE  
LICENCIADO EN CIENCIAS DE LA COMPUTACIÓN**

**PRESENTA:**

**LEIF OSWALDO ALDUNCIN GIL**

**DIRECTORA DEL PROYECTO**

**M. EN C. MARÍA GUADALUPE ELENA IBARGUENGOITIA GONZÁLEZ**



**MÉXICO D.F.**



**FACULTAD DE CIENCIAS  
SECCION ESCOLAR**

**AGOSTO 2004**

**TESIS CON  
FALLA DE ORIGEN**



Universidad Nacional  
Autónoma de México



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

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

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

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



UNIVERSIDAD NACIONAL  
AVENIDA 14  
MEXICO

**ACT. MAURICIO AGUILAR GONZÁLEZ**  
**Jefe de la División de Estudios Profesionales de la**  
**Facultad de Ciencias**  
**Presente**

Comunicamos a usted que hemos revisado el trabajo escrito:

"Aplicación de TSPi para un sitio WEB en CONACULTA"

realizado por Alduncin Gil Leif Oswaldo

con número de cuenta 09407683-3 , quien cubrió los créditos de la carrera de:

Ciencias de la Computación

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

Director de Tesis

Propietario M. en C. María Guadalupe Elena ~~Iparguengoitia González~~ *Iparguengoitia*

Propietario Dra. Hana Oktaba *H Oktaba*

Propietario M. en I. O. María de Luz Gasca Soto *Lu Gasca*

Suplente M. en C. Gustavo Arellano Sandoval *Gustavo Arellano Sandoval*

Suplente Dra. Amparo López Gaona *Amparo*



Consejo Departamental de Matemáticas

*Francisco Hernández Quiroz*  
Dr. Francisco Hernández Quiroz

CONSEJO DEPARTAMENTAL

MATEMÁTICAS

## **DEDICATORIA**

Dedico este trabajo a mi madre María de Lourdes Gil Chávez y a Carolina Campuzano García, porque son las mujeres más importantes en mi vida y las amo con todo mi corazón.

A Ulises Ortiz Reyes y Alberto Rojas Blanco por ser los hermanos y amigos que escogí y estar siempre a mi lado en las buenas y en las malas con palabras de aliento.

A mi familia y a la memoria de mi padre Eugenio Enrique Alduncin Riquelme, lo prometido es deuda papá.

A Dios que me ha permitido llegar a este momento bajo su protección y consuelo.

¡Ah ¡ y también a Carolina.

Porque el dolor es la piedra angular del aprendizaje.

Anónimo.

## AGRADECIMIENTOS

A Dios por darme vida y salud para la culminación de este trabajo.

A mis hermanos Percival y Erwin por enseñarme muchas lecciones de la vida.

A la Universidad Nacional Autónoma de México por ser mi segunda casa y ser un lugar lleno de oportunidades.

A mi directora María Guadalupe Elena Ibargüengoitia por ser una mujer líder y haberme apoyado en todo el proceso de mi trabajo siempre con una sonrisa.

A mis sinodales por sus valiosas atribuciones a mi proyecto.

A Luis Antonio Hernández Nieto por todo el apoyo incondicional y por su ejemplo de fortaleza ante los malos momentos y ser un padre de familia excelente.

A mi madrina Lupita Puertas por ser una mujer tan llena de amor en su corazón.

Al grupo de amigos de la generación quienes me han apoyado durante estos años.

Al Ingeniero Felipe Flores por todo su apoyo cuando más lo necesitaba y todo lo que le he aprendido.

A las familias de mis amigos por abrirme sus puertas y ofrecerme todo el apoyo.

Y todos aquellos que ya no mencione pero que se que están en mi corazón.

Gracias de verdad muchas gracias.

Leif Oswaldo Alduncin Gil.

## Índice

<b>Introducción.....</b>	<b>1</b>
--------------------------	----------

### PARTE I

<b>Capítulo 1. Panorama sobre el Proceso de Software en Equipo.....</b>	<b>2</b>
<b>Capítulo 2. Etapa de Lanzamiento.....</b>	<b>5</b>
<b>Capítulo 3. Fase de Estrategia.....</b>	<b>10</b>
<b>Capítulo 4. Fase de Planeación.....</b>	<b>13</b>
<b>Capítulo 5. Fase de Requerimientos.....</b>	<b>15</b>
<b>Capítulo 6. Fase de Diseño.....</b>	<b>19</b>
<b>Capítulo 7. Fase de Implementación.....</b>	<b>23</b>
<b>Capítulo 8. Pruebas de Integración y del Sistema.....</b>	<b>26</b>
<b>Capítulo 9. Postmortem.....</b>	<b>30</b>

### PARTE II

<b>Introducción al desarrollo.....</b>	<b>33</b>
<b>Capítulo 10. Fase de Lanzamiento.....</b>	<b>37</b>
<b>Capítulo 11. Fase de Estrategia.....</b>	<b>44</b>
<b>Capítulo 12. Fase de Planeación.....</b>	<b>53</b>
<b>Capítulo 13. Fase de Requerimientos.....</b>	<b>63</b>
<b>Capítulo 14. Fase de Diseño.....</b>	<b>83</b>
<b>Capítulo 15. Fase de Implementación.....</b>	<b>104</b>
<b>Capítulo 16. Plan de Pruebas de Integración y del Sistema.....</b>	<b>131</b>
<b>Capítulo 17. Postmortem.....</b>	<b>141</b>
<b>Conclusiones.....</b>	<b>150</b>
<b>Bibliografía.....</b>	<b>152</b>

### PARTE III

<b>Anexo.....</b>	<b>153</b>
-------------------	------------

## Introducción

El desarrollo de software requiere de trabajo en equipo: fijando metas, objetivos comunes, con una comunicación fluida y todo esto se refleja en la calidad del producto. Un equipo de trabajo es más que un grupo de personas trabajando juntas que tienen ciertas actividades correlacionadas para la solución de un problema, es un grupo de personas que se relacionan con fines de desarrollar software que funcione y que sea útil.

Entonces ¿Porqué los proyectos de desarrollo en equipo de software fallan? Uno de los principales problemas de los equipos de software es aprender cómo dirigir el trabajo conjunto; una mala forma de responder a él hace que los proyectos fallen. Regularmente cuando un proyecto falla es porque los equipos responden con métodos pobres, manejados por la presión. La excesiva presión puede ser destructiva, pues causa que la gente se preocupe e imagine problemas y dificultades que pueden no ser reales.

Los equipos necesitan conocer como trabajar eficientemente para producir productos de calidad sobre todo cuando están bajo un intenso periodo de presión. Trabajar en equipo tiene muchas ventajas pero también puede ser un problema sin un buen líder, ausencia de participación, retrasos, falta de compromisos y avances lentos.

Cuando se trabaja en equipo donde no hay metas, tareas, planes, objetivos ni misiones definidas y claras, cada uno de los miembros del equipo imagina que debe hacer y el proyecto falla.

Para responder a estos problemas el libro "Team Software Process Introduction" proporciona una guía para la solución de problemas en equipo, dando niveles de calidad para el desarrollo de software los cuales son explicados a detalle más adelante.

Este trabajo consiste de dos partes, en la primera se encuentra el marco teórico que está basado directamente en TSPi. La segunda parte, es una aplicación de la misma, que se desarrolló en las instalaciones de CONACULTA (Consejo Nacional para la Cultura y las Artes).

En los primeros capítulos del 1 al 9 se hace un resumen de TSPi . En los capítulos 10 al 16 se describe con detalle cómo se aplicó TSPi al proyecto que consiste en administrar todos los cursos culturales de CONACULTA (carteleros y sesiones) que se generan a nivel República Mexicana, en él se administran a las personas que los toman (promotores culturales), los coordinadores de cada curso y los datos de quienes imparten los cursos (Instructores).

# Capítulo 1. Panorama sobre el Proceso de Software en Equipo.

## 1.1. Definición de TSPi (Team Software Process introduction).

Team Software Process introduction (TSPi) fue creado por Watts S. Humphrey en el SEI (Software Engineering Institute – Instituto de Ingeniería de Software)<sup>1</sup> para guiar a los equipos de desarrollo de software de alta calidad.

Es una estructura de trabajo para la construcción y dirección de equipos de ingeniería de software, permite un balance entre el proceso, productos, y trabajo en equipo. (Watts S. Humphrey. 2000).

El principal beneficio es mostrar a los equipos de ingeniería de software cómo administrar su trabajo e ir construyendo sus propios planes y procesos.

TSPi tiene cinco objetivos:

1. Construir equipos de software auto-dirigidos que planifican y siguen su trabajo, establecen metas, procesos y planes propios.
2. Indicar a los niveles gerenciales como dirigir, motivar y ayudar a sus equipos para que mantengan un alto desempeño.
3. Facilitar a la enseñanza universitaria con habilidades de trabajo en equipo a nivel industrial.
4. Dar una guía de mejora para alcanzar madurez organizacional.
5. Acelerar el proceso de mejora de software, facilitando a la organización para alcanzar el nivel 5 de CMM.<sup>2</sup>

## 1.2. Definición de un equipo.

Un equipo consiste en:

- Mínimo de dos personas.
- Estar trabajando hacia metas, objetivos y misiones comunes.
- Cada persona ha sido asignada con roles específicos o funciones a desarrollar.
- Terminar todas las misiones requeridas dependiendo del número de personas del equipo.

Los equipos pueden ser desde dos integrantes hasta una docena de ellos, sin embargo en la práctica se ha demostrado que los equipos resultan ser eficientes cuando son pequeños.

Un equipo es un grupo de personas fuertemente unidas, en que se hace la suma de las partes, la producción de la gente en equipo es mayor que trabajando de forma individual.

---

<sup>1</sup> El Software Engineering Institute (SEI) es un centro de investigación y desarrollo fundado con fondos federales por el Departamento de Defensa de EEUU, y operado por la Universidad Carnegie Mellon.

<sup>2</sup> Capability Maturity Model (Modelo de Capacidad de Madurez) es un modelo desarrollado por el SEI para determinar la madurez del proceso del software en una organización.

No todos los grupos son equipos, aquí hay tres condiciones básicas para que un grupo funcione adecuada y exitosamente como equipo:

1. Las tareas son claras y distintas; así que el trabajo en un equipo esta explícitamente definido.
2. El equipo es claramente identificado; los miembros saben el alcance del grupo, quien está y quien no lo está. Todos se conocen entre sí, el trabajo de cada uno es visible y se conoce el rol de cada persona del equipo.
3. El equipo tiene el control sobre las tareas; los miembros del equipo saber qué hacer, cómo hacerlo, cuándo hacerlo y cuándo está terminando. Los miembros saben quienes son los responsables del trabajo y ellos tienen el control del proceso en uso.

Las metas son un elemento crítico de la formación de equipos. Primero éstas metas deben de ser específicas y alcanzables. Cada miembro del equipo debe aceptar cada una de esas metas para sí mismo.

### **1.3. Estructura común de trabajo.**

Las metas del equipo deben ser eficientes y claras, el camino a seguir debe de estar bien especificado. Los miembros del equipo deben saber como llegar a las metas y conocer que expectativas tienen. Deben entender sus roles, responsabilidades y deben estar de acuerdo en ellas. Cada persona sabe:

- ¿Qué tareas deben de hacer?
- ¿Cuándo?
- ¿En qué orden?

TSPi guía a los equipos pensando la construcción de pasos y un conjunto de metas, selección de roles, establecimiento de planes y manteniendo la comunicación entre los miembros del equipo. Las fases para el desarrollo de proyectos en TSPi y sus objetivos son las siguientes:

#### **Fase de lanzamiento.**

- Formar y construir el equipo.
- Determinar los roles de los miembros del equipo.
- Establecer las metas del equipo.

#### **Fase de estrategia.**

- Crear una estrategia para realizar el producto.
- Crear un diseño conceptual del producto.
- Hacer estimaciones preliminares sobre el tamaño del producto y tiempo de desarrollo.
- Documentar la estrategia.

#### **Fase de planeación.**

- Realizar el plan de desarrollo del proyecto.
- Desarrollar el plan de calidad.
- Aplicar el criterio de estándares de calidad para generar un producto de alta calidad.

**Fase de requerimientos.**

- Describir el proceso de requerimientos en TSPi.
- Explicar qué son y por qué son necesarios.
- Discutir las características más importantes.

**Fase de diseño.**

- Proporcionar un diseño completo y de alta calidad que se utilice como base para la fase de implementación.

**Fase de implementación.**

- Describir el proceso a través del criterio de diseño.
- Describir el proceso a través de los estándares y estrategias de implementación.
- Realizar revisiones e inspecciones.

**Pruebas de integración y del sistema.**

- Integrar las pruebas y la documentación de las mismas.
- Indicar los objetivos de las pruebas.
- Indicar las estrategias de las pruebas.
- Indicar la planeación de las pruebas.

**Postmortem.**

- Describir las necesidades del postmortem, su proceso y utilidad.
- Revisar el trabajo concluido por los miembros del equipo al final del ciclo.
- Revisar todos los datos requeridos.
- Proporcionar de una manera estructurada el aprendizaje y mejora tanto personal como en equipo.

TSPi define como paso inicial del equipo acordar metas de los miembros del equipo, seguido de una secuencia de ciclos, el equipo revisará y ajustará estas metas basadas en su experiencia con el desarrollo de un primer ciclo.

Después de que el equipo está de acuerdo sobre las metas y los roles, el siguiente acuerdo es llevar a cabo estas metas. El equipo entonces define los contenidos funcionales de cada ciclo, el tamaño esperado, los caminos para integrar y las pruebas a producir al producto terminado. Los miembros del equipo deciden sobre el proceso de la documentación que usarán para hacer el trabajo. Con los procesos definidos, el equipo entonces estima el tamaño del producto en cada ciclo, el tiempo de producir cada producto. Cuando han terminado, tienen el plan de desarrollo.

Con los roles definidos, procesos, planes y medidas estimadas, los miembros del equipo pueden comunicarse clara y concisamente.

TSPi guiará al líder del equipo para llevar a cabo reuniones semanales, basándose en los reportes de evaluación para los status. Esos reportes muestran como se ha acoplado el equipo y que planes deben de hacer para la siguiente semana.

## Capítulo 2. Etapa de Lanzamiento.

El objetivo es formar y construir el equipo, determinar los roles del equipo, establecer sus objetivos, ayudar a decidir sobre la práctica de éstos y cómo establecer los objetivos del equipo.

Los objetivos deben ser medibles, dado que establecen la estructura para la estrategia y el plan.

### 2.1. Objetivos del proyecto:

1. Definir los objetivos del proyecto.
2. Especificar cómo medirlos.
3. Proporcionar una copia de los objetivos a los integrantes del equipo y al instructor.
4. Proporcionar una copia de los objetivos al administrador de configuración para que los anexe en la carpeta del proyecto.

Para establecer los objetivos del proyecto en primer lugar se debe considerar que el producto cumpla con las expectativas del cliente. Por ello TSPi considera los siguientes objetivos básicos:

- Objetivo 1: Generar un producto de calidad.
- Objetivo 2: Realizar un proyecto bien administrado y productivo.
- Objetivo 3: Finalizar a tiempo.

Objetivo 1: Generar un producto de calidad. TSPi propone los siguientes objetivos para desarrollar un producto de calidad.

- Porcentaje de defectos encontrados en la primera compilación: 80%.
- Número de defectos encontrados en las pruebas del sistema: 0.
- Requerimientos de funciones incluidas en el proyecto: 100%.

Objetivo 2: Realizar un proyecto bien administrado y que sea productivo.

- Error en el tamaño del producto estimado menor al 20%.
- Error en las horas desarrollo estimadas menor al 20% .
- Porcentaje de datos registrados e incluidos en la carpeta del proyecto: 100%.

Objetivo 3: Finalizar a tiempo.

- Días antes o después en completar el ciclo de desarrollo a lo más 4.

Objetivo común.

El objetivo común consiste en ser un integrante de equipo eficaz y cooperativo. Todos tienen el objetivo principal de trabajar en cooperación con el equipo entero para generar un producto de alta calidad de acuerdo al calendario establecido para el proyecto. El éxito del equipo depende de que todos sus integrantes contribuyan con su mejor esfuerzo personal, apoyando a los demás y trabajando conjuntamente para resolver problemas y desacuerdos.

## 2.2. Objetivos de los miembros del equipo.

Se espera que cada integrante del equipo conozca los objetivos del proyecto y que además cumpla los siguientes objetivos como integrante y sepa cómo medirlos.

Objetivo 1. Ser un miembro del equipo efectivo y cooperativo.

Objetivo 2. Hacer un trabajo disciplinado.

- Porcentaje de datos personales registrados en la carpeta del proyecto: 100%.
- Porcentaje de las semanas en las cuales se realizaron y finalizaron las actividades registradas en cada forma personal SEMANA<sup>3</sup>: 100%.

Objetivo 3. Planear y mantener un seguimiento de trabajo personal.

- Porcentaje de datos del proyecto registrados en los formatos REGT: 100%.

Objetivo 4. Generar productos de calidad.

- Promedio de porcentaje de defectos encontrados antes de la primera compilación mayor al 70%.
- Densidad de defectos encontrados durante la compilación: 10/KLOC.
- Densidad de defectos encontrados durante las pruebas unitarias: 5/KLOC.
- Densidad de defectos encontrados después de la prueba unitaria: 0.

## 2.3. Objetivos del rol

Líder del equipo.

El líder del equipo es el responsable de que el equipo funcione adecuadamente.

Los objetivos específicos del líder del equipo son:

1. Construir y mantener un equipo eficaz.
2. Motivar a todos los miembros del equipo a trabajar tenazmente en el proyecto.
3. Resolver la mayoría de los problemas del equipo que se susciten.
4. Tener a todos los miembros del equipo y al instructor informados sobre los avances.
5. Actuar efectivamente como moderador en las reuniones.

Administrador de desarrollo.

El administrador de desarrollo está principalmente involucrado con la producción y funcionalidad de un producto de alta calidad de desarrollo.

Los objetivos principales del administrador de desarrollo son las siguientes:

1. Generar un producto de alta calidad.
2. Identificar plenamente las destrezas y habilidades de cada miembro del equipo.

Administrador de planeación.

Los objetivos principales del administrador de planeación son guiar al equipo en la producción de un plan detallado y guiar el progreso del producto de acuerdo al plan.

---

<sup>3</sup> Se debe dar seguimiento del tiempo, tamaño y los datos de defecto por trabajo personal registrándolos en la forma REGD. En la forma SEMANA se debe reportar el progreso del trabajo realizado contra lo planeado en la reunión semanal.

Los objetivos específicos del administrador de planeación son:

1. Producir un plan completo, preciso y detallado para cada miembro del equipo y para todo el equipo.
2. Proporcionar los reportes del estado semanal del equipo.

Administrador de calidad y proceso.

El objetivo principal del administrador de calidad y proceso es asegurar que el equipo esté siguiendo TSPi.

Los objetivos específicos del administrador de calidad y proceso son:

1. Asegurar que todos los miembros del equipo reporten y usen apropiadamente los datos del proceso de TSPi.
2. Asegurar que el equipo siga fielmente TSPi y producir un producto de calidad.
3. Asegurar que todas las metas del equipo sean bien realizadas y reportadas.
4. Asegurar que todas las metas del equipo sean reportadas con exactitud y que los reportes sean colocados en la carpeta del proyecto.

Administrador de configuración.

El principal objetivo del administrador de configuración es garantizar que el proyecto está propiamente soportado y controlado.

Los objetivos específicos del administrador de configuración son:

1. Asegurar que el equipo tiene disponibles las herramientas y métodos para que apoyen su trabajo.
2. No realizar cambios sin autorización en la línea base.
3. Todos los riegos y asuntos del equipo son registrados en el sistema de seguimiento.
4. Asegurar que el equipo conoce los objetivos de reuso para el ciclo de desarrollo.

## 2.4. Roles y sus responsabilidades.

Responsabilidad	LE	AD	AP	ACP	AC
Construye y mantiene un equipo efectivo.	X				
Resuelve asuntos entre los integrantes del equipo	X				
Realiza seguimiento y reporte del progreso del equipo	X				
Dirige cada reunión	X				
Se reúne con el instructor	X				
Realiza el mantenimiento a la carpeta del proyecto	X				
Ayuda a asignar las tareas en el equipo	X				
Dirige todo el trabajo de desarrollo.		X			
Dirige la planeación del equipo y el seguimiento del progreso.			X		
Dirige el seguimiento y planeación de calidad				X	
Proporciona apoyo al proceso del equipo.				X	
Dirige cada inspección.				X	
Mantiene los estándares y glosarios.				X	
Realiza el reporte de cada reunión.				X	
Alerta al equipo en los problemas de calidad.				X	
Obtiene las herramientas y apoyo necesario.					X
Maneja la administración de configuración.					X
Dirige la mesa de control de configuración.					X
Dirige el reuso del equipo.					X

Realiza el seguimiento de asuntos y riesgos.					X
Mantiene el glosario del sistema					X
Desarrolla el producto.	X	X	X	X	X
Realiza planes personales	X	X	X	X	X
Realiza seguimiento en el trabajo personal	X	X	X	X	X
Genera productos de calidad	X	X	X	X	X
Sigue prácticas profesionales disciplinadas.	X	X	X	X	X

Donde:

- Líder del equipo (LE).
- Administrador de Desarrollo (AD).
- Administrados de Planeación (AP).
- Administrador de Calidad y Proceso (ACP).
- Administrador de Configuración (AC).

### 2.5. Objetivos del Producto.

En esta parte se describe la construcción del producto. Si existe algún tipo de antecedente, se explican las características en las que se encuentra y dónde va a iniciar el producto.

### 2.6. Estándar para entrega de documentación.

El estándar de documentación permite al equipo mantener una misma línea en la características de la información, de manera tal que no existan copias de una misma versión ni confusiones entre los documentos en cada fase del proyecto.

El estándar de documentación contiene lo siguiente:

- Uso de procesador de textos y sus versiones.
- Tipo de letra y tamaño, ya sea tanto para títulos principales como para subtítulos.
- Tabuladores, margen, tamaño de pie de página.
- Tipo de alineación.
- Color de tinta y tamaño de la hoja para la impresión.
- Espacio de interlineado y tipo de viñeta.
- Formato de fecha.
- Características del contenido en la cabecera de cada documento.
- Identificación de cada documento en el pie del documento.

### Carpeta del Proyecto.

TSPi propone a los equipos mantener una carpeta para el proyecto. El objetivo es crear un estándar para salvaguardar todos los puntos importantes de información del proyecto. El líder del equipo es responsable para establecer y mantener la libreta del proyecto, y todos los miembros del equipo son responsables para proveer los materiales necesarios para la carpeta.

### 2.7. Reuniones del equipo y la primera reunión.

Después de que el equipo se ha establecido y los roles han sido asignados, es tiempo de hacer la primera reunión. Estas reuniones son hechas para dar al equipo la oportunidad de discutir acuerdos y las metas a desarrollar en el ciclo. Esta primera reunión también ayuda a entender el propósito de las reuniones semanales.

El equipo debe establecer un tiempo estándar, un lugar para las reuniones y reunir al equipo entero cada semana.

Otra función de las reuniones semanales es analizar los datos del equipo de la semana previa. Revisando los datos individuales, los miembros del equipo pueden ver en que parte se encuentran trabajando de acuerdo al plan, dónde necesitan balancear y ajustar sus planes personales.

## Capítulo 3. Fase de Estrategia.

### 3.1. La Estrategia de TSPi.

El objetivo es crear una estrategia para realizar el trabajo. Crear una gráfica de dependencias del producto. Hacer estimaciones preliminares sobre el tamaño del producto y el tiempo de desarrollo. Administrar los riesgos y el reuso. Finalmente, documentar la estrategia.

La estrategia del desarrollo se produce antes de iniciar el proyecto porque las actividades de estrategia y planeación están completamente relacionadas. El desarrollar un proceso comienza con un primer ciclo en el cual se diseña, implementa y se prueba el producto con una versión base produciendo un producto de calidad, con las funciones requeridas en el tiempo estimado.

### 3.2. Seleccionar la estrategia de desarrollo.

El siguiente paso es producir una propuesta de estrategia a seguir en el desarrollo. Uno de los caminos para hacer esto, es examinar las funciones deseadas para el producto total y entonces seleccionar la mínima carga de trabajo por subconjuntos. El objetivo es tener diferentes elementos funcionales bajo el desarrollo de cada ciclo. Esto da como resultado que cada ingeniero pueda contribuir al diseño e implementación del producto. En el diseño de la estrategia si se intenta incluir muchas funciones en el primer ciclo, el trabajo podría ser mucho más grande de lo planeado y no habría suficiente tiempo para completar subsecuentes desarrollos. Aunque el equipo puede seleccionar cualquier método para la estrategia de desarrollo se plantea una estrategia inicial y una gráfica de dependencias de necesidades.

### 3.3. Gráfica de Dependencias entre Necesidades [Hanna Oktaba].

Una Gráfica de Dependencias entre Necesidades es una gráfica dirigida con dos tipos de vértices, que representan las relaciones de dependencia total o parcial entre las necesidades funcionales de un problema dado. Se construye a partir de los siguientes pasos:

1. Se identifican las necesidades independientes, colocándose como nodos terminales de la gráfica. Una necesidad  $a$  se considera independiente si su satisfacción no depende de la satisfacción de ninguna otra necesidad.
2. Se analizan una por una el resto de las necesidades, revisando si su satisfacción tiene dependencia total de la satisfacción de otra necesidad previamente colocada en la gráfica. Una necesidad  $b$  depende totalmente de otra necesidad  $a$  si la satisfacción de  $b$  requiere previamente de la satisfacción completa de la necesidad  $a$ . Si este es el caso, la necesidad  $b$  se coloca en la gráfica como un nuevo nodo, unido por un vértice dirigido hacia el nodo de la necesidad  $a$  de la cual depende totalmente.
3. Se repite el paso anterior hasta colocar como nodos todas las necesidades.
4. La gráfica resultante se analiza, a fin de identificar los nodos cuyas necesidades se satisfagan parcialmente a partir de la satisfacción de otros nodos. Una necesidad  $b$  depende parcialmente de otra necesidad  $a$ , si  $b$  puede satisfacerse de modo incompleto mediante la satisfacción completa de la necesidad  $a$ . En estos casos, la

necesidad *b* se conecta mediante un vértice dirigido (trazado como una línea discontinua) hacia el nodo de la necesidad *a*, de la cual depende parcialmente.<sup>4</sup>

Nota: El análisis se realiza a partir de una descripción en lenguaje natural, por lo que puede ser susceptible a distintas interpretaciones. En consecuencia, es de esperar que este análisis pueda generar más de una gráfica representativa del problema.

La Gráfica de Dependencias de Necesidades ayuda a seleccionar estrategia que cumpla con los Criterios de Estrategia.

1. El producto del primer ciclo debe proporcionar un subconjunto ejecutable de funcionalidades mínimas del producto final. La gráfica permite revisar y seleccionar las funcionalidades mínimas del primer criterio de la fase de Estrategia, mediante la selección de los nodos terminales (no necesariamente todos) y una o más trayectorias, de poca longitud, que se dirijan a esos nodos. Los nodos incluidos dentro de las trayectorias representan necesidades a satisfacer en el primer ciclo.
2. El producto del primer ciclo debe proporcionar una base fácilmente extensible. A partir de los nodos seleccionados como parte de las funcionalidades mínimas, es posible ir añadiendo otros nodos dependientes durante los siguientes ciclos. De esta forma, el esfuerzo de desarrollo (costo y trabajo) por cada ciclo se aprovecha e incrementa de manera más eficiente.

El diseño de producto debe tener una estructura modular que permite trabajo independiente de los miembros del equipo. La gráfica sugiere una probable división de trabajo. Observando aquellos nodos que tienen más de un vértice de entrada (necesidades cuya satisfacción habilita la satisfacción de dos o más necesidades), sus nodos descendientes representan potencialidad de desarrollo independiente dentro de la estructura del producto, lo que los hace candidatos a una primera aproximación de componentes del diseño conceptual.

### **3.4. Administración de Riesgos.**

Un *riesgo* se detecta como un suceso con probabilidades de ocurrir y que afecta negativamente el proyecto. Para evitarlos o controlarlos, es conveniente identificarlos desde el principio y determinar las precauciones adecuadas para ello, el equipo deberá evaluar los riesgos que sean más significantes del proyecto, tomando una decisión cual de éstos son los más probables. De esta manera se clasifican tomando la escala de B(Baja), M(Mediana), A(Alta), dependiendo el impacto sobre el proyecto y clasificarlos.

### **3.5. Una Estrategia de Reuso.**

Desarrollar una gran cantidad de código genera riesgos ya que las funciones podrían requerir sustancialmente mas código. Uno de los caminos a reducir los riesgos es adoptar una estrategia de reuso. Se puede desarrollar código que se sabe que podría utilizarse más adelante. Esta estrategia puede reducir el esfuerzo, antes de desarrollar el producto se puede especificar y diseñar las partes que se pueden reusar, crear una biblioteca de reuso e incorporarla cuando el producto se esté desarrollando. En TSPi el administrador de

---

<sup>4</sup> Véase ejemplo práctico en la página 47

configuración es el encargado de proporcionar las partes de reciclaje que se van a guardar en la biblioteca.

### **3.6. Producir un plan de Configuración.**

Cuando el equipo no usa un plan adecuado de configuración, no conocen qué módulos han sido aprobados, aumentados, o probados. Se construyen productos con errores o no se conocen las pruebas que han sido corridas o cuáles defectos han sido inyectados y por lo tanto esto es tiempo perdido, por lo que se necesitará rehacer el trabajo que en realidad ya se había completado. Este tipo de problemas son realmente inquietantes porque cuando pasa, usualmente se está a punto de terminar el modulo o el ciclo, y se tiene muy poco tiempo para recobrarlo.

El proceso de administrar la configuración tiene los siguientes pasos [Humphrey, 1997, páginas 61 y 62]:

1. Copiar la versión de cada elemento del producto.
2. Registrar todos los cambios hechos por cada línea base.
3. Registrar quién hace los cambios.
4. Registrar cuando se hicieron los cambios.
5. Registrar qué cambios fueron hechos.
6. Registrar porqué se hicieron los cambios.

Por lo tanto, los miembros del equipo buscan la versión final de cualquier elemento del programa. También se conocerán los cambios hechos, cuándo y por qué. El administrador de configuración produce el plan de configuración y lo revisa con el resto del equipo.

## Capítulo 4. Fase de Planeación.

El objetivo es realizar el plan de desarrollo del proyecto y el plan de calidad. Aplicar el criterio de estándares de calidad para generar un producto de alta calidad.

Un plan está balanceado cuando todos los ingenieros completan sus tareas planeadas en el tiempo acordado. Ayuda a dirigir y administrar el trabajo, además se analizan los requerimientos, se revisan e inspeccionan los productos, y se documenta la información. También se puede desarrollar un plan de prueba, ejecutar las pruebas y reportar cada trabajo.

### 4.1. Diagrama de Gantt.

El diagrama de Gantt muestra el tiempo y el trabajo que debe ejecutarse en ese tiempo, la ocurrencia de actividades en paralelo o en serie de un determinado periodo. Las líneas que se trazan horizontalmente a través de ese espacio y muestran la relación entre el volumen realmente ejecutado de trabajo en ese tiempo y el volumen programado.

Para la elaboración del diagrama de Gantt deben seguirse los siguientes pasos:

- a) Se elabora una lista de actividades que intervienen en el proyecto, las cuales se relacionan y ordenan de acuerdo con su ejecución. En el caso de TSPi se obtiene de la forma ESTRA.
- b) Se establece en forma horizontal una escala de tiempos representada en años, meses, semanas, días, horas, según sean las necesidades o la duración del ciclo.
- c) Se estima la duración de cada actividad y la fecha planeada de inicio.
- d) Se representa la duración estimada de cada actividad con una barra horizontal, cuya longitud obedecerá a la duración establecida de acuerdo con la escala horizontal.
- e) El control se realiza por la simple comparación de las barras en una fecha determinada.
- f) Se puede hacer uso de herramientas como Microsoft Project para Windows o mr Project para GENOME.

### 4.2. La forma SEMANA.

Para generar la forma SEMANA:

1. Obtenga el total de horas estimadas para todos los integrantes del equipo en cada ciclo de vida del proyecto.
2. Compare el tiempo total del primer paso con el tiempo total requerido para cumplir con todas las tareas de cada ciclo.
3. Si el tiempo en cada tarea excede el tiempo total planeado deberá continuar con el siguiente paso.
4. Si la carga de trabajo es mucha, deberá decidir entre trabajar más horas o reducir el trabajo planeado; lo cual se deberá modificar en el diagrama de Gantt y verificar si está de acuerdo con el tiempo disponible.
5. Si se decide reducir la cantidad de trabajo, en la fase de estrategia revise las funciones que se implementarán en cada ciclo. Haga otra vez el plan.

#### **4.3. Producto Final.**

Para la conclusión de la planeación se deberá entregar los siguientes productos:

- Diagrama de Gantt del equipo y por cada integrante.
- Forma semana personal
- Forma semana por equipo

#### **4.4. Plan de Inspecciones.**

Para llevar a cabo un desarrollo con calidad se debe hacer una planeación que permita obtener tiempo suficiente para revisiones e inspecciones. Se recomienda utilizar las siguientes líneas guía:

- Revisiones e inspecciones en requerimientos a lo más 2.0 páginas por hora cuyo texto tendrá interlineado sencillo.
- Revisiones e inspecciones en diseño de alto nivel a lo más 5 páginas por hora de diseño.
- Revisiones e inspecciones en diseño detallado a lo más 100 líneas por hora en pseudocódigo.
- Revisiones e inspecciones de código fuente a lo más 200 LOC/hora.

Al momento de hacer revisiones no se asegura que se encontrarán defectos, por eso es necesario utilizar métodos de revisión efectivos. Para revisiones e inspecciones de código se recomienda utilizar una lista de verificación personal y actualizarla frecuentemente a partir de los defectos de compilación y pruebas encontrados en los programas que se han revisado e inspeccionado. Para realizar revisiones de diseño efectivas es necesario que el diseño esté bien documentado y que se analice la lógica del programa.

## Capítulo 5. Fase de Requerimientos.

El objetivo es describir el proceso de requerimientos en TSPi. Explicar qué son, por qué son necesarios y discutir las características más importantes.

### 5.1. ¿Qué son los requerimientos?

En esta fase el equipo genera la Especificación de Requerimientos de Software (ERS). En tal documento se hace una descripción clara de lo que será el producto; deberá incluir el criterio preciso para evaluarlo cuando esté terminado y asegurar que las funcionalidades sean las correctas. También proporciona retroalimentación al cliente acerca de lo que se pretende construir.

Se debe saber exactamente lo que el producto debe hacer antes de construirlo. Durante el desarrollo de los requerimientos se revisan las necesidades del cliente y se formulan preguntas acerca de las funciones del producto, si existieran necesidades que no son claras. A través de este proceso se logra un común acuerdo acerca de lo que se va a construir; por ésta razón es que cada integrante debe participar en la definición de los requerimientos y se define un proceso bien estructurado para construirlos. Los requerimientos se escriben en lenguaje coloquial y se revisan con los usuarios para verificar si realmente es lo que desean. El administrador de desarrollo dirige al equipo para formar el formato ERS e identificar todas las tareas que se deben realizar. Después de dividir las secciones el líder del equipo asigna a los integrantes y a establecer acuerdos en fechas de términos de cada una.

### 5.2. Cambios en los requerimientos.

Con frecuencia se suelen hacer cambios en los requerimientos. Generalmente, los usuarios no saben lo que necesitan en forma precisa hasta que tratan de usar el producto terminado. La parte difícil del proceso de requerimientos de software es entender qué es lo que los usuarios creen que necesitan y ayudarlos a definir esas necesidades en términos de funciones que generen un producto útil.

El siguiente paso es realizar la Especificación de Requerimientos que represente el acuerdo común entre los usuarios y el equipo de trabajo acerca de lo que necesitan. Es importante iniciar un acuerdo preciso de lo que es el producto y así tener un mecanismo que establezca los desacuerdos encontrados; eso será posible con una Especificación de Requerimientos clara y precisa, en la cual se podrán administrar los cambios y sustente tal documento contra un cliente que insista en interpretar de manera diferente una función que ya haya sido acordada.

### 5.3. Extracción de los requerimientos.

Para obtener los requerimientos se interroga a los clientes, usuarios y otros involucrados, para determinar lo que se necesita realmente.

Los pasos importantes para la obtención de los requerimientos son los siguientes:

1. Evaluar la factibilidad del sistema.
2. Conocer y comprender los asuntos organizacionales.
3. Identificar a todos los involucrados en el proyecto.
4. Registrar las fuentes de los requerimientos.

5. Definir el ambiente operativo del sistema.
6. Evaluar asuntos del negocio.
7. Definir las restricciones del dominio.
8. Registrar la razón de los requerimientos.
9. Realizar un primer prototipo de los requerimientos (al inicio pobremente entendido).
10. Definir los casos de uso.
11. Definir los procesos operacionales.

#### **5.4. Especificación de los requerimientos (ERS).**

En este documento el equipo describe las funciones que se planean desarrollar, cómo se pretende generar el producto, puesto que es una responsabilidad de la fase del diseño.

En TSPi el equipo se concentrará en los requerimientos operativos, mencionando a continuación los principales:

- Requerimientos funcionales: entradas, salidas, procesos y casos de uso.
- Requerimientos de interfaz externos: usuario, hardware, software y comunicaciones.
- Restricciones del diseño: formatos de archivo, lenguajes, estándares, compatibilidad.
- Atributos: disponibilidad, seguridad, mantenimiento, conversión.
- Otros requerimientos: Base de datos, instalación.

A continuación se muestran los puntos clave del ERS.

- 1 Tabla de contenido.
- 2 Introducción.
  - 2.1 Propósito del documento Especificación de Requerimientos de Software.
  - 2.2 Definición del problema.
  - 2.3 Información del equipo.
- 3 Requerimientos funcionales.
  - 3.1 Definición de los requerimientos funcionales del sistema
  - 3.2 Requerimientos del ciclo 1.
  - 3.3 Requerimientos del ciclo 2.
  - 3.4 Estructura top-down (estructura de lo general a lo particular)
- 4 Definición de reglas utilizadas en los requerimientos.
- 5 Requerimientos de interfaz externa.
  - 5.1 Interfaz del usuario.
  - 5.2 Formatos en pantalla.
- 6 Restricciones en diseño/implementación.
  - 6.1 Acuerdos en estándares.
  - 6.2 Restricciones en desarrollo.
- 7 Requerimientos especiales del sistema
  - 7.1 Documentación.
  - 7.2 Compatibilidad.
- 8 Referencias y fuentes de información.

### **5.5. Seguimiento del trabajo.**

Para asegurar la completéz, enumerar los párrafos y secciones de la ERS para identificar cada requerimiento. En la fase de estrategia el equipo realizó un seguimiento de los mismos, el cual continúa en esta fase y en el diseño.

### **5.6. Revisión de la definición de Necesidades.**

El administrador de desarrollo dirige al equipo en la revisión y se debe centrar en aquellas funciones que se van a incluir en cada ciclo de desarrollo. Sin embargo no se puede tener certeza absoluta de cuáles funciones serán necesarias hasta que se haya terminado el diseño.

### **5.7. Aclaración en la definición de necesidades.**

Después de revisar la definición de las necesidades, el administrador de desarrollo revisa la lista de preguntas y las notas que se tienen que aclarar con el cliente, para ello, se requiere hacer una reunión con el cliente, puede estar el equipo completo, y posteriormente documentar las respuestas a las preguntas y dar copias a los integrantes del equipo.

### **5.8. Documentación de los Requerimientos.**

En la documentación del ERS se realiza una definición breve y clara de lo que se pretende construir. Para los casos de uso, se listan los pasos en un guión simple, se usan tablas y listas con viñetas, tratar de evitar párrafos con demasiado texto. Una forma de asegurar el acuerdo entre el equipo es hacer el caso de uso de cada función, los cuales pueden servir como escenarios de pruebas del sistema para verificar las funciones del producto. Cada tarea terminada se debe entregar al administrador de desarrollo quien las debe unir en un borrador de ERS y dar copias a los demás ingenieros para realizar la revisión.

### **5.9. Inspección de los requerimientos y del Plan de Pruebas del Sistema.**

Se debe realizar una inspección en equipo del borrador del ERS y del Plan de Pruebas del Sistema. Se recomienda tener tiempo suficiente para tal inspección (una hora por página con espaciado sencillo). El objetivo de la inspección es poder encontrar problemas e inconsistencias antes de iniciar la labor de diseño. El administrador de calidad y proceso debe dirigir la inspección y utilizar la forma REGD<sup>5</sup>.

### **5.10. Plan de Pruebas del Sistema.**

A partir de la fase de requerimientos se empiezan a planear las pruebas del sistema, para asegurar que contenga los acuerdos pactados entre los usuarios y el equipo de trabajo. Probar bajo condiciones de error considerando la usabilidad y los resultados de recuperación. Todo esto se debe mencionar en el plan y explicar cuales sí y cuales no se evaluarán y las razones para eso.

### **5.11. Actualizaciones de requerimientos.**

Después de que aplicó la inspección en los requerimientos y se obtuvo la aprobación, se deben actualizar los documentos ERS y Plan de Pruebas del Sistema de acuerdo con los problemas encontrados en la inspección. Las secciones corregidas se proporcionarán al

---

<sup>5</sup> Para mayor información acerca de las formas consultar la página:  
<http://ada.fciencias.unam.mx/Ingenieriasoftware/ligas.html>

administrador de desarrollo, quien las reúne para obtener el documento final del ERS y distribuir las copias al resto del equipo.

#### **5.12. Revisión del ERS por el usuario.**

Después de producir, inspeccionar, y corregir el ERS y el plan de pruebas del sistema, los usuarios finales deben leer el ERS y estar de acuerdo en lo que se describe en ellos. Esto es una base para generar un producto de calidad y para prevenir los cambios después del ERS aprobado.

#### **5.13. Línea base de los requerimientos.**

El administrador de configuración documenta la línea base el documento ERS, el equipo puede solicitar cambios solamente utilizando el procedimiento de control de cambios mediante el formato de solicitud de cambios.

## Capítulo 6. Fase de Diseño.

El objetivo es proporcionar un diseño completo y de alta calidad que se utilice como base para la fase de implementación.

Diseño es el principio creativo mediante el cual se decide cómo construir un producto, se debe contener una especificación completa y precisa de la construcción del producto. En un diseño completo se definen las partes principales de un producto, se describe cómo esas partes interactúan y se especifica cómo unirlos para producir el producto final. Cuando el diseño de alto nivel es completo y preciso, los ingenieros rápidamente pueden producir los diseños detallados de los componentes; para eso, necesitan conocer las especificaciones funcionales completas de cada componente, sus interfaces y comportamientos de estado.

Cuando se diseña un producto en equipo, las principales preguntas conciernen cómo producir el diseño y el orden en el cual se van a diseñar las diferentes partes de los productos. Un problema común en el diseño de sistemas grandes de software es la necesidad de definir la estructura completa del sistema antes de especificar cualquier otra cosa. Mientras los diseñadores del sistema producen las especificaciones de los componentes externos, se pueden identificar las alternativas posibles para diseñar esos componentes y construir prototipos.

### 6.1. Criterios para el diseño.

**Propósito:** Guiar al equipo en el desarrollo e inspección de las especificaciones del diseño para un proyecto de desarrollo en equipo.

### 6.2. Estándares de diseño.

Existen varios estándares del diseño, de los cuales se consideraron los más importantes:

- **Convenciones de nombrado:** Es el primer estándar en el cual se especifica la estructura del nombramiento. El administrador de configuración debe establecer el glosario del sistema. Se deben definir los nombres de los tipos jerárquicos de programas (tales como sistema, producto, componente, módulo u objeto), las convenciones usadas en nombres de programas, archivos, variables y parámetros; y los procedimientos para establecer, controlar y cambiar nombres.
- **Formato de interfaz:** Aquí se definen los formatos y el contenido de las interfaces de los componentes. Esto consiste en indicar parámetros para variables, código de error u otras condiciones.
- **Mensajes del sistema y de error:** Es conveniente establecer los procedimientos para mensajes de error y del sistema. Un sistema útil debe tener consistencia y los mensajes deben ser comprendidos fácilmente.
- **Estándares de representación:** El estándar de representación del diseño define el producto que se generará a partir del diseño. Es importante definir y utilizar éstos estándares porque una representación del diseño imprecisa o ambigua puede producir problemas graves en la implementación y en las pruebas. Por eso es necesario documentar cada diseño, usar especificación operacional, funcional, de estados, lógica y algunos otros tipos de representaciones, esto permite ahorrar tiempo y producir un diseño revisado, el cual pueda ser verificado antes de la implementación.

- **Estándares de documentación de reuso.**

Los estándares de documentación marcan la diferencia entre las partes de reuso y las que no lo son. Un buen reuso permite que los ingenieros puedan utilizar un programa sin tener que analizar el código fuente. Por eso las partes de reuso deben contener una especificación completa del comportamiento externo, también es buena idea añadir una sección de comentarios al principio del programa fuente en cada parte.

### **6.3. Asignación de tareas en la fase de diseño.**

El administrador de desarrollo dirige al equipo en la generación de la Especificación de Diseño de Software (EDS) identificando el trabajo necesario y dividiéndolo en secciones. Con la ayuda del líder del equipo se asignan las secciones a los integrantes del equipo y se compromete a la fecha de entrega.

### **6.4. La especificación de diseño (EDS).**

Mientras se produce el documento EDS, defina las interfaces externas y las especificaciones funcionales de cada componente, genere un conjunto completo de escenarios que cubran las funciones externas de cada uno. El paso final en el diseño de alto nivel es producir los diversos documentos de diseño especificando la lógica y la estructura del programa principal del sistema para asegurar que las funciones de los componentes y las interfaces están completamente especificadas. Al producir las EDS, cada integrante debe revisar su trabajo y arreglar los problemas, después de eso se debe entregar al administrador de desarrollo para que lo incorpore en el borrador completo del EDS y distribuya copias al equipo.

### **6.5 Estructura completa del producto.**

Al definir la estructura completa del producto en el ciclo 1, primero se produce el diseño conceptual de alto nivel, en el cual se identifican los componentes, se nombran las funciones generales y se especifica la relación entre las mismas. Para documentar el diseño se pueden usar gráficas de flujo, casos de uso.

Para asignar las funciones del producto a los componentes utilice la tabla de seguimiento similar a la tabla definida en la estrategia. Para indicar cada componente y las funciones correspondientes, se recomienda seguir estos pasos:

- Agrupe las funciones por componentes, junto con las referencias de párrafos de Especificación de Requerimientos de Software por cada función.
- Asigne una letra a cada componente para hacer referencia a las funciones que se generan en cada ciclo.
- Indique si una función es generada mediante varios componentes

### **6.6. Calidad en la parte de reuso.**

La alta calidad es una parte importante en la estrategia de reuso. Para obtener una calidad de alto nivel se utiliza un proceso definido completamente y efectuar revisiones personales e inspecciones entre colegas sobre el diseño y el código.

Deberá considerar las siguientes preguntas al decidir reutilizar un programa:

1. ¿El programa tiene funciones correctas?
2. ¿La interfaz del programa es portable en la nueva aplicación?
3. ¿Todos los materiales necesarios están disponibles? Por ejemplo: código fuente, casos de pruebas, datos de pruebas e instrucciones de la aplicación.
4. ¿El programa es portable en cuanto a estándares? como el nivel del lenguaje; estándares de codificación; estándares en nombramiento y en archivos; y estándares en mensajes y ayuda.
5. ¿Se puede demostrar que el programa es de alta calidad?

Si existe alguna respuesta negativa es recomendable dedicarle tiempo considerable al componente para reuso.

### **6.7. Plan de Pruebas de integración (PPI).**

Es importante generar éste documento en las especificaciones de diseño. Con las pruebas de integración se van a revisar y verificar todas las interfaces entre los componentes del sistema, para ello también se debe indicar la manera para hacer las pruebas. Para asegurar que a todas las interfaces se les han aplicado las pruebas es conveniente inspeccionar el Plan de pruebas de integración al inspeccionar la EDS.

### **6.8. Soporte de aplicación.**

En TSPi el administrador de configuración es el responsable de proporcionar las partes de reuso y mantener registros completos de las mismas, también mantiene los puntos de alto nivel de reuso en el equipo, y durante el diseño e inspecciones de código, ayudando al equipo a identificar los reusos probables y asegurar que éstas son las más usadas.

### **6.9. Diseñando para usabilidad.**

Uno de los caminos para hacer productos usables es producir escenarios para cada función. Analizar esos escenarios y asegurar que éstas reflejen los tipos del sistema que se creen que los usuarios desean. Cuando un equipo planea adecuadamente las pruebas del sistema y de integración, generalmente se encuentran más defectos durante la planeación de pruebas. Un diseño completo y un conjunto claro de escenarios pueden acelerar las pruebas de planeación y mejorar la efectividad de las pruebas.

### **6.10. Pruebas de caja blanca y caja negra.**

Existen dos tipos de pruebas: caja negra y blanca. La primera verifica las especificaciones externas del sistema y no considera su estructura interna. La segunda, considera los caminos y la estructura lógica del programa. Las pruebas de caja negra a nivel del sistema simulan la manera en la cual el usuario va a usar el sistema. Respecto a los módulos o componentes individuales requieren controladores de interfaz, así como programas de soporte especiales para simular el ambiente del sistema. Las pruebas de caja blanca permiten evaluar aspectos algorítmicos.

### **6.11. Revisiones e inspecciones en el diseño.**

Las revisiones y las inspecciones en el diseño ayudan a incrementar su calidad. Para la inspección lo primero que se necesita es tener un diseño bien documentado y posteriormente preparar la inspección. Se tiene que revisar cada elemento del diseño: las interfaces, la inicialización de ciclos, cada paso y su terminación; y analizar el comportamiento de los estados.

### **6.12. Inspección del diseño.**

El administrador de calidad y proceso dirige al equipo en la inspección del borrador de EDS y del PPI. El equipo completo debería participar en la inspección para encontrar el número máximo de defectos y asegurar que todos entienden el diseño. Documentar la inspección y registrar los defectos en la forma REGD.

### **6.13. Actualización del diseño.**

Después de la inspección del diseño, corregir los errores identificados en la EDS y en el PPI, si fuese necesario haga que uno o dos de los ingenieros revisen las correcciones, después de eso proporcione la EDS y las secciones del plan de pruebas al administrador de desarrollo, quien produce y distribuye los documentos terminados.

### **6.14. Línea base del diseño.**

El administrador de desarrollo declara al documento EDS como línea base y obtiene una copia oficial del mismo.

### **6.15. Criterio de éxito.**

Los productos del diseño son los siguientes:

- La Especificación de Diseño de Software terminada, inspeccionada y corregida, incluyendo todos los materiales de diseño necesarios.
- El Plan de Pruebas de Integración terminado, inspeccionado y corregido.
- La forma de registro de Defectos REGD.
- Los datos de los ingenieros en tiempo y defectos registrados en las formas debidas.
- Copia de todos los materiales de diseño en la carpeta del proyecto.

## **Capítulo 7. Fase de Implementación.**

El objetivo es describir el proceso de implementación a través del criterio de diseño, estándares de implementación, estrategias de implementación, revisiones e inspecciones

### **7.1. Niveles de diseño.**

Cuando se pasa del diseño de alto nivel a la implementación se deben considerar los siguientes niveles:

- Sistema
- Subsistema
- Componente
- Módulo
- Clase

Este proceso de diseño interactivo continúa hasta que se hayan producido las especificaciones para los elementos básicos del sistema como objetos o rutinas de nivel menor que ya ha sido desarrolladas, las cuales pueden ser partes de reuso o funciones de bibliotecas disponibles.

### **7.2. Implementación paralela.**

Aunque es riesgoso empezar la implementación de alguna parte del sistema antes de que se hayan definido todas las especificaciones de alto nivel, se pueden minimizar algunos problemas dividiendo el sistema en componentes e iniciar la implementación si cada una de sus especificaciones externas están completas y han sido revisadas.

### **7.3. Estándares de implementación.**

Una vez que el equipo está de acuerdo con respecto de los estándares que se necesitarán, el equipo puede discutir sobre el contenido de las especificaciones estándares que usarán. Los estándares de implementación agregan elementos a los estándares definidos durante la fase de diseño y los extienden.

A continuación se tratan los siguientes temas relacionados con los estándares:

- Nombre, interfaces y mensajes estándares.
- Estándares de codificación.

### **7.4. Nombre, interfaz y mensajes estándares.**

Los estándares de nombres, de interfaz y de mensajes desarrollados durante la fase de diseño se revisan para asegurar que son apropiados y están siendo utilizados.

La lista de rutinas de reuso para verificar que estén terminadas y que todos los integrantes del equipo las estén utilizando.

El glosario de nombres para asegurarse de que cada módulo esté usando el mismo nombre para el mismo producto y que todos los nuevos nombres se van agregando al glosario.

Los nombres de los componentes, los nombres de las variables compartidas, de los parámetros y de los archivos se revisan para verificar su consistencia.

Las interfaces y los mensajes estándares para asegurarse de que hayan sido definidos, registrados en el glosario de nombres y que son conocidos y utilizados por los integrantes del equipo.

#### **7.5. Estándares de Codificación.**

Un estándar común de codificación asegura la consistencia en el código del equipo; esto facilita la realización de inspecciones, permite que se hagan de una manera más rápida y que sean efectivas. Un estándar de codificación bien estructurado define una sección para comentarios, lo cual permite mayor flexibilidad al momento de realizar incrementos en ciclos subsecuentes.

#### **7.6. La estrategia de revisión en la implementación.**

En las revisiones en la implementación sucede lo contrario, se inicia trabajando desde el detalle hasta llegar al panorama general. La estrategia de TSPi que se ha encontrado más eficiente es aplicar revisiones de funcionalidades desde el nivel más bajo al general (bottom-up). Primero se deben revisar todos los objetos que se encuentran en el nivel inferior. Después verificar que el desempeño de esas partes atómicas es el que se indicó en las especificaciones pase al nivel superior. Cuando se encuentren objetos ya revisados en el nivel anterior evite revisarlos de nuevo.

#### **7.7. Estrategia de reuso en la implementación.**

El reuso mejora la calidad si se discuten en los planes de implementación en reuniones de equipos semanales. En ellas se proponen las bibliotecas de reuso y los ingenieros que están implementando pueden averiguar si alguien tiene algún objeto o rutina de reuso que ellos estén necesitando, o saber las partes de reuso con sus especificaciones funcionales; todo esto les permite utilizar de la mejor manera el tiempo dedicado a la implementación.

#### **7.8. Revisiones e inspecciones.**

En esta parte se discuten las revisiones de implementación y sus inspecciones que contienen los siguientes puntos:

- Errores aleatorios.
- Inspección de diseño en programas fuentes.

#### **7.9. Errores aleatorios.**

Muchos de los defectos de implementación se introducen al momento de escribir el código, a tales errores se les llama errores aleatorios, los cuales no tienen ningún fundamento lógico.

Los errores aleatorios pueden causar un gran impacto, sobre todo en sistemas grandes. Además son difíciles de detectar al momento de realizar las pruebas, porque sólo se encuentran los errores que están bajo condiciones de pruebas específicas.

#### **7.10. Criterio de entrada.**

Para la implementación en el primer ciclo se debe tener el plan de desarrollo y la estrategia terminados; las especificaciones de requerimientos y de diseño terminadas, revisadas y actualizadas; un estándar de codificación definido y documentado; copias disponibles en las cuales se integran la lista de especificación funcional del producto; el glosario de nombre y otros estándares que el equipo haya definido.

#### **7.11. Planeando la implementación.**

En la planeación de la implementación, el primer paso es revisar el trabajo a hacer y estar seguro que todas las tareas sean asignadas a los miembros del equipo. Lo mejor es que los integrantes que realizaron el diseño sean quienes realicen la implementación, lo cual se deja al criterio del equipo. El administrador de planeación mantiene actualizado al equipo.

#### **7.12. Diseño detallado y revisión en diseño.**

A partir de este punto se está listo para desarrollar el diseño detallado para la implementación. Después se realiza la revisión del diseño.

#### **7.13. Desarrollo de Pruebas.**

Después de que se arreglaron los errores que se encontraron en la revisión, se desarrolla el código para las pruebas unitarias. Generalmente, durante el desarrollo de las pruebas se encuentran más errores del diseño que en la inspección de diseño o en las pruebas unitarias, por eso es conveniente aplicar las pruebas antes de la inspección en el diseño detallado.

#### **7.14. Inspección en el diseño detallado.**

Después de la aplicación de pruebas, el administrador de calidad y proceso dirige la inspección del diseño detallado. Los errores mayores se deben registrar en el formato REGD y proporcionar las copias al líder del equipo y administrador de calidad y proceso para integrarlas en la carpeta.

#### **7.15. Revisiones de codificación y código.**

Después de la inspección de diseño, se realiza la codificación. También se revisa la consistencia en nombres, inicializaciones, puntuaciones, llamadas de secuencias y de inclusión (*include*). Se lleva el control del tiempo utilizado en la búsqueda de defectos. Se pide a otro ingeniero que revise el código antes de que se compile.

#### **7.16. Inspección en Código.**

Después de la compilación se compara el tiempo dedicado al diseño y su revisión, al código y su revisión con los tiempos del plan de calidad del equipo. Se aconseja utilizar la siguiente tabla de Criterio Estándar de Calidad:

- El tiempo dedicado al diseño debería ser mayor que el tiempo de la codificación
- El tiempo dedicado a la revisión del diseño debería ser mayor que el 50% del tiempo del diseño.
- El tiempo dedicado a las revisiones de código debería ser mayor que el 50% del tiempo de codificación y mayor que el 75%.
- Por lo menos se encontraron dos veces el número similar de defectos en la revisión de código como en la compilación.
- La tasa de revisión fue menor a 200 LOC por hora.

## Capítulo 8. Pruebas de Integración y del Sistema.

El objetivo es integrar las pruebas y su documentación. Indicar los objetivos, las estrategias y planeación de las pruebas.

### 8.1. Estrategia de Pruebas de TSPi.

La estrategia de pruebas tiene como objetivo verificar que los productos a los que se les aplican las pruebas son de alta calidad. Las actividades principales son las siguientes:

1. Utilizar las partes desarrolladas a las que se les ha aplicado pruebas unitarias para construir el sistema.
2. Utilizar las pruebas de integración para verificar que el sistema está debidamente construido, que todas las partes se encuentran desarrolladas y que funcionan en conjunto.
3. Validar con las pruebas del sistema que el producto realiza los requerimientos del sistema.

De manera simultánea se puede:

1. Identificar módulos o componentes de baja calidad y entregarlos al administrador de calidad y proceso para su evaluación y depuración.
2. Identificar componentes de baja calidad que estén causando problemas, aún después de que se ha depurado; regresarlos al administrador de calidad y proceso para que trabaje de nuevo con ellos o los reemplace.

Se construyen los planes de las pruebas de integración y del sistema. Para la construcción se reúnen las diferentes partes del sistema dentro de un sistema completo de manera que se encuentre listo para la fase de pruebas. A esto se le llama la *construcción del sistema*. Las pruebas de integración aseguran que todas las partes han sido incluidas en el sistema y que sus interfaces son compatibles. Finalmente las pruebas del sistema validan las funciones y el desempeño del sistema de acuerdo a los requerimientos.

### 8.2. Estrategia de pruebas del sistema.

Consiste en evaluar cada una de las funciones bajo condiciones de estrés; evaluar la usabilidad y el desempeño. Una de las estrategias consiste en juntar todas las áreas funcionales tratando de cubrir todos los aspectos antes de pasar a la siguiente. La estrategia evita que las pruebas se dupliquen, pero no trabaja con el comportamiento del sistema.

Otra estrategia consiste en aplicar pruebas a las funciones de nivel mas bajo en condiciones normales, anormales y de estrés. Después se deberá continuar al nivel superior y probar con funciones agregadas para verificar que funciona. Después se aplican las pruebas bajo condiciones normales y de estrés. Se deberán aplicar las mismas pruebas a los niveles superiores hasta cumplir con el sistema completo. La estrategia es útil para los sistemas de baja calidad porque existen funciones que no funcionan desde el inicio. La desventaja es que se requiere bastante tiempo para probar todas las combinaciones de funciones en un sistema grande.

La siguiente estrategia es el camino inverso. Se inicia con las funciones de nivel superior y posteriormente se pasa a niveles inferiores, aplicando pruebas bajo condiciones normales y de estrés. El sistema se prueba de acuerdo a la funcionalidad, casos de uso y guiones de pruebas. Aunque cubre el sistema más rápidamente.

Esta estrategia consiste en reunir todas las partes al mismo tiempo. Primero se tienen dos partes funcionando y después se agrega la tercera y se vuelven a poner a funcionar. Entonces se pueden poner la cuarta y quinta parte y así sucesivamente. La desventaja que tiene esta estrategia es que requiere mayor trabajo en la fase de pruebas.

### **8.3. Planeación de las pruebas.**

Se realiza en diferentes momentos de acuerdo al proceso de TSPi. El plan de pruebas describe que se planea aplicar, el orden de las mismas y los materiales necesarios. Se necesita el plan de pruebas para construir las pruebas de integración, y las actividades de prueba del sistema. Aunque el construir y el planear las pruebas puede ser simple, es importante planear las pruebas antes de hacerlas.

Con el plan completo se debe mostrar la manera en la que cada requerimiento será probado y cómo cada escenario o guión cubre las áreas de los requerimientos, también se debe saber que áreas serán probadas y cuales tendrán un amplio cubrimiento. Además se deberá nombrar a cada prueba por anticipado, estimar los defectos encontrados en cada fase, el tiempo total en la detección de defectos y el tiempo de pruebas totales. Después se deberán estimar los materiales requeridos.

Al final de la planeación de las pruebas se deberá tener:

- Una lista de todos los pasos que se van a desarrollar en las pruebas.
- Los materiales requeridos, las características para cada una y su tiempo de desarrollo.
- Los posibles resultados que generarán.
- Estimación del tiempo de ejecución libre de defectos, de los defectos que serán encontrados y del tiempo total para cada prueba.
- Estimación del trabajo requerido.
- Los objetivos de cada prueba.
- El responsable de desarrollar cada material y la fecha final de terminación.

### **8.4. Registro de pruebas.**

En cada ciclo de desarrollo de TSPi se ejecutará un conjunto completo de pruebas. En el formato de pruebas se deberá indicar lo siguiente:

- El nombre de quien registra la prueba.
- Nombre o número de las pruebas.
- El producto y configuración que se está probando.
- Los resultados de las pruebas.
- Herramientas o facilidades utilizadas.
- Si se requirió la intervención de algún operador y cuánto tiempo.
- Se deberá anexar el formato de registro de pruebas en la carpeta.

### **8.5. Módulo de defectos.**

Cuando las pruebas que no están bien pensadas, se pueden generar muchos errores después de las pruebas. En otras palabras, la mayoría de los errores que se encuentren en las pruebas muchos de ellos no se podrán corregir. Utilice los datos de pruebas para evaluar el riesgo que el sistema tendría en una o más partes defectuosas. Ordene los datos de defectos por módulo para encontrar cuales tuvieron la mayoría de los defectos encontrados en cada prueba. Generalmente, aquellos módulos con la mayoría de los defectos mantendrán los mismos después de las pruebas y de manera general no serán encontrados. Se aconseja que los módulos con esas características se suspendan las pruebas temporalmente, se reinspeccione y corrijan tales componentes defectuosos antes de continuar con las pruebas.

### **8.6. Seguimiento en los datos defecto.**

Para el seguimiento y análisis de los defectos por módulo, se necesitan los datos de cada defecto para cada prueba. Es importante que en las reuniones semanales se revisen los defectos encontrados en la construcción, integración o pruebas del sistema con el equipo completo. Posteriormente el administrador de calidad y proceso deberá dirigir una revisión de todas las pruebas de construcción e integración, y de los defectos de las pruebas del sistema con el equipo completo. También alguien deberá buscar y arreglar los defectos no encontrados. Se deberá actualizar el proceso de acuerdo a los cambios identificados.

### **8.7. Documentación.**

En la fase de pruebas de TSPi, algunos miembros del equipo hace un bosquejo y las revisiones de la documentación del usuario mientras que el resto del equipo realiza las pruebas. Durante el primer desarrollo del ciclo es buena idea asignar más ingenieros a las pruebas y después en los ciclos subsecuentes incrementar el número de ingenieros asignados a la documentación.

La documentación es parte esencial de cada producto de software. El mejor momento para documentar una función es después de que se ha diseñado. Por otro lado si se aplaza el trabajo de documentación, éste podría tomar mucho más tiempo. TSPi incluye el trabajo de generación de documentación en la fase de pruebas.

### **8.8. Diseño de la documentación.**

Después de que se construyó el producto, es natural querer describir lo que se construyó. Desafortunadamente lo normal es que se quiera hacer la creación de lo que se hizo y no lo que el lector necesite. Escribir manuales de usuario útiles es un reto para los ingenieros de software. Una guía completa para la determinación de la calidad de un manual es la tabla de contenidos. Si el manual está organizado con respecto al diseño del producto, esto debe de seguir las necesidades del lector y no de la estructura del producto. En general, la primera sección explica al usuario cómo empezar. Después se deberá explicar al usuario qué puede hacer con el producto. A continuación se presenta una guía útil:

- Utilizar un glosario para definir los términos que no estén en el diccionario estándar.
- Incluir una sección con mensajes de error y de recuperación de errores.
- Tener un índice de términos importantes.
- Proporcionar una tabla de contenido detallado.

### **8.9. Línea de documentación.**

El primer paso en la documentación es producir un bosquejo detallado. Empezar con un nivel alto y después diseñarlo. Antes de empezar a escribir el texto, verificar el trazo para asegurar que se cubren todos los objetivos en la construcción de los escenarios de usos. La única excepción son aquellos escenarios que serán cambiados en subsecuentes ciclos en el desarrollo. Si se describen en ese momento probablemente sería pérdida de tiempo.

Después de que se termina la línea de documentación se revisa con los ingenieros que están creando el plan de pruebas para asegurar que se entiendan las mismas funciones en la misma dirección y que nadie ha omitido algo importante. La documentación y el plan de pruebas dan una perspectiva diferente para el uso. De hecho el equipo encuentra más defectos en la documentación y en la planeación de las pruebas que durante las pruebas.

### **8.10. Estilo de escritura.**

En general hay que escribir frases cortas, utilizar palabras simples y aplicar viñetas cuando sea necesario. Cuando se organizan las palabras en forma de lista, es más fácil para el usuario entienda el trabajo y ver cómo se hace éste.

### **8.11. Revisión de la documentación.**

Se aconseja que uno o más colegas revisen la documentación generada. Si el producto es para clientes o usuarios, aplique pruebas de usuario para asegurarse de que la escritura clara, precisa, completa y entendible. A continuación se presentan los siguientes elementos para las revisiones:

- Organización de los documentos. ¿El documento está organizado de acuerdo a lo que el usuario hará o de acuerdo a sus necesidades y no a la estructura del contenido del producto.
- Terminología del documento. ¿El documento presume conocimiento que el usuario probablemente no tenga? Debe definir cualquier palabra en el diccionario.
- Contenido del documento. ¿El documento cubre el contenido del producto?
- Precisión. ¿El desarrollo de los métodos y procedimientos es de acuerdo a su descripción.?
- Legibilidad. ¿Es fácil de leer el documento? Lea el documento verificando que es fácil decir lo que está escrito.
- Entendimiento. ¿Las personas entienden lo que está escrito? Generalmente, esta pregunta es la más difícil de responder. Las mejores pruebas consisten en solicitar a alguien que no ha utilizado el sistema que lo haga. Obsérvese sus reacciones, registre los problemas y modifique el documento de acuerdo a los problemas identificados.

## Capítulo 9. Postmortem.

**Objetivo:** Describir la necesidad del postmortem, su proceso y utilidad

En el Postmortem se revisa el trabajo del equipo para asegurar que se han completado todas las tareas necesitadas y todos los datos requeridos y registrados. También se re-examina lo hecho en el ciclo, qué fue lo correcto y a ver cómo hacer el trabajo mejor para el siguiente ciclo. El postmortem nos da un camino ordenado para identificar las áreas que necesitan ser mejoradas y crear los cambios necesarios.

En TSPi el ciclo concluye en la fase de postmortem, por lo cual nos da un camino estructurado para aprender y mejorar. En el postmortem se examina que se hizo comparado con lo que se planeó hacer. Se busca oportunidades para mejorar y decidir si cambiar las prácticas para el siguiente ciclo.

En la fase de postmortem, se verán los cambios para el otro ciclo. En el primer ciclo TSPi da la línea base; se evalúan los productos producidos, el esfuerzo realizado, y los pasos del proceso que se deben seguir. Se determina que tan adecuado fue el plan con el proceso seguido. Se identifican los problemas, determinan sus causas, se trazan medidas de prevención. La fase de postmortem verifica oportunidades de mejoras y específicamente se decide dónde y cómo incorporar esos cambios en el personal y en el proceso del equipo.

### 9.1. Propósito de Mejora en el Proceso.

La clave en las mejoras exitosas son los cambios pequeños. Cada mejora pequeña ayudará un poco y combinando esos pequeños cambios se verá un cambio significativo. El problema con los cambios pequeños, sin embargo es que son fáciles de olvidar. Estas ideas pueden incluir en las mejoras prácticas personales, mejoras de las herramientas, cambio en los procesos, etcétera.

### 9.2. Revisión de calidad.

Como parte de la revisión del proceso se compara el desempeño personal y del equipo de acuerdo al plan de calidad. Se inicia con un análisis sobre los datos referentes a los errores y se evalúa uno o más errores en las pruebas del sistema, se deberá evaluar la parte del proceso que tuvo deficiencias para realizar mejoras en el futuro. Guíese mediante las siguientes preguntas:

- ¿Cómo comparó el desempeño actual con el planeado?
- ¿Cómo se puede aprender a partir de ésta experiencia?
- ¿Se debería aplicar un criterio personal o por equipo en ciclos posteriores?
- ¿Dónde se pueden aplicar mejoras y cuáles son las razones de ello?

Para postmortem posteriores evalúe las mejoras realizadas y juzgue la efectividad de las mismas. Guíese mediante las siguientes preguntas:

- ¿Tuvo deficiencias el equipo y dónde no?
- ¿Qué metas u objetivos son factibles de aplicarse en ciclos o proyectos posteriores?
- ¿Dónde se modificarían los procesos utilizados?

### **9.3. Evaluaciones de roles.**

El líder del equipo examina cada rol, en estas evaluaciones se fijan los objetivos. Se inicia con el líder del equipo y después se revisan los otros roles, para hacer esto se consideran las siguientes preguntas para cada rol:

- ¿Dónde surgieron los problemas?
- ¿Dónde se pueden realizar mejoras?
- ¿Qué metas mejoraron y cuales de ellas podrían utilizarse para el siguiente ciclo o proyecto?

### **9.4. Preparar el reporte del ciclo.**

Se reporta lo que se produjo, el proceso utilizado y los roles desempeñados; lo que si se pudo aplicar y lo que no fue conveniente, para ello se pueden hacer sugerencias que pueden aplicarse en el siguiente ciclo. Se deberá describir el desempeño que cada integrante tuvo respecto al rol que desempeñó, de acuerdo a los indicativos por TSPi así como el rol correspondiente al área de desarrollo. Si es posible, se justifican las conclusiones con datos actuales del equipo y se comparará el desempeño en ciclos anteriores. Tal reporte debe ser breve y objetivo.

El reporte del ciclo deberá tener el siguiente contenido:

- Índice.
- Resumen.
- Reporte de rol, en cuanto a:
  - Liderazgo.
  - Desarrollo
  - Planeación.
  - Proceso.
  - Calidad.
  - Apoyo o soporte.
  - Reporte de los ingenieros.

### **9.5. Reporte de roles.**

El líder del equipo será el responsable de generar el índice y el resumen, en ese último se describirán fortalezas y debilidades de cada área brevemente.

El en reporte de roles se sugiere añadir dos secciones, una en la que se describirá la evaluación sobre el rol desempeñado individualmente y la evaluación sobre el desempeño del equipo de acuerdo a la perspectiva de TSPi sobre cada rol.

El objetivo de este reporte consiste en proporcionar una guía para desempeñar de mejor manera cada rol tomando en cuenta las recomendaciones escritas.

El líder del equipo deberá analizar el desempeño del equipo desde la perspectiva del liderazgo, tomando en cuenta asuntos de motivación y acuerdos establecidos. También deberá tomar en cuenta las reuniones del equipo y mejorar esa responsabilidad en el futuro.

El administrador de desarrollo deberá comparar el contenido del producto con los requerimientos y evaluar la efectividad de la estrategia de desarrollo; para ello deberá analizar si la estrategia generó los resultados esperados y si deberá identificar que cambios podrían aplicarse. Respecto al diseño y a la implementación se deberá analizar el reuso aplicado, el desempeño, la compatibilidad, instalación, mantenimiento, seguridad, etcétera.

El administrador de planeación compara el desempeño del equipo real con lo que se planeó, tomando en cuenta las horas semanales y los valores ganados, también analizar el desempeño del equipo en ciclos anteriores en otros proyectos.

El administrador de calidad y proceso compara los datos actuales con los objetivos de calidad establecidos para describir el desempeño del equipo. Muestra las tendencias de calidad de los ciclos de desarrollo terminados hasta la fecha actual. En la sección de proceso deberá evaluar la disciplina del mismo, tomando en cuenta hasta qué grado los integrantes siguieron el proceso, también deberá revisar el seguimiento en los estándares establecidos y en las revisiones aplicadas.

El administrador de configuración describe las facilidades de soporte, problemas enfrentados y mejoras en cualquier área. Hace comentarios sobre el procedimiento de control de cambios y la administración de configuración, tomando en cuenta aspectos que intervinieron en los cambios, los impactos de los mismos, hace recomendaciones para un mejor manejo de ciclos o proyectos futuros.

#### **9.6. Reporte de Ingenieros.**

Cada ingeniero debe describir el desempeño personal de acuerdo a lo planeado y la calidad del trabajo que realizó. Es importante indicar cómo mejorar el trabajo para la siguiente vez. Cada ingeniero deberá entregar al líder del equipo sus comentarios. El administrador de calidad y proceso revisa el reporte completo y sugiere las correcciones necesarias. Cuando el reporte final del ciclo esté correcto deberá entregarse al instructor y a los integrantes del equipo.

#### **9.7. Evaluaciones de los roles y Formato EEC.**

Consiste en llenar el formato de Evaluación del Equipo por Colegas. En tal formato se evalúa el desempeño del equipo y de cada rol. En la primera sección se evalúa el trabajo requerido y la dificultad de los roles en cada una de las actividades respectivas. Se asigna a cada rubro un porcentaje correspondiente, donde la suma total debe ser igual a ciento por ciento. Para los siguientes rubros, en cada aspecto se selecciona el número correspondiente, el rango, inicia en 1 y termina en 5, catalogado desde inadecuado hasta superior, conforme avanza la numeración.

## PARTE II

### Introducción al desarrollo

El Departamento de Vinculación y Capacitación Cultural de CONACULTA (Consejo Nacional para la Cultura y las Artes) haría una certificación de manera oficial con la SEP (Secretaría de Educación Pública) la cual consistiría en la que cualquier persona con nacionalidad mexicana que tomara cursos culturales y con a una serie de ellos pudiera obtener la certificación como promotor cultural y por tanto se necesitaba tener un software que pudiera mantener, administrar y garantizar que la información vertida fuera consistente y verificable. De allí nacería el proyecto que se llama SRNCC (Sistema de Red de Capacitación Cultural).

Un problema al que nos enfrentamos fue que el departamento gubernamental en el cual se desarrollaría SRNCC no constaba con el equipo adecuado ni con los alcances económicos para desarrollarlo, algo que no se nos fue notificado y el cual tuvimos que solucionar de tres formas: Una de ellas fue primero tratar de modificar el software actual para que el proceso de captura empezara a ser coherente y ordenado de manera que la certificación se pudiera llevar acabo y que la presión sobre el departamento de sistematización fuera aligerándose. La segunda, empezar a deshacernos de la papelería que estaba repetida y que ocupaba demasiado espacio, en síntesis fue ordenar la información en papel. La tercera fue conseguir un equipo de cómputo que tuviera la plataforma y el software necesario para empezar la programación del proyecto.

Una vez que se solucionaron estos problemas se pudo llevar acabo el arranque del proyecto SRNCC, pero en el cual aun habría muchas cosas que lo retrasarían continuamente. Por ejemplo, en las reuniones programadas para ir estableciendo las partes que conformarían el software se hacían cambios constantemente y se desplazaban las juntas hasta por tres semanas más; no había ningún responsable directo de la dirección, por lo cual, todos opinaban constantemente y nos llevaban a perdida de tiempo. Para resolverlo se asignó una subdirección encargada de la comunicación. Mas allá de los problemas burocráticos en los cuales se vio embebido el proyecto, valía la pena sacarlo adelante y ahora es una realidad, se han obtenido logros como un mejor equipo de cómputo, un servidor (con ciertas carencias), etcétera.

El desarrollar el software bajo estas condiciones nos han permitido aprender mucho, sobre todo, aun con la inexperiencia, no siempre existirán las condiciones ideales en todos los proyectos, por lo cual, creo que el habernos enfrentado a estos problemas nos permitió conocer más opciones donde podamos recurrir para vencerlos.

El desarrollo de las fases contiene diferentes componentes de los cuales se hablará a detalle más adelante, pero cada una de ellas será estará compuesta de una página principal que es la portada y la cual deberá estar formada de un título que menciona el nombre de la fase y una serie de productos.

En la fase de lanzamiento se establecieron diferentes objetivos para el proyecto, los cuales nos darían una idea de un producto de calidad, los datos registrados e incluidos, y el tiempo de tolerancia para terminar, también los objetivos de los miembros del equipo nos permiten saber exactamente que rol vamos a tener en el equipo, las responsabilidades y las características del trabajo que el equipo tiene que cumplir para iniciar el proyecto, esto es muy importante porque permite a los integrantes del equipo saber exactamente acerca de cada rol y de esa manera poder decidir con respecto a las capacidades individuales y tomar el que se acople a sus posibilidades y a las del equipo, así mismo en esta fase se establecieron los objetivos del producto, el cual nos permitió plasmar los antecedentes y tener una idea general de lo que serían los objetivos del proyecto. Se crea un estándar de documentación el cual es un documento que permite al equipo manipular la información que se creará en el futuro de una manera ordenada y sin errores, y por último se agregó la minuta, la cual contiene todos los asuntos relacionados que se trabajaron en la semana, cualquier comentario y dudas de los integrantes del equipo.

En la fase de estrategia se genera la estrategia de desarrollo la cual consiste en especificar de cuantos ciclos contendrá el proyecto, y sus funcionalidades de manera general, también generamos una gráfica de dependencias entre necesidades la cual se expresó aquellas funcionalidades que fueran de menor dependencia a la de mayor dependencia, esta gráfica se desarrolla expresando en lenguaje coloquial cada una de las necesidades N1 y se le asignaba la funcionalidad de menor dependencia, así hasta la Nn que sería la funcionalidad de mayor dependencia, para este caso N1 sería la “alta de instructores” y la N24 “Generación de datos de apoyo”, lo cual pudimos generar el formato ESTRA el cual consistiría en enumerar las funcionalidades a mayor detalle con la cantidad de líneas de código y el tiempo de trabajo de cada una estas para el ciclo uno, además dándonos una aproximación de la cantidad de trabajo que estaríamos desarrollando y cuanto tiempo nos ocuparía. También se agregó el formato RR el cual se escriben los posibles riesgos que el sistema bajo las funcionalidades que se generaron podría enfrentar, las personas asignadas a verificar esos riesgos y la gravedad de estos ya sea baja, media o alta.

En la fase de planeación el equipo conoce exactamente que cantidad de trabajo que tiene que realizar, la planeación de esas tareas a realizar es uno de los principales objetivos. En el diagrama de Gantt se vierten las tareas y se distribuyen en un tiempo determinado para realizarlas, este diagrama permite al equipo tener una forma gráfica de cómo va a poder distribuir las de manera que siempre sea sencillo revisar en caso de cualquier duda. El trabajo de planeación que el equipo decidió fue de aproximadamente seis meses en el ciclo uno, que va desde el 14 de enero del 2003 al 9 de junio del 2003. La realización de las formas semanales permiten a el equipo como a cada miembro hacer estimados del total de horas en cada ciclo de vida en el proyecto, así mismo permite comparar el tiempo requerido para cumplir con todas ellas, también se genera el plan de inspecciones que permite al equipo llegar acuerdos de cómo van a ser las revisiones e inspecciones en las siguientes fases.

La fase de requerimientos nos permite ir conociendo cada una de las necesidades del cliente el cual especificamos con un objetivo y una definición general del sistema y así mismo con la información del equipo, generamos los requerimientos funcionales que son entradas y salidas, procesos, casos de uso, los caso de uso nos permitieron describir escenarios del sistema con pasos enumerados para los cuales se elaboraron 8 grandes módulos los cuales

se destinan 5 para el primer ciclo y 3 para el segundo y un prototipo para cada bloque; también lo que son otro tipo de requerimientos como inercias externas, restricciones del diseño e implementación, portabilidad, desempeño, usabilidad, etc. También en esta fase el equipo elabora el Plan de Pruebas del Sistema, este permite al equipo planear el tipo de pruebas al que sistema requerirá bajo condiciones de error considerando la usabilidad y los resultados de recuperación. Cada uno de estos aspectos son muy importantes para el buen desarrollo del sistema ya que establecen los requerimientos que el equipo, el cliente y el sistema tendrá que cumplir durante el desarrollo, en esta parte ya manejamos el formato REGD el cual nos apoyo en el manejo de las revisiones e inspecciones, como la fecha, número de error, la fase en la que se generó, el tiempo de corrección, si está corregido o no y la descripción de cada uno.

En la fase de diseño generamos el estándar de diseño el cual está conformado con los convenciones de nombramientos para las clases, los métodos, las variables, los parámetros y los archivos, los formatos de interfaz, los mensajes de error y del sistema; considerando el tipo, el mensaje y sus características, y estándares de diseño para el reuso, también creamos las especificaciones de diseño de software (EDS), el cual elaboramos el diseño arquitectónico que fue el de tres capas, el cual seleccionamos por que consideramos el más adecuado para nuestro diseño, este consta de una capa de presentación la cual nos permitió diseñar la estructura de el sistema en materia de interfaces, la capa lógica de presentación en la cual se integró el diagrama de clases y el diagrama de dependencias de cada uno de los módulos que nos serviría para generar la base de datos, así mismo se generaron 5 bloques de diagramas de secuencias, de las cuales hay en la segunda parte 2 y en el anexo 3 mas; generamos un diagrama de componentes con la finalidad de expresar de manera visual los componentes de instalación que contiene el sistema y por ultimo el plan de pruebas de integración el cual es un documento que nos apoyo mucho pues en él diseñamos el plan de cómo sería las pruebas, las características y si era necesario utilizar a diferente personas que nos apoyaran que no estuvieran involucradas en el proyecto, etcétera.

Para la fase de implementación se define un estándar de codificación que permite aclarar de manera detallada las características de programación para el lenguaje a utilizar en este caso el de postgres y JavaScript, abarcando la organización de los archivos, indentación, tipos de comentarios, declaraciones, sentencias y convenciones de los nombres. También se genero un diagrama de dependencia entre los módulos de manera tal, que con éste se podría hacer un mecanismo de implementación por orden de dependencia, abarcado desde el módulo mas dependiente al de menor dependencia, por lo que el módulo de promotores seria el mayor dependencia seguido de carteleras, instructores, coordinadores, y por último el modulo de menor dependencia, el de bienvenida. Una vez establecida la dependencia se genero una tabla de detalles de cada uno de los módulos, mostrando el contenido y sus especificaciones al igual para las funciones de reuso. En esta fase se empiezan a generar algunos cambios que serán tratados en la mesa de control de cambios, los cuales serían en el módulo de instructores, módulo de carteleras, y coordinadores. En esta parte también se genera una tabla con todas las funciones que se implementaran en el sistema, con las responsabilidades de cada una, tipos de variables que manejan y el módulo al que pertenecen. Por otro lado, se genera una grafica de archivos (un total aproximado de 95 archivos y 11 directorios), dividido por directorios de manera tal que éste funja como un mapa en la navegación de los archivos que se generaron en la implementación.

En la fase de pruebas de integración y del sistema se generaron las pruebas de integración a detalle las cuales se dividieron por módulos, el valor esperado, el valor probado y los respectivos comentarios, para el caso de las pruebas del sistema se dividieron por casos de uso, métodos, el tipo de error que se esperaba, las operaciones que se utilizaban y los comentarios respectivos de cada uno. Otro tipo de prueba que se aplicó fueron las pruebas de Stress las cuales dividieron también por módulo y consistieron en someter al sistema a cualquier tipo de imprevisto que pudiera surgir, de manera tal que las validaciones funcionaran adecuadamente. Por último se generaron las listas de verificación las cuales se revisa que el código cumpla con el estándar de verificación, la sintaxis, los métodos, los casos de uso, los diagramas de secuencia, la verificación de errores, de ejecución y validación por módulo así como su documentación.

La última fase, la de postmortem, se generaron los reportes de ciclo donde se resume los estados de cambios por fase y los estados de implementación, los tamaños de los productos por fase y el tipo de unidades de cada producto, se hace el reporte de roles de cada uno de los integrantes de manera que cada uno exprese su desempeño y las experiencias que se generaron durante el desarrollo. Se hizo una gráfica con el contenido de las pruebas mostrando los tiempos de respuesta en las pruebas del sistema por módulo, esto nos sirvió para darnos cuenta cuánto tiempo nos llevo en cada módulo y así en el segundo ciclo balancear mejor la cantidad de trabajo. Y por último la evaluación por cada rol, esta nos permitió saber que desempeño tuvimos en el desarrollo del sistema y en dónde debimos trabajar más, es una parte muy importante porque nos permitió saber qué cosas deberíamos mejorar para el siguiente ciclo.

# SISTCAP

---

## CAPITULO 10

### FASE DE LANZAMIENTO

#### Productos

- **Objetivos del proyecto.**
- **Objetivos de los miembros del equipo.**
- **Objetivos del rol.**
- **Objetivos del producto.**
- **Estándares para entrega de documentación.**
- **Minuta de reunión.**

## OBJETIVOS DEL PROYECTO

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2 página 6; "2.1 Objetivos del proyecto").

### 1. Generar un producto de calidad.

- Requerimientos de funciones incluidas en el proyecto: 100%

### 2. Realizar un proyecto bien administrado y que sea productivo.

- Porcentaje de datos registrados e incluidos en la carpeta del proyecto: 100%.

### 3. Finalizar a tiempo.

- Días antes o después en completar el ciclo de desarrollo no más de cuatro.

El proyecto SRNCC (Sistema de Red de Capacitación Cultural) consiste en un sistema de información WEB capaz de administrar, validar y salvaguardar la información que se genera a través de diferentes cursos culturales que se dan en toda la República Mexicana.

Por otro lado SRNCC tiene el objetivo de que se sistematicen y aceleren procesos en la captura y flujo de la información, que la información generada pueda ser consistente, que desarrolle documentos oficiales y de manejo en el departamento de Capacitación Cultural, que se pueda llevar a cabo una certificación la cual consistirá en que las personas que tomen los cursos puedan obtener la validez oficial como promotores culturales por la SEP (Secretaría de Educación Pública), además de que se realicen evaluaciones de los cursos y de quienes los imparten.

## OBJETIVOS DE LOS MIEMBROS DEL EQUIPO

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2 página 7; "2.2 Objetivos de los miembros del equipo").

### 1. Ser un miembro del equipo cooperativo y efectivo

### 2. Hacer un trabajo disciplinado

- Porcentaje de datos personales registrados en la carpeta del proyecto: 100%.
- Porcentaje de las semanas en las cuales se realizaron y finalizaron las actividades registradas en cada forma personal SEMANA: 100%

### 3. Planear y mantener un seguimiento de trabajo personal

- Porcentaje de datos del proyecto registrados en los formatos REGT:100%

### 4. Generar productos de calidad.

- Promedio de porcentaje de defectos encontrados antes de la primera compilación no menos del 70%
- Densidad de defectos encontrados durante la compilación: 10/KLOC
- Densidad de defectos encontrados durante las pruebas unitarias: 5/KLOC
- Densidad de defectos encontrados después de la prueba unitaria: 0

## OBJETIVOS DEL ROL

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2 páginas 8 y 9; "2.4 Objetivos del rol").

### Líder del equipo.

1. Construir y mantener un equipo eficaz.
2. Motivar a todos los miembros del equipo a trabajar agresivamente sobre el proyecto.
3. Resolver todos los problemas del equipo que susciten.
4. Tener a todos los miembros del equipo y al instructor informados sobre los avances.
5. Actuar efectivamente como moderador en las reuniones.

### Administrador de desarrollo.

1. Producir un producto de alta calidad.
2. Usar plenamente las destrezas y habilidades de cada miembro del equipo.

### Administrador de planeación.

3. Producir un plan completo, preciso y detallado para cada miembro del equipo y para todo el equipo.
4. Proporcionar los reportes del estado semanal del equipo.

### Administrador de calidad y proceso.

1. Asegurar que el equipo siga fielmente TSPi y producir un producto de calidad.
2. Asegurar que todas las metas del equipo sean bien realizadas y reportadas.
3. Asegurar que todas las metas del equipo son reportadas con exactitud y que los reportes son colocados en la carpeta del proyecto.

### Administrador de configuración.

4. Asegurar que el equipo tiene disponibles herramientas y métodos que apoyen su trabajo.
5. No realizar cambios sin autorización en la línea base.
6. Todos los riegos y asuntos del equipo son registrados en el sistema de seguimiento.
7. Asegurar que el equipo conoce los objetivos de reciclaje para el ciclo de desarrollo.

## OBJETIVOS DEL PRODUCTO

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2 página, 9; "2.5 Objetivos del producto").

### **Antecedentes.**

Actualmente existe una base de datos diseñada en Visual Fox Pro 6.0, el cual realiza un procedimiento donde reúne información de diferentes Diplomados que se imparten a través de instancias de CONACULTA (Consejo Nacional para la Cultura y las Artes) en el interior de la República. Ésta información es capturada y procesada por Alta de Cartelera, Altas de instructores (quienes imparten los cursos), Altas de promotores (quienes toman los cursos) y calificaciones a los instructores, lo cual resulta insuficiente pues es mono-usuario y se desea hacer búsquedas por vía Internet.

### **Objetivos del proyecto.**

Desarrollar un sistema que sea capaz de administrar toda la información de "SRNCC" (Sistema de Red de Capacitación Cultural), el cual consistirá en los siguientes puntos:

- Proporcionar seguridad de acceso (módulo administrativo).
- Administrar el acceso a usuario.
- Implementación de base de datos.
- Generar reportes de catálogos y estadísticas de consultantes.
- Administración de captura de la información.
- Módulo de acceso de consulta.
- Exportación de datos.

## ESTÁNDARES PARA ENTREGA DE DOCUMENTACIÓN

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2, página 9; "2.6. Estándares para entrega de documentación").

- Uso de Microsoft Word 97 o superior.
- El tipo de letra será Ariadle tamaño 12 para encabezados de subtítulos con negrita, tamaño 11 para texto normal, para título principal de tamaño 14 con negrita y mayúsculas.
- Interlineado sencillo.
- Tabuladores a 1.25cm.
- Márgenes superior e inferior a 2.5cm del borde.
- Márgenes izquierdo y derecho a 3cm del borde.
- El pie de página irá a 1.5cm del borde.
- El pie de página contendrá el número de página en el extremo derecho y en el extremo izquierdo una referencia al archivo que contiene al documento adjuntando su extensión, todos estos datos estarán colocados debajo de una línea continua desde el margen izquierdo al derecho.
- Todo documento deberá tener alineación de justificado.
- Impresión con tinta negra con hoja de tamaño carta.
- Espacio entre párrafos de 2 interlineado.
- Formato para fechas será el siguiente **DD/MM/AA** donde **DD** es el día, **MM** es el mes, y **AA** corresponde al año.
- Las viñetas que se utilizarán serán cuadrados rellenas.
- Los números que se utilicen deberán ser seguidos por punto, espacio e información.
- La entrega de las formas es de la siguiente manera:
- El título de cada forma deberá ir centrado.
- La cabecera del documento contendrá logotipo con el nombre del equipo, responsable, fecha, ciclo, todo en español. En cuanto a la forma que pertenezca a TSPi, se realizará la traducción de la documentación original en idioma español para uniformar toda la documentación.
- El documento será identificado de la siguiente manera **FF-cn-eee** donde **FFF** será la fase en la que se está trabajando, **TTTTT** será una abreviación del tema principal del documento, **c** será un caracter fijo que significa ciclo y **n** al número del ciclo, **eee** es la extensión o tipo de archivo.

## MINUTA DE REUNIÓN

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 2, página 9; "2.7. Reuniones del equipo y la primera reunión").

En la reunión se llevó a cabo por ser la primera vez de manera extraordinaria en la cafetería de la Facultad de Ciencias. Aquí nos reunimos el Sr. Luis Antonio Hernández Nieto y el Sr. Leif Oswaldo Alduncin Gil, quienes seremos los participantes del proyecto a realizar.

Se habló de manera general sobre el proyecto, haciendo y dibujando algunos esbozos sobre el proyecto, también se habló que no se empezaría de inmediato, pues, faltan revisar con la Institución CONACULTA (Consejo Nacional para la Cultura y las Artes), lo que será la forma de pago y poder hacer un análisis detallado del sistema que se encuentra ahora funcionando.

Las siguientes reuniones se darán en las instalaciones de CONACULTA en la oficina de Sistematización, piso 9, éstas serán una vez por semana los viernes y la duración será de una hora aproximadamente.

Se asignaron los roles:

Líder del Equipo: Leif Alduncin Gil  
Administrador de Desarrollo: Leif Alduncin Gil.  
Administrador de Planeación: Luis Hernandez Nieto  
Administrador de Calidad y Proceso: Luis Hernandez Nieto  
Administrador de Configuración: Leif Alduncin Gil

# SISTCAP

---

## CAPITULO 11

### FASE DE ESTRATEGIA

#### Productos

- **Estrategia de Desarrollo.**
- **Gráfica de Dependencias.**
- **Forma ESTRA**
- **Forma RR.**
- **Plan de Configuración.**

## **ESTRATEGIA DE DESARROLLO**

El sistema se desarrollará en su totalidad a los largo de dos ciclos. En el primer ciclo el sistema contará con las siguientes funcionalidades.

- El administrador podrá dar de alta, eliminar y consultar un instructor en la base de datos.
- El administrador podrá dar de alta, eliminar y consultar a un coordinador en la base de datos.
- El administrador podrá dar de alta, eliminar una sesión en la base de datos.
- El administrador podrá dar de alta, eliminar, consultar una cartelera en la base de datos.
- El administrador podrá dar de alta, eliminar, y consultar un promotor en la base de datos.
- El administrador será el único que puede dar de alta, actualizar y eliminar una contraseña y un usuario.
- El administrador y el usuario se autenticarán en el sistema.
- El número de usuarios que podrán entrar al sistema es grande.

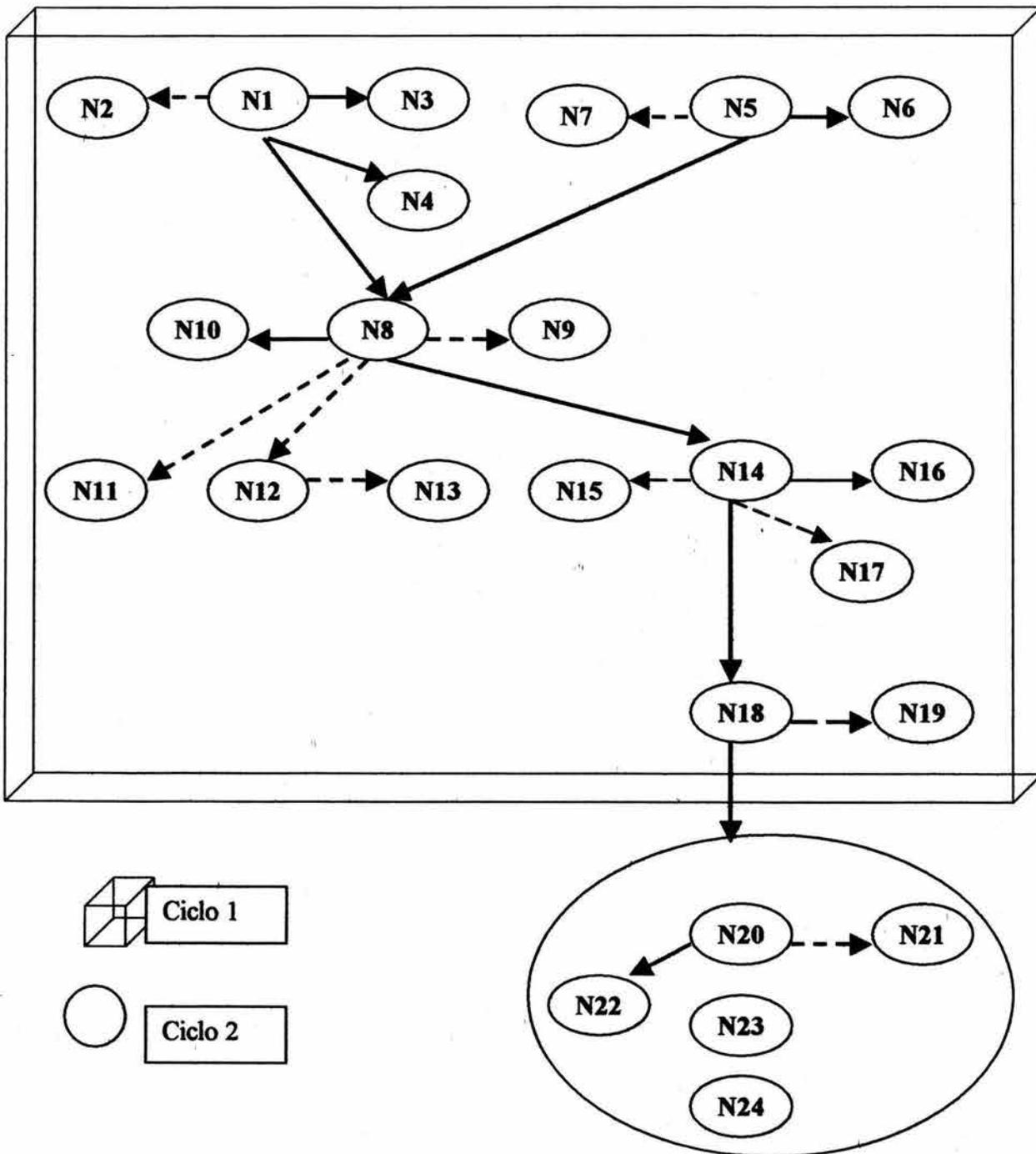
Existe la posibilidad de que algún componente requiera de que otro componente este terminado para poder seguir la fase de desarrollo.

El desarrollo se repartirá entre los integrantes del equipo los cuales entre todos se realizarán las pruebas de los mismos.

## GRÁFICA DE DEPENDENCIAS ENTRE NECESIDADES

- N1.** Alta de instructor: se podrá dar de alta un instructor para sus datos personales.
- N2.** Modificar instructor: se podrán modificar los datos personales de un instructor.
- N3.** Eliminar instructor: se podrá eliminar un instructor en caso de baja definitiva del sistema.
- N4.** Consulta de instructor: se podrán consultar diferentes datos del instructor.
- N5.** Alta de coordinador: se podrá dar de alta el coordinador para sus datos personales.
- N6.** Eliminar coordinador: se podrá eliminar un coordinador si es necesario.
- N7.** Modificar coordinador: se podrá modificar los datos del coordinador si es necesario.
- N8.** Alta de cartelera: se podrá dar de alta una cartelera para un diplomado, curso o taller.
- N9.** Modificar cartelera: se podrá modificar la cartelera para corregir cualquier información.
- N10.** Eliminar cartelera: se podrá eliminar la cartelera si es necesario.
- N11.** Consultas de cartelera: se podrá consultar cualquier cartelera del sistema.
- N12.** Alta de sesiones: se podrá dar de alta una sesión para varios diplomados, cursos o talleres.
- N13.** Modificar sesiones: se podrá modificar una sesión en caso de correcciones.
- N14.** Alta de promotor: se podrá dar de alta el promotor para sus datos personales.
- N15.** Modificar promotor: se podrán modificar los datos personales de un promotor.
- N16.** Eliminar promotor: se podría eliminar al promotor de baja definitiva del sistema.
- N17.** Consulta de promotor: se podrán consultar diferentes datos del promotor.
- N18.** Asignar curso al promotor: se podrá asignar curso al promotor para diplomados, etc.
- N19.** Desasignar curso al promotor: se podrá desasignar curso al promotor si es necesario.
- N20.** Certificar promotor: el promotor que este dado de alta se podrá certificar.
- N21.** Eliminar certificación: se podrá eliminar una certificación a un promotor en caso de que sea necesario.
- N22.** Evaluar un instructor: se evaluara a un instructor que se encuentre en el sistema y tenga un diplomado, taller, etcétera.
- N23.** Generación de datos oficiales: se podrán generar diferentes documentos oficiales de la información capturada.
- N24.** Generación de datos de apoyo: se podrán generar diferentes documentos para apoyo en el departamento.

### GRÁFICA DE DEPENDENCIAS ENTRE NECESIDADES



**FORMATO ESTRA**

Nombre	<u>Leif Oswaldo Alduncin Gil</u>	Fecha	<u>24/03/03</u>
Equipo	<u>SISTCAP</u>	Instructor	<u>Mtra. Guadalupe Ibarguengoitia</u>
Parte	<u>Estrategia</u>	Ciclo	<u>1</u>

Referencia	Funcionalidades	Ciclo I [LOC]	Ciclo [h]
1	Ventana de sesión	180	3.0
1.1	Submódulo de instructores en JavaScript	70	1.16
1.1.2	Creación de submenú para instructores en HTML	25	0.41
1.1.3	Creación de interfaz de solicitud de curp de instructor en HTML	85	1.41
1.1.4	Validación del curp del instructor en JavaScript	70	1.16
1.1.5	Interfaz en HTML e inserción de alta de un instructor	380	6.33
1.2	Interfaz para selección de un instructor en HTML	50	0.59
1.2.1	Actualización de datos de un instructor (usa interfaz de 1.3.3 para modificar datos de instructor)	120	2.0
1.3	Interfaz (ir a 1.2) y eliminación de instructor	15	0.25
2	Creación de submenú para coordinadores en HTML	25	0.41
2.1	Creación de interfaz de solicitud de curp de coordinador en HTML	85	1.41
2.1.2	Validación del curp del coordinador en JavaScript	70	1.16
2.1.3	Interfaz en HTML e inserción de alta de un coordinador	300	5.0
2.2	Interfaz para selección de un coordinador en HTML	50	0.59
2.2.1	Actualización de datos de un coordinador (usa interfaz de 2.1.3 para modificar datos de coordinador)	200	3.33
2.3	Interfaz (ir a 2.2) y eliminación coordinador	15	0.25
3	Submódulo de carteleras en JavaScript	70	1.16
3.1	Creación de submenú para cartelera en HTML	40	1.16
3.1.2	Creación de interfaz de solicitud de curp de coordinador y validación.	85	1.41
3.1.3	Interfaz en HTML e inserción de alta de una cartelera	70	1.16
3.2	Interfaz para selección de una cartelera en HTML	350	6.23
3.2.1	Actualización de datos de una cartelera (usa interfaz de 3.2 para modificar datos de una cartelera)	70	1.16
3.3	Interfaz (ir a 3.2) y eliminación de una cartelera	30	0.30
3.4	Creación de submenú para cartelera (esta interfaz es la misma del 3.1, por lo que no se cuenta como tiempo extra ni código extra, tan solo es referencia para esta sección, ya que es la misma)	X	X
3.4	Creación de interfaz en HTML para selección de cartelera	350	6.23
3.4.1	Interfaz de alta sesión e inserción de sesión	50	0.59
3.5	Actualización o modificación de una sesión (ir a 3.4.1)	150	2.50
4	Creación de submenú para promotores en HTML	25	0.41
4.1	Creación de interfaz de solicitud de curp del promotor en HTML	85	1.41
4.1.2	Validación del curp del promotor en JavaScript	70	1.16
4.1.3	Interfaz en HTML e inserción de alta de un promotor	520	9.06
4.2	Interfaz para selección de un promotor en HTML	50	0.59
4.2.1	Actualización de datos de un promotor (usa interfaz de 4.1.3 para modificar datos de promotor)	730	121.33
4.3	Interfaz (ir a 4.2) y eliminación de promotor	60	1.0

Referencia	Funcionalidades	Ciclo I [LOC]	Ciclo [h]
4.4	Creación de submenú para asignar curso en HTML (esta interfaz es la misma del 2.1, por lo que no se cuenta como tiempo extra ni código extra, tan solo es referencia para esta sección, ya que es la misma)	X	X
4.4	Creación de interfaz de solicitud de curp del promotor y de la cartelera en HTML	85	1.41
4.4.1	Validación de curp de la cartelera y del promotor	150	2.5
4.4.2	Asignación de curso	150	2.5
4.5	Copiar cartelera	100	1.06
5.	Bienvenida diseñada y desarrollada en HTML	155	2.58
5.1	Alta, actualización y eliminación de tipo de usuario	190	3.16
5.2	Validación de usuario en JavaScript	40	1.6
<b>TOTALES</b>		<b>5165</b>	<b>199.07</b>

**Comentarios:**

Las funciones descritas solamente corresponden al ciclo 1.

Las funciones principales (designadas con un solo número) corresponden a las interfaz gráficas consideradas principales para el sistema en el ciclo 1.

Los símbolos con una "X" significa que se ha ya considerado anteriormente, ver la columna de funcionalidades, ya que se especifica a que parte hace referencia el comentario.

**Forma Registro de Riesgos RR**Equipo SISTCAP Fecha: 24/03/2003  
Ciclo: 1 Instructor Guadalupe Ibangüergoitia

Fecha	24/03/2003
Riesgo o asunto	Posibles Cambios del Diseño
Descripción	Es posible que existan cambios en el la gráfica de Dependencias y en los requerimientos.
Gravedad B, M, A	M
Asignado a	Administrador de Configuración
Fecha de resolución	

Fecha	24/03/2003
Riesgo o asunto	Visualización en navegador
Descripción	En la forma en como se encuentra programado el sistema puede no verse en cualquier navegador.
Gravedad B, M, A	M
Asignado a	Administrador de Desarrollo
Fecha de resolución	

Fecha	24/03/2003
Riesgo o asunto	Versión del lenguaje de programación para el uso de algunas herramientas
Descripción	Es posible que la versión de el lenguaje de programación postgres no contenga el soporte para elaborar los formatos de impresión para los reportes.
Gravedad B, M, A	M
Asignado a	Administrador de Desarrollo
Fecha de resolución	

**Forma Registro de Riesgos RR**

Fecha	24/03/2003
Riesgo o asunto	Posible falta de equipo para el adecuado desarrollo del sistema
Descripción	Aún el departamento de el área no contiene mas que solo un equipo de computo que es dónde se captura la información, se espera la asignación de nuevos equipos para poder trabajar conforme al plan.
Gravedad B, M, A	A
Asignado a	Administrador de configuración
Fecha de resolución	27 de enero del 2003.

Fecha	24/03/2003
Riesgo o asunto	Posibles cambios retrasos por cambios constantes en las reuniones
Descripción	Las reuniones con el cliente se llegan a desplazar muy continuamente.
Gravedad B, M, A	A
Asignado a	Administrador de configuración
Fecha de resolución	27 de enero del 2003.

## PLAN DE CONFIGURACIÓN

### Mesa de Control de Cambios

- Leif Oswaldo Alduncin Gil ( Líder del Proyecto)
- Luis Hernández Nieto (Administrador de configuración)

### Productos a ser controlados

- Los requerimientos.
- Los productos del diseño.
- El código fuente de los programas.
- Los materiales de prueba y los resultados de las pruebas aplicadas.
- Productos estándares.
- Estándares para interfaz y pantallas.
- Biblioteca de código reusable.

### Procedimiento para la Solicitud de Cambios

Para solicitar un cambio se deberá llenar el Formato de Solicitud de Cambios, la cual constará del nombre del equipo, fecha, ciclo, tipo de cambio, características y razón del cambio y ésta será entregada al administrador de configuración para que solicite una reunión con los demás integrantes del equipo.

Cuando se solicite un cambio el administrador de configuración dará una copia del la solicitud de cambios al líder del equipo y se hablará de ésta en la reunión semanal.

Los cambios serán verificados por todo el equipo para llegar a un acuerdo, si se resuelve positiva el o los cambios el administrador de desarrollo realiza los cambios necesarios acerca del modulo que se requiera cambiar y el administrador de planeación verifica y hace cambios si son necesarios en la planeación del proyecto.

El administrador de configuración dará copia a todos los miembros del equipo acerca de los cambios aprobados.

### Sobre las reuniones del Equipo de Trabajo

El Administrador de configuración estará de forma permanente atendiendo los asuntos inmediatos del equipo. La reunión se solicitará primeramente con el líder del equipo y entre ambos tratarán de dar una solución a cada petición. En caso de que no converjan se convocará a reunión a los demás miembros del equipo para emitir una resolución final.

# SISTCAP

---

## CAPITULO 12

### FASE DE PLANEACIÓN

#### Productos

- **Planeación del proyecto.**
- **Diagrama de Gantt en equipo**
- **Forma SEMANA por equipo**
- **Forma SEMANA personal.**
- **Plan de Inspección**
- **Minuta de Reunión.**

## PLANEACIÓN DEL PROYECTO

La finalidad de éste documento es explicar a detalle el diagrama de Gantt que se generará para la planeación del proyecto en el Ciclo 1.

El proyecto estará planeado con la realización de diferentes tareas expuestas en varias etapas en el diagrama de Gantt, las cuales están propuestas de la siguiente manera:

- El proyecto se planea arrancar el 14 de enero del 2003.
- En la primera etapa del proyecto se planea:
  - Regular la base de datos existente.
  - Normalización de la información en papel.
  - Reporte de características del departamento.
  - Equipo adicional para la implementación del proyecto.

### Detalles primera etapa:

Se trata de regular la base de datos que se está trabajando en este momento en el departamento, al mismo tiempo regular la información en papel que se encuentra con un retraso de dos años, éstas tareas serán llevadas a cabo por el líder del equipo y el administrador de planeación. Se planea llevar un total de tres meses para la normalización de la documentación (ver fase de Lanzamiento, "Objetivos del Producto") la cual se comenzará el 14 de enero del 2003 y se planea concluir el 14 de Marzo aproximadamente. También se generarán requerimientos específicos para la apertura y captura de cursos nuevos que se planea que se utilizarán en la documentación del nuevo sistema.

Se solicitará un equipo adicional para el desarrollo del proyecto, ya que el departamento sólo cuenta con un equipo y no se puede ocupar pues es donde ahora se utiliza la base de datos para los cursos que se están realizando.

- La segunda etapa del proyecto se planea:
  - Documentos modificados y listos para la apertura de cursos.
  - Diseño de la base de datos.
  - Implementación de la base de datos.
  - Pruebas desde postgress a la base de datos.

### Detalles segunda etapa:

Los documentos se empezarán a modificar a partir del 10 de Marzo del 2003 para la apertura e inscripción de los cursos, ya que de esto dependerá las interfaces y la base de datos. El diseño de la base de datos la cual se empezará a realizar a partir del 17 de Marzo del 2003, ésta tarea se realizará el líder del equipo y el administrador de configuración y se planea una duración de 5 días de manera que sea lo más coherente posible con la existente, para evitar problemas futuros.

## PLANEACIÓN DEL PROYECTO

La elaboración de la base de datos, que tiene como características la administración y salvaguarda de la información de los cursos(carteleras y sesiones), los datos del promotor, del instructor, del coordinador y además se necesitará la realización de aproximadamente 20 catálogos. La implementación de la base de datos se planea iniciar el 24 de Marzo del 2003 y concluirá del 7 al 11 Abril del 2003 dando un total de 15 a 20 días. Se realizarán pruebas a la base de datos desde el manejador de bases de datos postgres, éstas se realizarán en la última semana del diseño, que puede ser ya sea en la última semana de la implementación del 7 al 11 de Abril del 2003.

- La tercera etapa del proyecto se planea:
  - Diseño de prototipo.
  - Creación de interfaces para la captura de la información.
  - Diseño de submenús para las interfaces.
  - Creación de submenús

### Detalles tercera etapa:

El diseño de las interfaces se iniciará a partir del 14 de Abril del 2003 hasta el 18 de Abril del 2003 y estará a cargo del administrador de desarrollo y administrador de planeación, la cual estará basada en los formatos corregidos. La implementación de las interfaces se iniciará a partir del diseño concluido y es del 21 de Abril hasta el 5 de Mayo del 2003. El diseño del submenús será desde el 28 de Abril al 2 de Mayo del 2003. La implementación de éstos se llevará acabo a partir del 5 al 12 de Mayo del 2003.

- La cuarta etapa del proyecto se planea:
  - Diseño de bienvenida y sesión.
  - Creación de ventana sesión y ventana bienvenida.

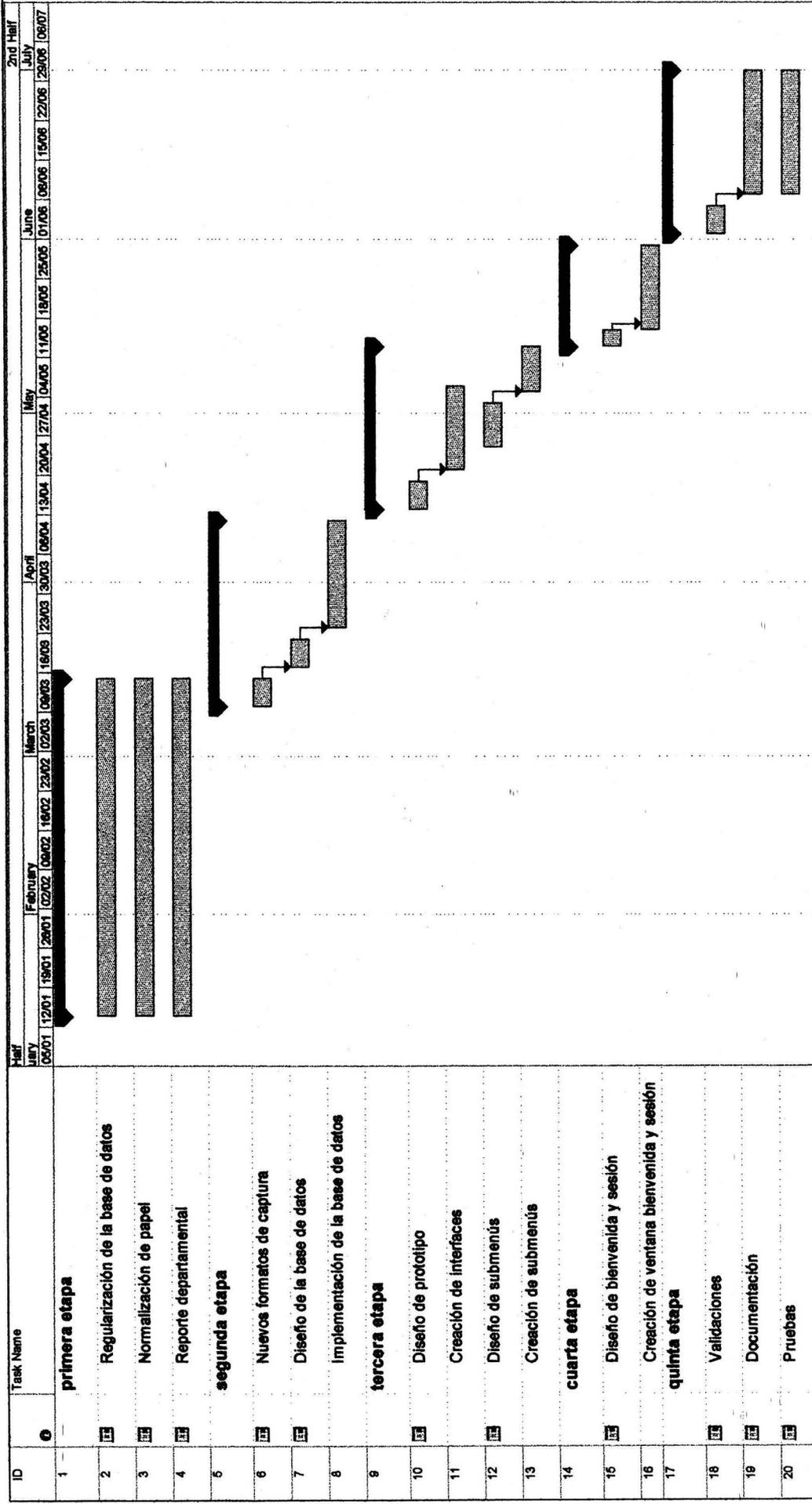
### Detalles cuarta etapa:

El diseño de la página de bienvenida y ventana sesión será llevada acabo a partir del 13 al 15 de Mayo y estará a cargo del administrador de planeación y de administrador de desarrollo. Del 16 al 30 de mayo del 2003 se crearán los prototipos de la ventana sesión y bienvenida a carga del administrador de planeación y desarrollo.

## PLANEACIÓN DEL PROYECTO

- La quinta etapa del proyecto se planea:
  - Validaciones.
  - Documentación.
  - Pruebas

Detalles quinta etapa: los métodos de validaciones serán llevados a cabo por el administrador de desarrollo y el administrador de planeación a partir del 2 al 6 de Junio del 2003, y la documentación correspondiente será llevada a cabo a partir del 9 de Junio del 2003, correspondiente a la finalización a los etapas anteriores se iniciaran las pruebas correspondientes a TSPi.



Project: Project2  
Date: Fri 14/06/02

Task  
Split

Progress  
Milestone

Summary  
Project Summary

External Tasks  
External Milestone

Deadline

**FORMA SEMANA PERSONAL**Nombre Leif Oswaldo Alduncin Gil Equipo SISCAP  
Fecha 01/04/03 Ciclo 1 Semana # 3

<b>Tareas de desarrollo efectuadas</b>	<b>Horas actuales</b>	<b>Estado</b>	<b>Tamaño</b>
Reunión de Equipo	1 hr	Completo	1 hoja
Instalación de Microsoft Project	30 min.	Completo	10.7 MB
Diagrama de Gantt	2 hr	Completo	3 hojas
Documentación	1 hr	Completo	4 hojas
Lectura del capítulo 5	2 hrs.	Completo	43 hojas
Llenar forma SEMANA	10 min.	Completo	
<b>Totales</b>	<b>6 hrs. 40 min.</b>	<b>Completo</b>	

**FORMA SEMANA PERSONAL**Nombre Luís Antonio Hernández Nieto Equipo SISCAP  
Fecha 01/04/03 Ciclo 1 Semana # 3

<b>Tareas de desarrollo efectuadas</b>	<b>Horas actuales</b>	<b>Estado</b>	<b>Tamaño</b>
Búsqueda de Microsoft Project	30 min.	Completo	
Instalación de Microsoft Project	30 min.	Completo	10.7 MB
Diagrama de Gantt	2 hr	Completo	3 hojas
Documentación	1 hr	Completo	4 hojas
Lectura del capítulo 5	2 hrs. y 30 min.	Completo	43 hojas
Llenar la forma Personal	10 min.	Completo	1 hoja
<b>Totales</b>	<b>5 hr 40 min.</b>	<b>Completo</b>	

**FORMA SEMANA PARA EL EQUIPO**

Equipo SISTCAP Semana 3  
 Fecha 01/04/03 Ciclo 1

Datos semanales		Actual		
Horas del proyecto para esta semana		12 hrs. 20 min.		
Horas del proyecto de este ciclo a la fecha		12 hrs. 20 min.		
Horas totales para las tareas terminadas en esta fase a la fecha		15 hrs. 50 min.		

Tareas de desarrollo efectuadas	Horas actuales	Tamaño a la fecha	Estado actual	Semana planeada
Asignación de Actividades	4 hrs. 20 min.	1 lista de actividades	Fin	PLAN
Diagrama de Gantt	2 hrs.	3 diagramas	Fin	PLAN
Formas Semanales (Equipo/Personal)	1 hr.	3 formas	Fin	PLAN
Leer Capitulo 5	4 hrs. 30 min.	43 hojas	Fin	PLAN
Instalación de Microsoft Project	30 min.	10.7 MB	Fin	PLAN
Documentación	2 hrs.	4 hojas	Fin	PLAN
Reunión de equipo	1 hr.	1 hoja	Fin	PLAN
Búsqueda de Software	30 min.	1 CD de instalación	Fin	PLAN
<b>Totales</b>	<b>15 hrs. 50min.</b>			

## PLAN DE INSPECCIONES

Este documento está generado de acuerdo a los criterios y parámetros de TSPi (ver Capítulo 4 página 14; “4.5 Plan de inspecciones”).

- Revisiones e inspecciones en requerimientos no más de 2.0 páginas por hora cuyo texto tendrá interlineado sencillo.
- Revisiones e inspecciones en diseño de alto nivel no más de 5 páginas por hora de diseño.
- Revisiones e inspecciones en diseño detallado no más de 100 líneas por hora de pseudocódigo.
- Revisiones e inspecciones de código fuente no más de 200 LOC/hora

## MINUTA DE REUNIÓN

La reunión se llevó a cabo dentro de las oficinas de Capacitación Cultural en el 9º Piso a las 10:30am en la cual se llegaron a los siguientes acuerdos:

1. Se acordó que como el equipo es pequeño, se dividirá en partes iguales la corrección de los documentos anteriores.
2. En base a los conocimientos o el mejor desempeño de cada integrante, se repartieron las funcionalidades a desarrollar para el sistema.
3. Se aprobó el plan de inspecciones, que se realizara para el producto.
4. Se decidió utilizar para generar el diagrama de Gantt a Microsoft Project.

La hora de finalización fue a las 11:40am

# SISTCAP

---

## CAPITULO 13

### FASE DE REQUERIMIENTOS

#### Productos

- **Especificaciones de requerimientos de software**
- **Casos de uso general.**
- **Casos de uso a detalle.**
- **Prototipo.**
- **Plan de pruebas del sistema.**

## ESPECIFICACIÓN DE LOS REQUERIMIENTOS DE SOFTWARE (ERS)

### Tabla de Contenido.

#### **1. Introducción.**

- 1.1 Objetivo.
- 1.2 Definición general de Sistema.
- 1.3 Información del equipo.

#### **2. Requerimientos Funcionales**

- 2.1 Diagrama de Casos de Uso y Prototipo.

#### **3. Otros requerimientos**

- 3.1 Requerimientos de interfaz Externas.
  - 3.1.1 Hardware.
  - 3.1.2 Software.
- 3.2 Restricciones de Diseño e Implementación.
- 3.3 Requerimientos especiales.

#### **4. Plan de Pruebas del sistema.**

## 1. INTRODUCCIÓN

### 1.1 Objetivo.

Los requerimientos son una descripción de las características finales del producto. El objetivo principal de esta fase es identificar y documentar las necesidades del cliente y su reflejo en las características que tendrá el producto, de tal manera que sea legible tanto para el grupo de desarrollo como para el cliente. Se pondrá atención en su definición clara, así como en la detección de los posibles riesgos, a fin de controlar todos los aspectos relacionados con el proyecto, evitando así, posibles desviaciones de la planeación original.

### 1.2 Definición General del Problema.

**Objetivo:** Crear un sistema WEB capaz de administrar todos los diplomados, talleres, cursos, etc. del Departamento de Capacitación Cultural del Consejo Nacional para la Cultura y las Artes.

Se necesita crear un sistema de computo denominado Sistema de Red de Capacitación (SRNCC) que pueda administrar, mantener, y desarrollar documentos oficiales a través de un portal de Internet.

Requerimientos generales de SRNCC:

1. Módulo de Administración de Instructores.
2. Módulo de Administración de Coordinadores.
3. Módulo de Administración de Cartelera y sesiones.
4. Módulo de Administración de Promotores.
5. Módulo de Bienvenida.
6. Módulo de Certificación.
7. Módulo de Consultas.
8. Módulo de Reportes.

De acuerdo con lo que sugiere TSPi, el proyecto será desarrollado en ciclos incrementales. En el primer ciclo se elaborará el núcleo de funcionalidades básicas del sistema. Se establecerá, así mismo, una biblioteca para depositar las funciones y clases reusables, así como todas las constantes que se emplearán en el proyecto.

En el primer ciclo se construirán toda la interfaz de usuario, con las funcionalidades respectivas, utilizando un manejador de bases de datos que es postgres, abarcando los primeros 5 módulos.

En el segundo ciclo se abarcarán los Módulos 6, 7 y 8 con sus respectiva interfaz y funcionalidades.

### 1.3 Información del Equipo

El desarrollo de éstos módulos se repartirá entre los miembros del equipo de la siguiente forma:

<b>Módulo</b>	<b>Responsables</b>
Instructores	Leif y Luis
Coordinadores	Leif y Luis
Cartelera y Sesiones	Leif y Luis
Promotores	Leif y Luis
Bienvenida	Leif y Luis

Fecha de entrega (máxima): 06 de junio 2003

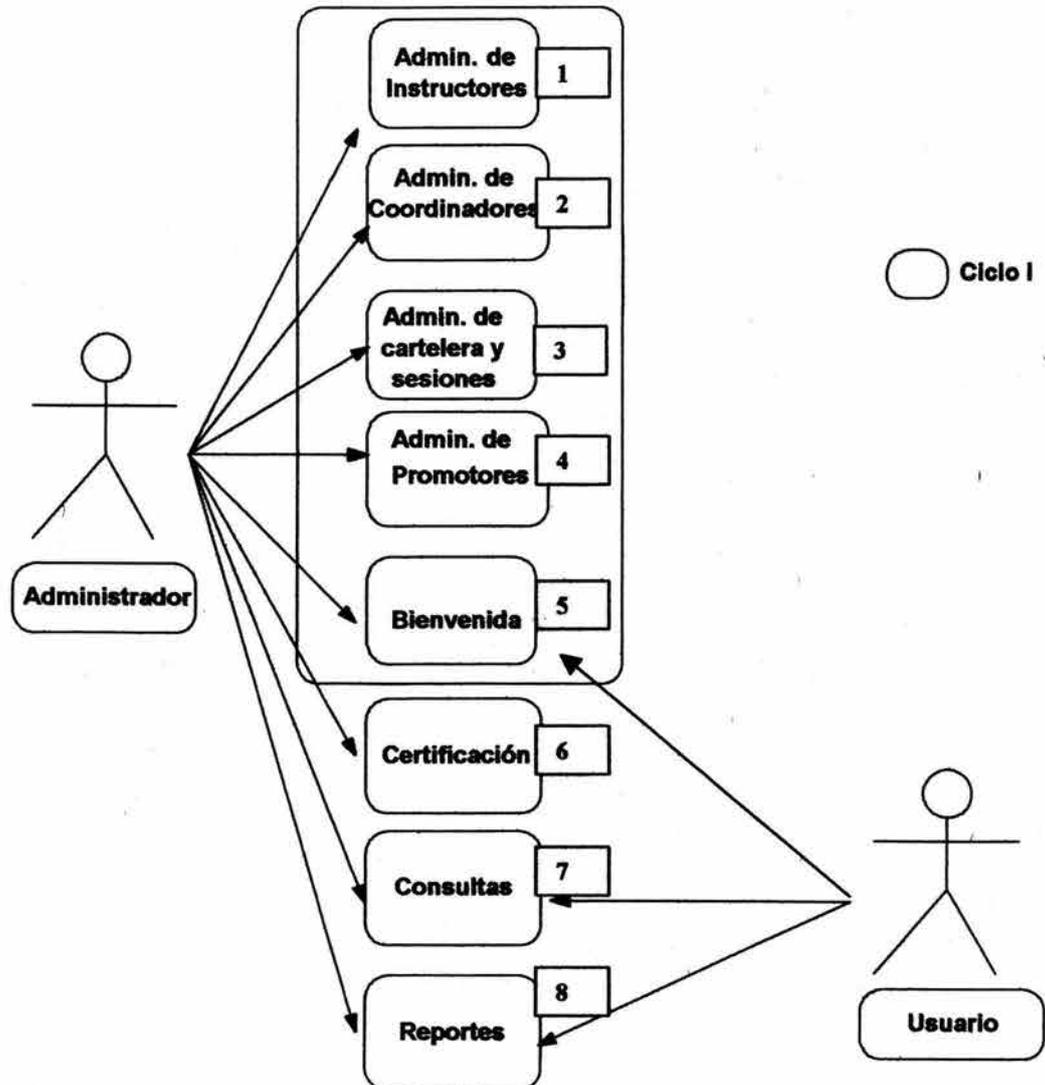
## 2. Requerimientos Funcionales

### 2.1 Casos de Uso y Prototipo.

Un caso de uso es un conjunto de escenarios con un objetivo común de un usuario. Entiéndase como escenario es una secuencia de pasos que describen una interacción entre un usuario (actor) y el sistema. Generalmente un caso de uso nos permite describir un escenario como una secuencia de pasos enumerados en las alternativas y variaciones. Los casos de uso son una herramienta esencial de los requerimientos, en el planeamiento y control de un proyecto interactivo. Los casos de uso nos permiten representar una visión externa del sistema y expresar los requerimientos funcionales de manera gráfica e indicando que función del sistema le sirve a que usuario.

### CASOS DE USO GENERAL

El objetivo de este documento es poder garantizar que se abarquen todos los casos uso que se requerirán en el sistema.

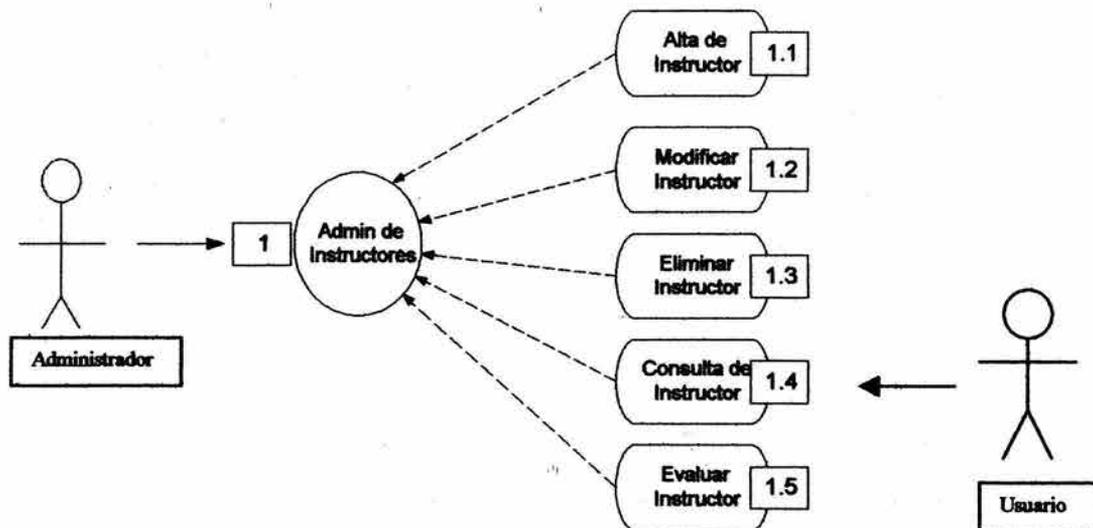


## CASOS DE USO

**Nombre:** 1. Administración de Instructores.<sup>1</sup>

**Objetivo:** Administrar la información de un instructor para darlo de alta, modificar, eliminar, consultar y evaluar en el sistema.

**Precondición:** Haber entrado al sistema como administrador.



<sup>1</sup> Los casos de uso restantes (2,3,4 y 5) se encuentran en el Anexo, página 152.

**1.1 Alta de Instructores**

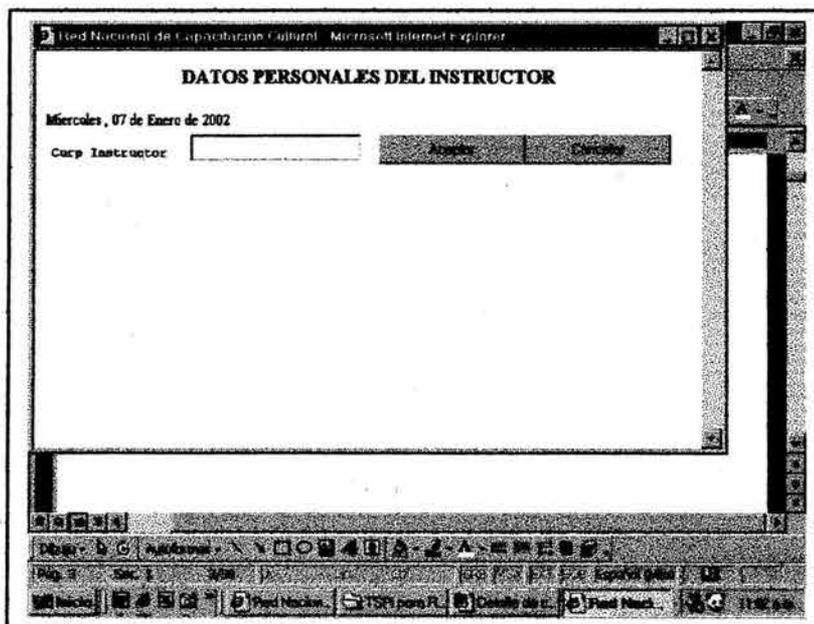
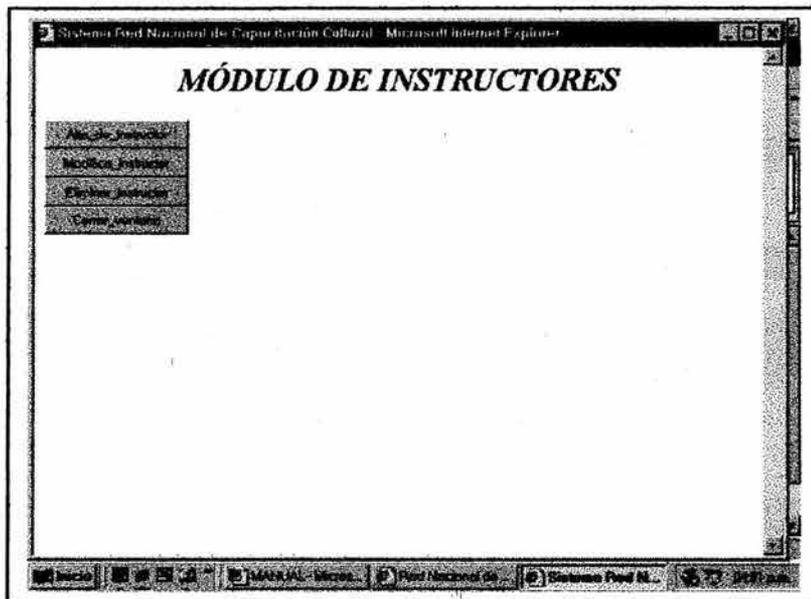
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Instructores" en la página de inicio.	2	El sistema abre la página "Módulo de Instructores"	
3	El administrador selecciona alta de instructor	4	El sistema abre otra ventana con título "DATOS PERSONALES DEL INSTRUCTOR" donde pide el Curp del Instructor. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador teclea el curp y le da click en el botón aceptar.	6	El sistema válida y revisa el curp	E 1.1

**Excepciones para 1.1**

Id	Descripción	Acción
E 1.1	El administrador no introduce curp y le da click al botón aceptar	El sistema muestra la leyenda "El curp no puede ser vacío, favor de corregirlo". Y un botón de regresar. Regresa a la página anterior

**Poscondición:** El sistema permite al administrador dar de alta al instructor.

**PROTOTIPO**

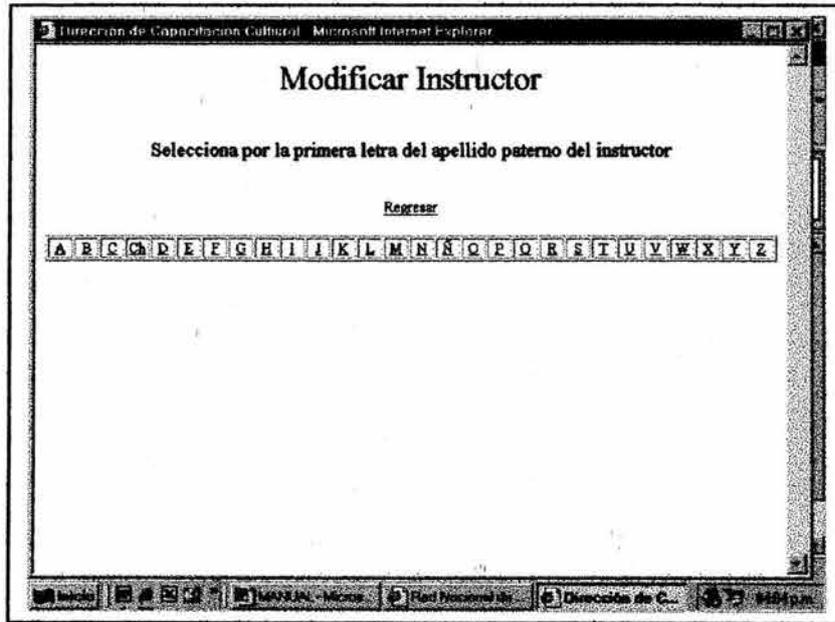


**1.2 Modificar Instructor**

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Instructores" en la página de inicio	2	El sistema abre la página "Módulo de Instructores"	
3	El administrador selecciona modifica_instructor	4	El sistema despliega una línea de letras de abecedario para buscar al instructor por la primera letra del apellido paterno. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al instructor y le da click en el botón modificar.	6	El sistema muestra los datos del Instructor	

**Poscondición:** El sistema permite al administrador modificar o consultar los datos del instructor seleccionado.

**PROTOTIPO**



**1.3 Eliminar Instructor**

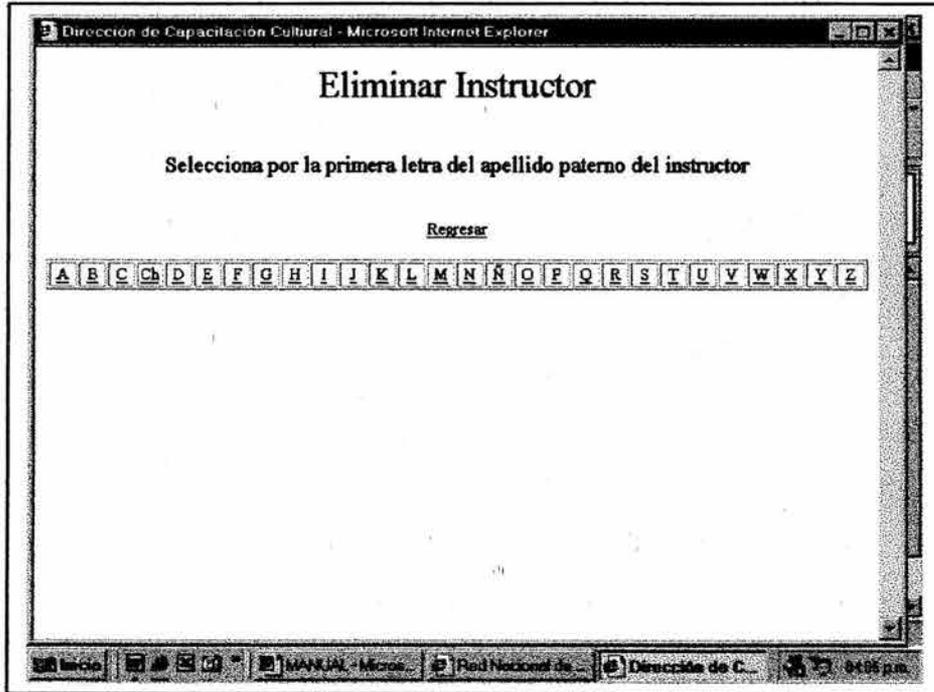
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Instructores" en la página de inicio.	2	El sistema abre la página "Módulo de Instructores"	
3	El administrador selecciona Eliminar_Instructor	4	El sistema despliega una línea de letras de abecedario para buscar al instructor por la primera letra del apellido paterno. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al instructor y le da click en el botón Eliminar.	6	El sistema confirma la eliminación.	E 1.2

**Excepción para 1.3**

Id	Descripción	Acción
E 1.2	El administrador decide eliminar al instructor o no.	El sistema muestra la pantalla de confirmación, si es afirmativa elimina al instructor, en caso contrario regresa al paso 3.

**Poscondición:** El sistema permite al administrador eliminar al instructor seleccionado.

**PROTOTIPO**



### 3. OTROS REQUERIMIENTOS

A continuación se muestran los principales requerimientos adicionales que deben cumplirse a fin de que el SRNCC (Sistema de Red de Capacitación) pueda funcionar en un ambiente de producción.

#### 3.1 Requerimientos de Interfaz Externas.

**3.1.1 Hardware.** El servidor debe tener la capacidad suficiente para procesar las peticiones que se hagan desde múltiples servidores que se conectarán con el primero vía Internet por medio del browser. Por esta razón se requiere una computadora con un procesador mayor a 200 MHz, con conexión a la red y 128 Mb de memoria RAM como mínimo.

Por otra parte, los clientes, no requieren de mayor capacidad de proceso que la requerida para mantener un browser de Internet funcionando.

**3.1.2 Software.** En el servidor se requiere un manejador de base de datos como (postgres), servidor de páginas http y JavaScript (se sugiere Apache), como sistema operativo se requiere Linux.

Los clientes únicamente requieren el browser de Internet, ya que únicamente recibirán páginas html. Sin embargo, se requiere que dicho browser tenga capacidad para soportar el código JavaScript a fin de ejecutar las validaciones de los datos que se enviarán al servidor.

#### 3.2 Restricciones de Diseño e Implementación

Las herramientas que se emplearán para el desarrollo de ésta aplicación serán las siguientes:

1. Rational Rose 200 Enterprise Edition. Para el análisis y diseño del sistema.
2. Flow4 para el diseño de la base de datos.
3. Algún procesador de texto para la documentación, puede ser Microsoft Word 2000 de Windows.

#### 3.3 Requerimientos especiales.

El SRNCC hará suyas las siguientes características:

**Portabilidad.** Ya que será implementado en PHP, la funcionalidad en el servidor podrá instalarse en cualquier en Linux para que soporte la interpretación de este lenguaje. Por otro lado, los clientes pueden tener una arquitectura física cualquiera, ya que únicamente se requiere que se posean un browser de Internet, así como una conexión a la red.

**Desempeño.** Debido a que todos los procesos se desarrollaran en el servidor, bastara con garantizar dicha capacidad en este para que el sistema completo se encuentre dentro de rango razonables de tiempo de respuesta.

**Usabilidad.** La interfaz de usuario se diseñaran de forma que el sistema resulte lo más intuitivo posible para el usuario.

**Mantenible.** Considerando que la construcción del sistema estará basada completamente en la documentación que se generará, se estima que cualquier modificación se podrá ejecutar sin complicaciones

---

## PLAN DE PRUEBAS

---

---

### 1.1 Alta Instructor<sup>2</sup>

---

Datos Prueba	Resultado esperado
Dar click en el botón instructores	Abrir modulo de instructores
Dar click en alta de instructores	Abrir modulo de alta de instructores
Introducir Curp de instructor para dar de alta	El sistema lo da de alta
Introducir Curp de instructor mal escrito	Mensaje de error de validación de Curp mal escrito
No introducir Curp de instructor	Mensaje de error de Curp vacío
Dar click en cancelar	El sistema regresa a la pantalla anterior

---

### 1.2 Modificar Instructor

---

Datos Prueba	Resultado esperado
Dar click en modificar instructor	Abrir modulo de modificar instructor
Dar click en modificar de instructor	No abrir modulo de modificar instructor
Seleccionar Curp de instructor	El sistema no tiene el Curp registrado
Seleccionar Curp de instructor	El sistema lo despliega en una lista
Dar click en cancelar	El sistema regresa a la pantalla anterior
Dar click en cancelar	El sistema no ejecuta ninguna acción

---

### 1.3 Eliminar Instructor

---

Datos Prueba	Resultado esperado
Dar click en eliminar instructor	Abrir modulo de eliminar instructor
Dar click en eliminar de instructor	No abrir modulo de eliminar instructor
Seleccionar Curp de instructor	El sistema no tiene el Curp registrado
Seleccionar Curp de instructor	El sistema lo despliega en una lista
Eliminar promotor seleccionado	Mensaje de verificación para eliminar
Dar click en cancelar	El sistema regresa a la pantalla anterior
Dar click en cancelar	El sistema no ejecuta ninguna acción

---

<sup>2</sup> El plan de pruebas de los módulos 2,3,4 y 5 se encuentran en el Anexo, página 152

**FORMA SEMANA PARA EL EQUIPO**

Equipo SISTCAP Semana 3  
Fecha 15/04/03 Ciclo 1

<b>Datos semanales</b>		<b>Actual</b>
Horas del proyecto para esta semana		12 hrs. 20 min.
Horas del proyecto de este ciclo a la fecha		12 hrs. 20 min.
Horas totales para las tareas terminadas en esta fase a la fecha		15 hrs. 50 min.

<b>Tareas de desarrollo efectuadas</b>	<b>Horas actuales</b>	<b>Tamaño a la fecha</b>	<b>Estado actual</b>	<b>Semana planeada</b>
Asignación de Actividades	4 hrs. 20 min.	1 lista de actividades	Fin	PLAN
Diagrama de Gantt	2 hrs.	3 diagramas	Fin	PLAN
Formas Semanales (Equipo/Personal)	1 hr.	3 formas	Fin	PLAN
Leer Capitulo 5	4 hrs. 30 min.	43 hojas	Fin	PLAN
Instalación de Microsoft Project	30 min.	10.7 MB	Fin	PLAN
Documentación	2 hrs.	4 hojas	Fin	PLAN
Reunión de equipo	1 hr.	1 hoja	Fin	PLAN
Búsqueda de Software	30 min.	1 CD de instalación	Fin	PLAN
<b>Totales</b>	<b>15 hrs. 50min.</b>			

**FORMA SEMANA PERSONAL**

Nombre Leif Oswaldo Alduncin Gil Equipo SISCAP  
Fecha 15/04/03 Ciclo 1 Semana # 3

<b>Tareas de desarrollo efectuadas</b>	<b>Horas actuales</b>	<b>Estado</b>	<b>Tamaño</b>
Reunión de Equipo	1 hr	Completo	1 hoja
Instalación de Microsoft Project	30 min.	Completo	10.7 MB
Diagrama de Gantt	2 hr	Completo	3 hojas
Documentación	1 hr	Completo	4 hojas
Lectura del capítulo 5	2 hrs.	Completo	43 hojas
Llenar forma SEMANA	10 min.	Completo	
<b>Totales</b>	<b>6 hrs. 40 min.</b>	<b>Completo</b>	

ESTA TESIS NO SALE  
DE LA BIBLIOTECA

**FORMA SEMANA PERSONAL**Nombre Luís Antonio Hernández Nieto Equipo SISCAP  
Fecha 15/04/03 Ciclo 1 Semana # 3

<b>Tareas de desarrollo efectuadas</b>	<b>Horas actuales</b>	<b>Estado</b>	<b>Tamaño</b>
Búsqueda de Microsoft Project	30 min.	Completo	
Instalación de Microsoft Project	30 min.	Completo	10.7 MB
Diagrama de Gantt	2 hr	Completo	3 hojas
Documentación	1 hr	Completo	4 hojas
Lectura del capítulo 5	2 hrs. y 30 min.	Completo	43 hojas
Llenar la forma Personal	10 min.	Completo	1 hoja
<b>Totales</b>	<b>5 hr 40 min.</b>	<b>Completo</b>	

**FORMA REGD**

Nombre: Luis Hernández Nieto Fecha: 15/04/03  
 Equipo: SISTCAP Ciclo: 1  
 Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Había un error en el casos de uso general, faltaba marcar lo correspondiente al ciclo1

Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Se imprimió mal la hoja de caso de cada caso de uso.

Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Había un error de escritura, pues en la excepción para 2.1 estaba mal el número.

Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Había un error en la excepción para 2.5 pues aparecía en la página para el punto 2.4

Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Había un error de escritura, pues en la excepción para 3.3 estaba mal el número.

Fecha: Número Tipo Inyectado Eliminado Tiempo de corrección Defecto corregido  
        
 Descripción: Había un error de escritura, pues en la excepción para 4.1 estaba mal el número.

## MINUTA DE SEMANAL

La reunión se llevó a cabo dentro de las oficinas de Capacitación Cultural en el 9° Piso a las 12:00pm en la cual se llegaron a los siguientes acuerdos:

1. Se generó la especificación de los requerimientos.
2. Se acordó el texto del planteamiento del problema para el sistema.
3. Se diseñó el casos de uso general del sistema.
4. Se aprobó el prototipo del sistema.
5. Se elaboró los casos de uso a detalle del sistema.
6. Se acordó el plan de pruebas del sistema.
7. Se llevaron a cabo las inspecciones a ERS y plan de pruebas.
8. Se generó el REGT.

La hora de finalización fue a las 13:00 hrs.

# SISTCAP

---

## CAPITULO 14

### FASE DE DISEÑO

#### Productos

- **Estándar de Diseño.**
- **Especificación de Diseño de Software (EDS).**
- **Plan de Pruebas de Integración.**
- **Forma REGD.**

## ESTÁNDARES DE DISEÑO

A continuación se describen las convenciones que se emplearán en el diseño de alto nivel.

### Convenciones de nombrado

**Clases.** El nombre de todas las clases empezarán con letra mayúscula. Si se tratará de un nombre compuesto por varias palabras, éstas deberán juntarse con guiones inferiores, y se iniciará cada palabra con una letra mayúscula. Se usará en las clases en nombre completo de la que se este refiriendo no importando que tan grande sea la palabra.

**Métodos.** Los nombres de los métodos iniciarán con una letra minúscula y los siguientes caracteres deberán ser en minúscula también, en el caso de que el método sea compuesto con otra palabra se deberá unir con uno o varios guiones bajos.

**Variables y parámetros.** Con respecto a las variables y parámetros siempre deberán ir en minúsculas, en caso de que llegase a existir un nombre compuesto deberá usar guión bajo.

**Archivos.** El nombre de los archivos que se generaran deberán ser en minúsculas, si el archivo contiene mas de una palabra podrá ser unida con un guión bajo. En el caso de ser un directorio se deberá escribir en mayúsculas.

### Formatos de Interfaz

Los formatos para las interfases serán de dos tipos:

1. Para todas las interfases será ".html"
2. Para todos las interfaces de error será utilizada con JavaScript ".js"

### Mensajes de error y del sistema.

Para los mensajes del sistema se utilizará en HTML o los del browser.

Los mensajes de error tendrán las siguientes características

TIPO	MENSAJE	CARACTERISTICAS
Error	El mensaje se localizará en la parte central de la ventana y a su vez éste deberá estar centrado	<ul style="list-style-type: none"><li>▪ Ventana de funcionalidad javaScript,</li><li>▪ El color de fondo será gris.</li><li>▪ El tamaño esta sujeto a la funcionalidad "alert" en javaScrpit.</li><li>▪ Tendrá un símbolo de admiración en el costado izquierdo con fondo amarillo.</li><li>▪ Contendrá un botón con la leyenda "Aceptar" o "Regresar" o "Inicio" ; el botón se encontrará situado en la parte inferior de la ventana y centrado.</li></ul>

**Prototipo de mensajes de error:****Estándares de diseño.**

En la especificación operacional, funcional, de estados y lógica se emplearán los diagramas de UML, componiéndose de diagramas de realización de clases y de secuencia.

**Estándares de documentación de reuso.**

En el caso de las funciones que son de reuso siguen las siguientes características:

**Métodos.** Los nombres de los métodos iniciarán con una letra minúscula y los siguientes caracteres deberán ser también en minúscula, en el caso de que el método sea compuesto con otra palabra se deberá unir con uno o varios guiones bajos.

**Variables y parámetros.** Con respecto a las variables y parámetros siempre deberán ir en minúsculas, en caso de que llegase a existir un nombre compuesto deberá usar guión bajo.

Las funciones o interfases que sean de reuso se agruparan en un directorio que tendrá por nombre REUSO con mayúsculas para la directa localización.

## ESPECIFICACIONES DE DISEÑO DE SOFTWARE (EDS)

### Tabla de Contenido.

- 1. Estructura del producto**  
Diseño Arquitectónico(3 capas)
  - 1.1. Capa de presentación.
  - 1.2. Capa lógica de la aplicación
    - 1.2.1. Diagrama de clases.
    - 1.2.2. Diagrama de dependencias
  - 1.3. Capa de almacenamiento
    - 1.3.1. Diagrama Entidad Relación.
- 2. Diagrama de Secuencia.**
- 3. Diagrama de Componentes.**
- 4. Plan de Pruebas de Integración.**

## 1.1 Diseño arquitectónico

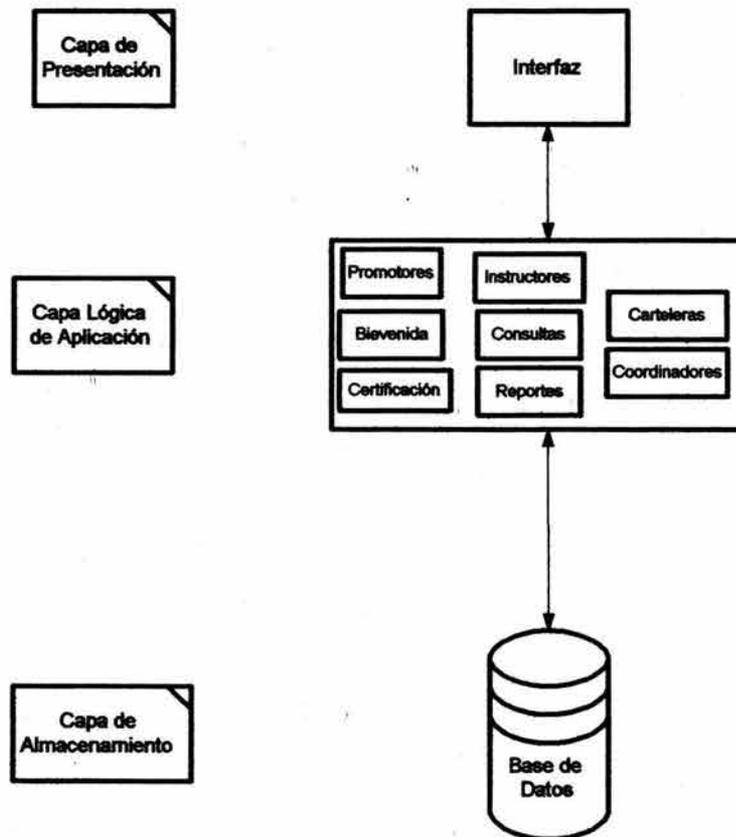
### Arquitectura (3capas)

Esta arquitectura se divide en tres partes:

**1.2 Capa de presentación:** reúne todos los aspectos del software que tiene que ver con las interfaces y la interacción con los diferentes tipos de usuarios. Estos aspectos típicamente incluyen el manejo y aspecto de las ventanas, el formato de los reportes, menús, gráficos y elementos multimedia en general.

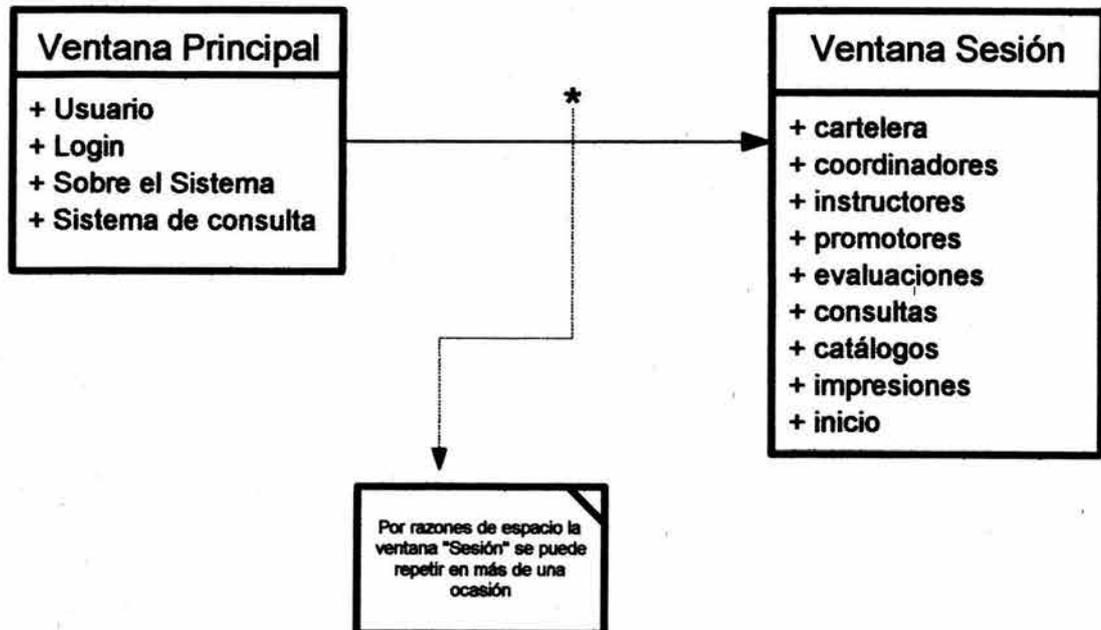
**1.3 Capa lógica:** reúne todos los aspectos del software que tienen que automatizan o apoyan los procesos que llevan a cabo los usuarios. Estos aspectos típicamente incluyen las tareas que forman parte de los procesos, las reglas y restricciones que aplican

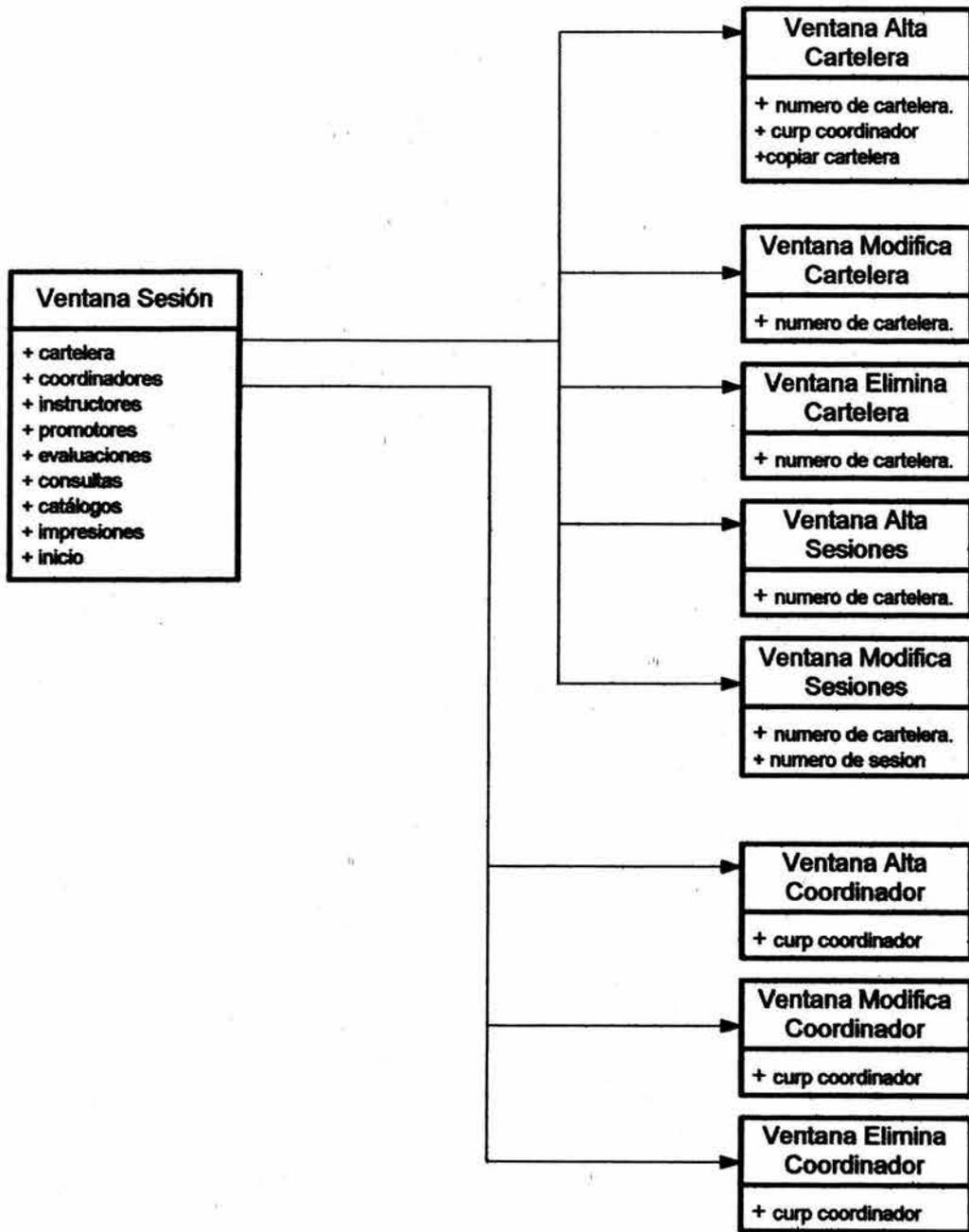
**1.4 Capa de almacenamiento:** reúne todos los aspectos del software que tienen que ver con el manejo de los datos persistentes

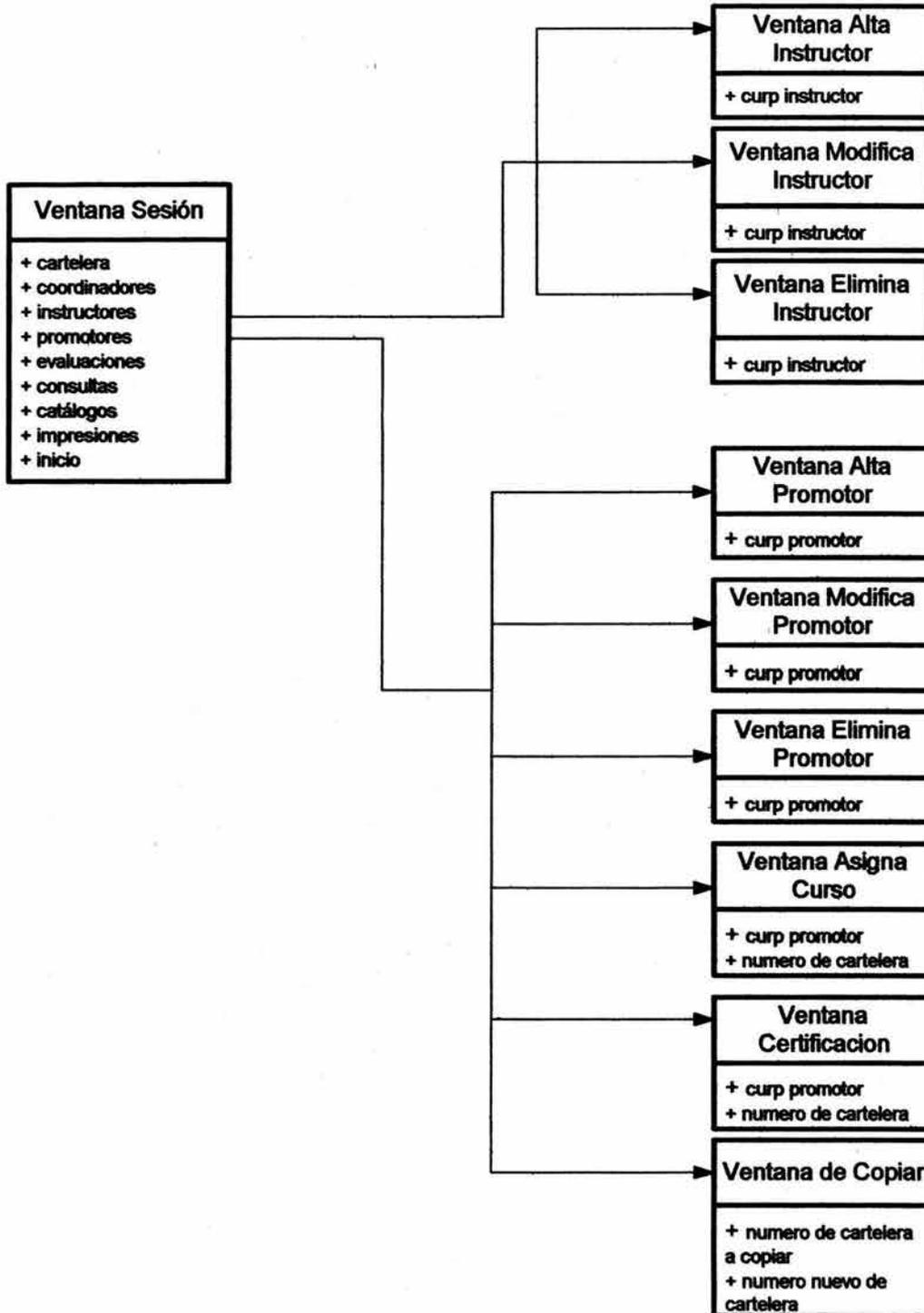


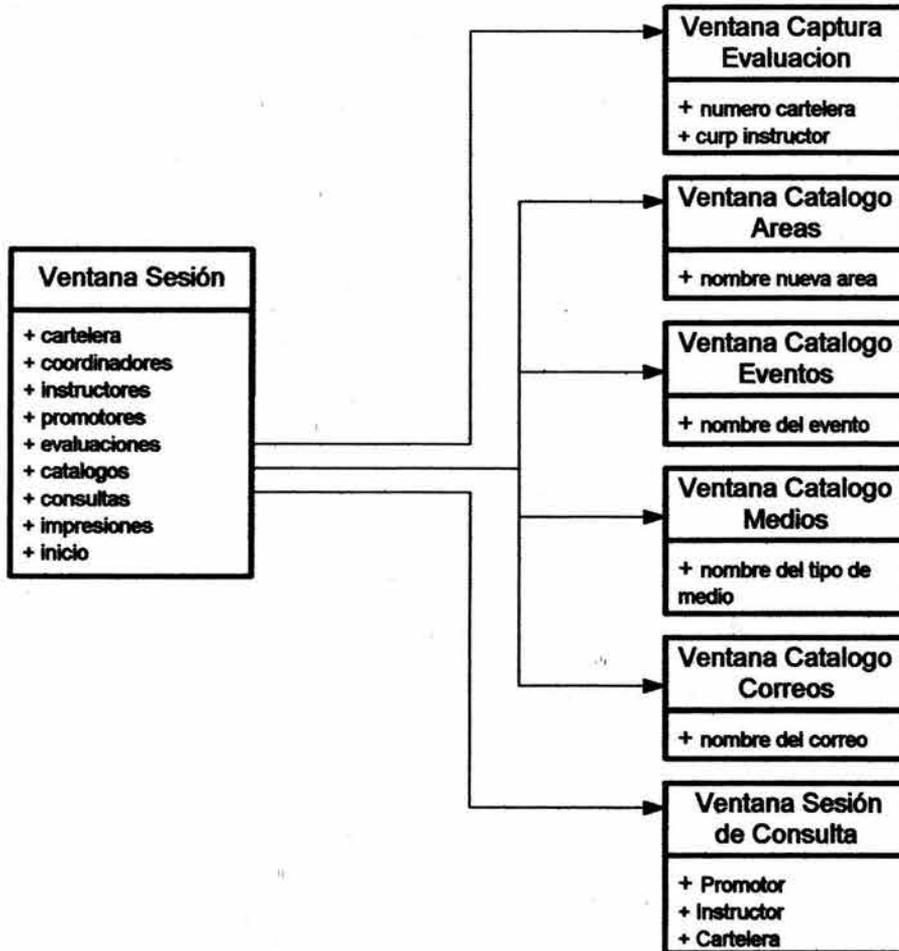
## 1.2 Capa de presentación.

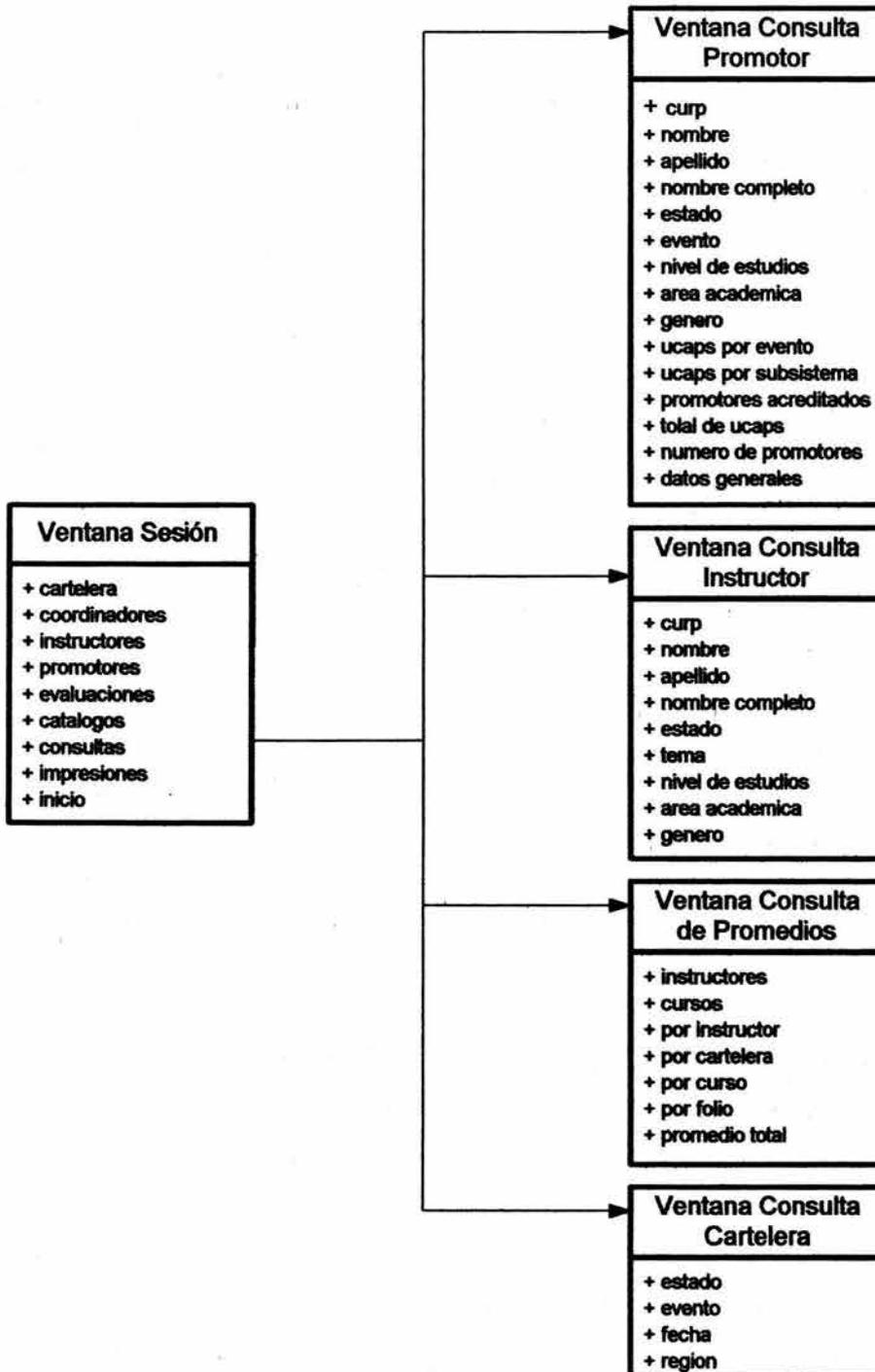
Este diagrama contiene el conjunto de interfaces que usará el usuario en el sistema; se parte a través de una ventana principal y las flechas nos indican las rutas a submenús u otras ventanas donde el usuario puede acceder.

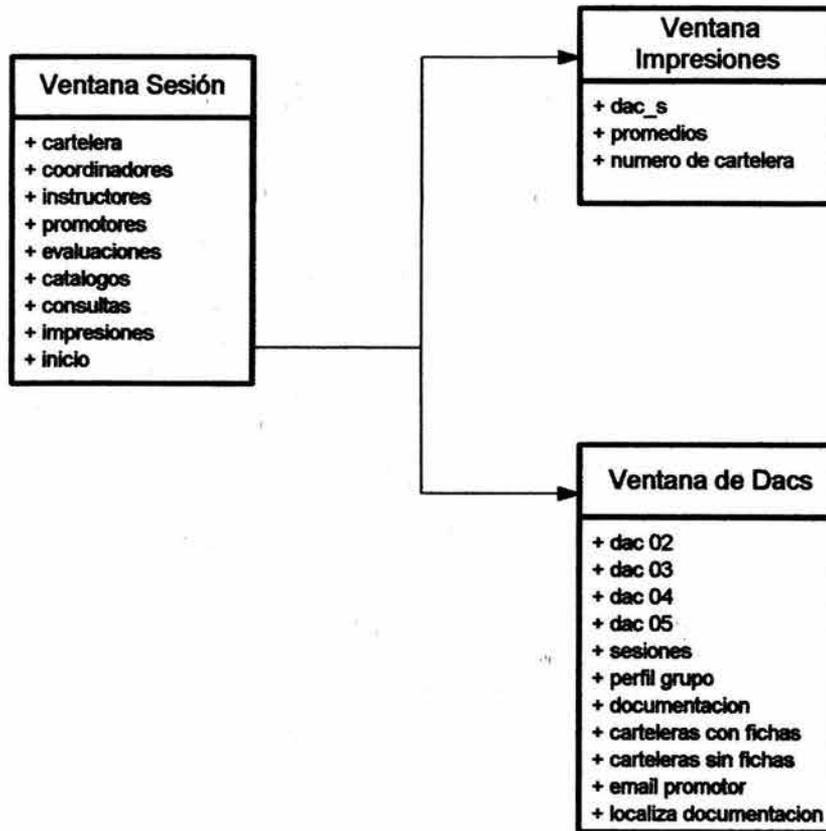






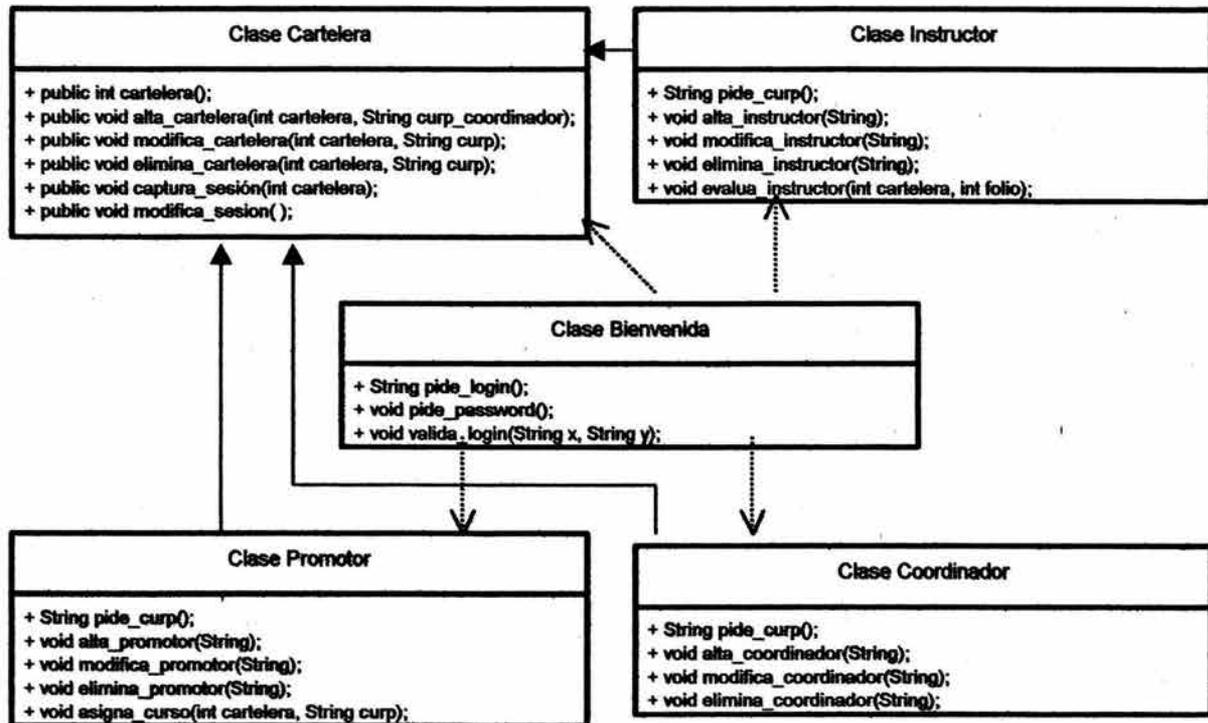




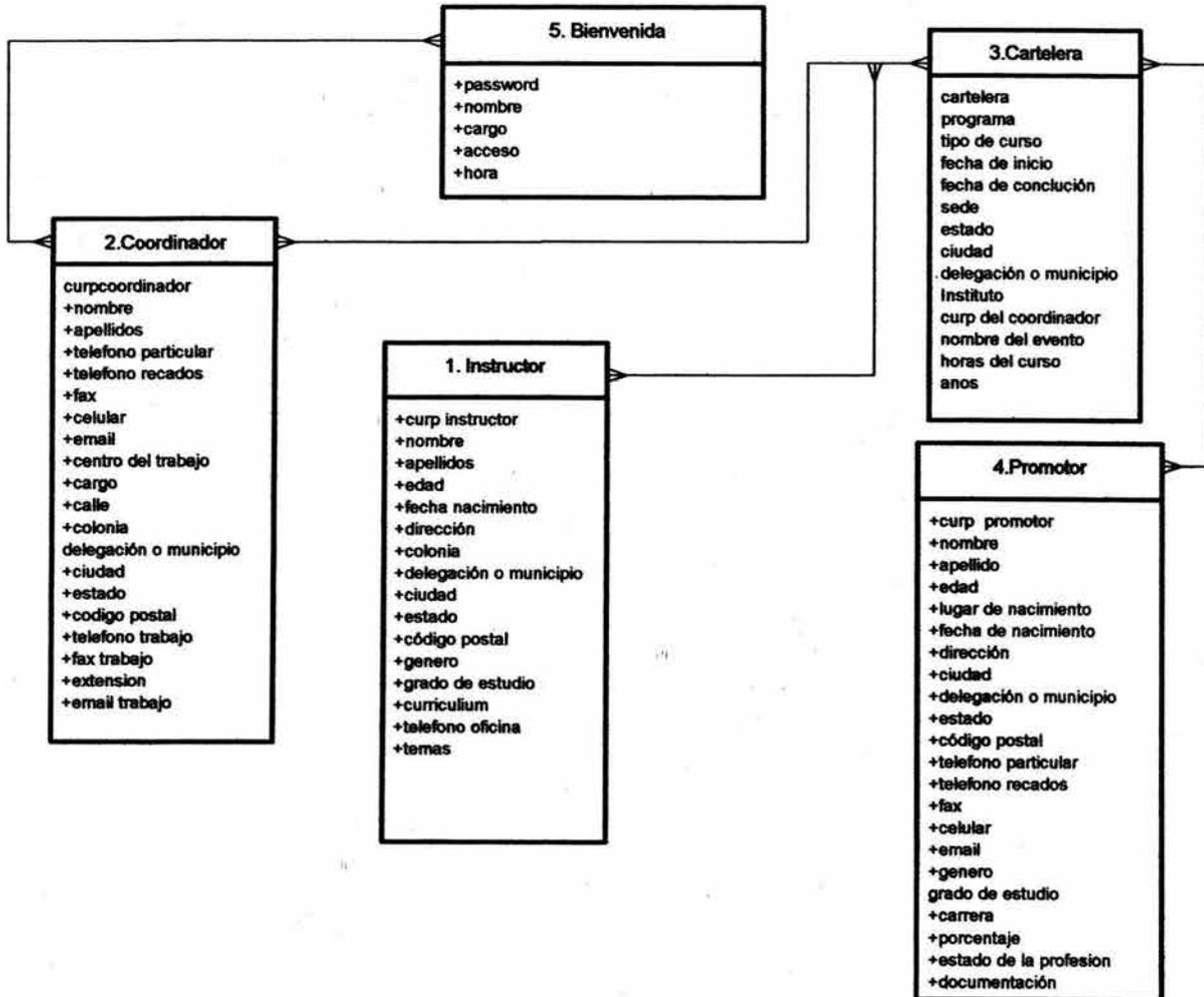


## 1.2 Capa lógica de presentación

### 1.2.1 Diagrama de Clases



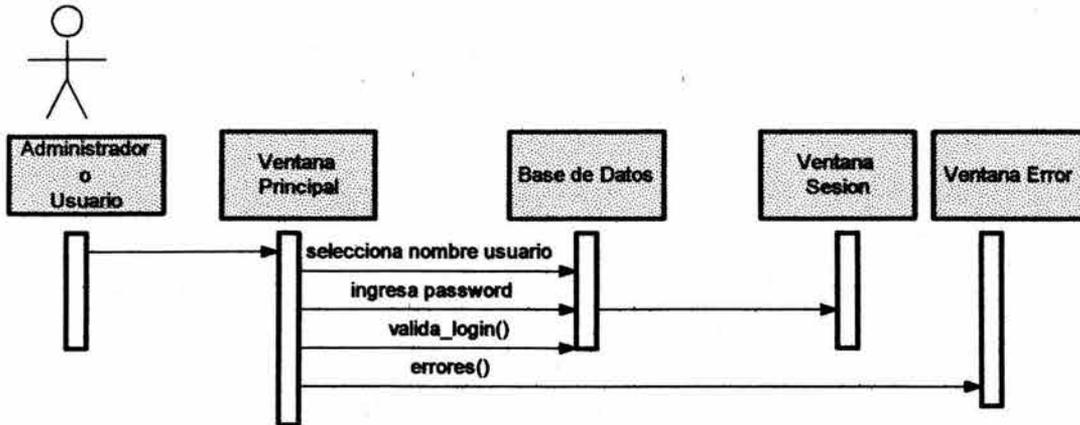
## 1.2.2 Diagrama de Dependencias



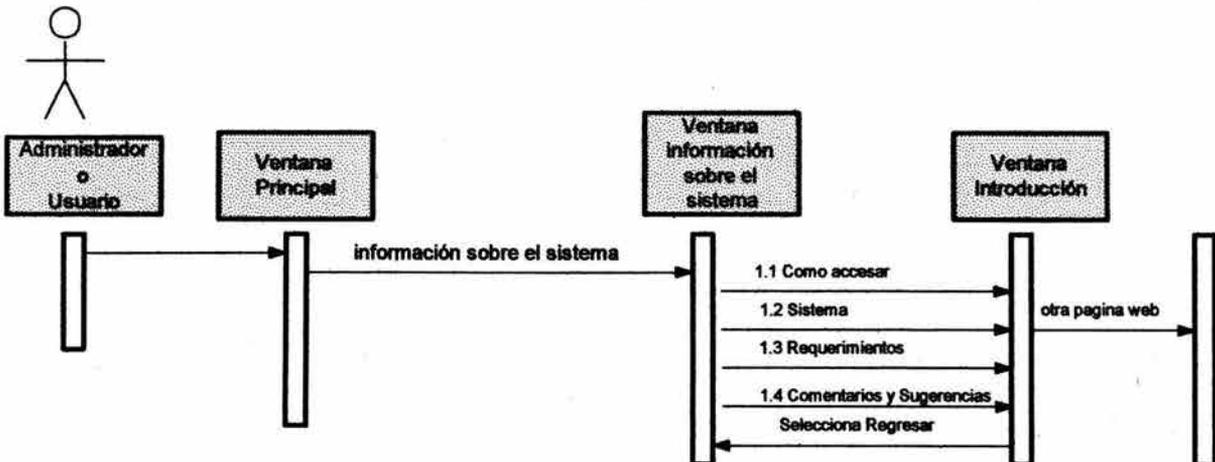


## 2. DIAGRAMA DE SECUENCIA

Caso de Uso: 5.1 Clave de Acceso  
Diagrama de Secuencia

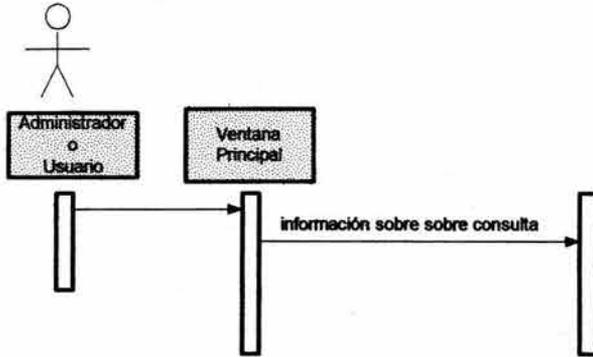


Caso de Uso: 5.2 Información sobre el sistema  
Diagrama de Secuencia



### Caso de Uso: 5.3 Información sobre Consulta

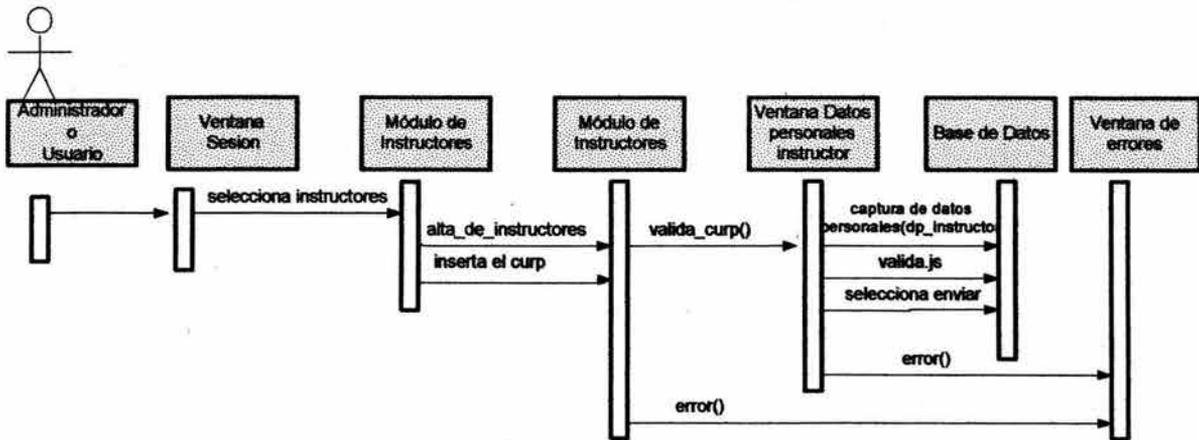
#### Diagrama de Secuencia



### Caso de Uso: 1.1 Alta de Instructores

#### Diagrama de Secuencia

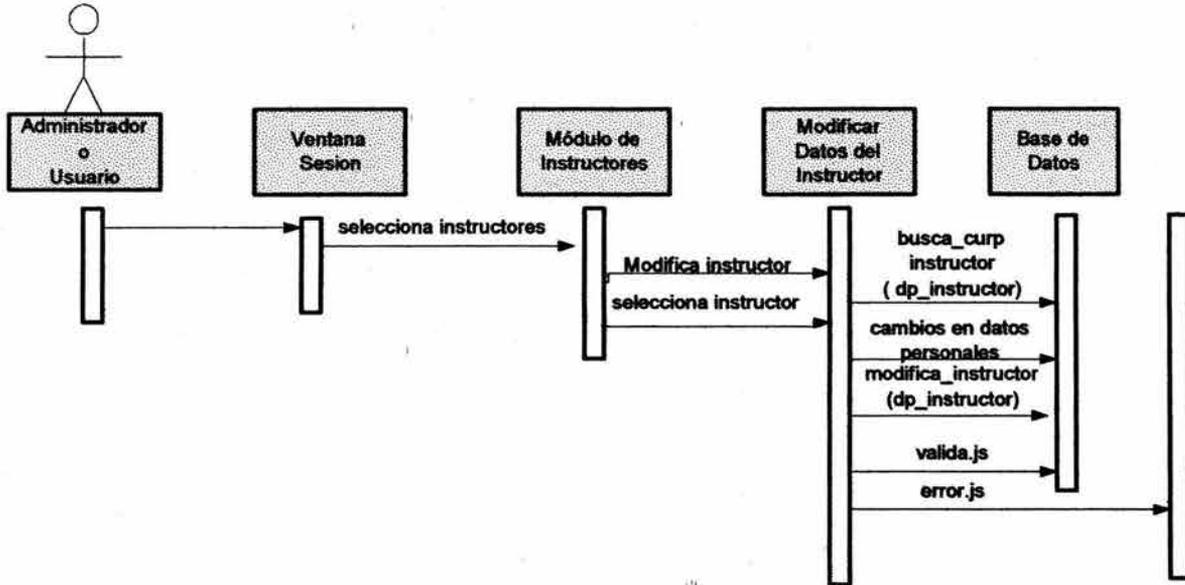
Precondición: Ver casos de uso: 5.1



### Caso de Uso: 1.2 Modificar Instructor

#### Diagrama de Secuencia

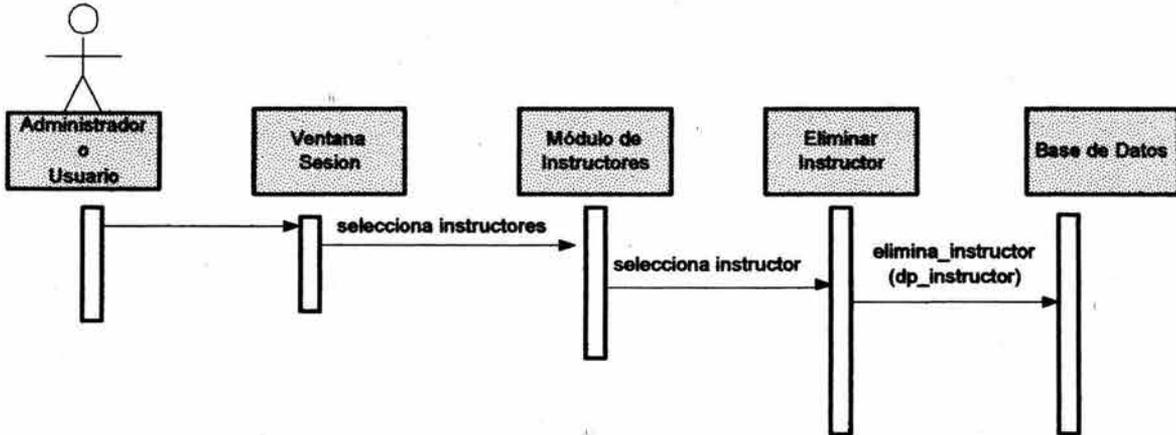
Precondición: Ver casos de uso: 5.1



### Caso de Uso: 1.3 Eliminar Instructor

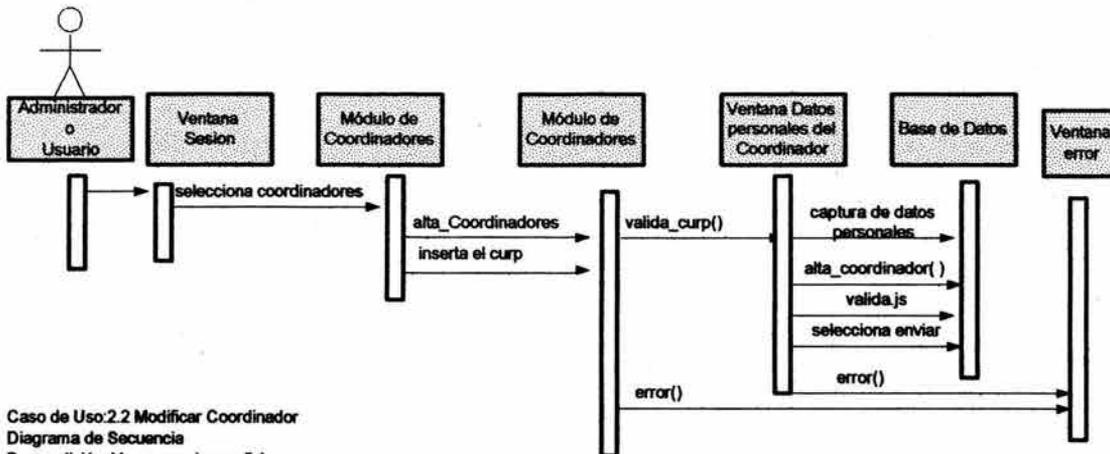
#### Diagrama de Secuencia

Precondición: Ver casos de uso: 5.1

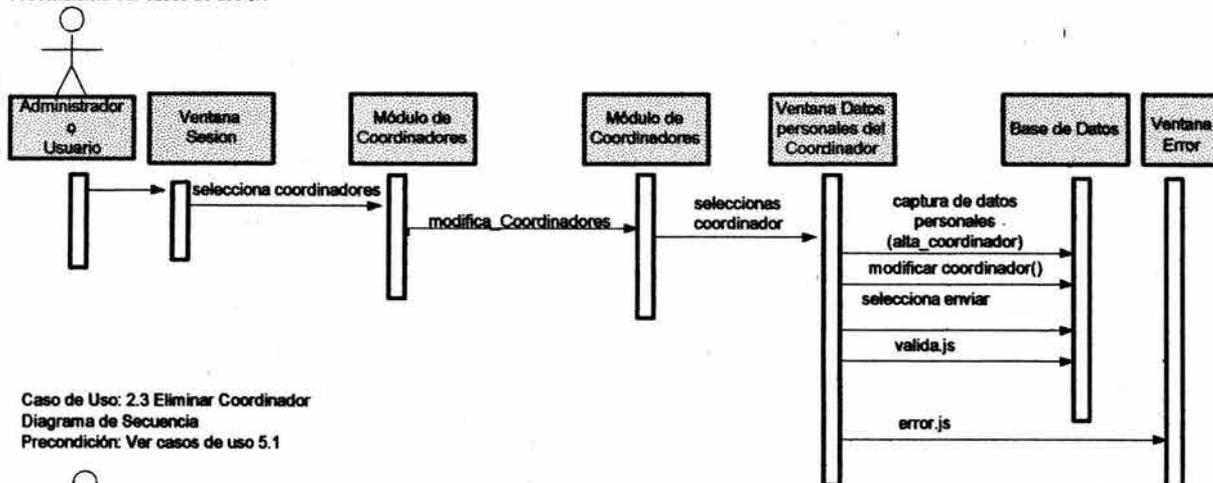


## 2. DIAGRAMA DE SECUENCIA

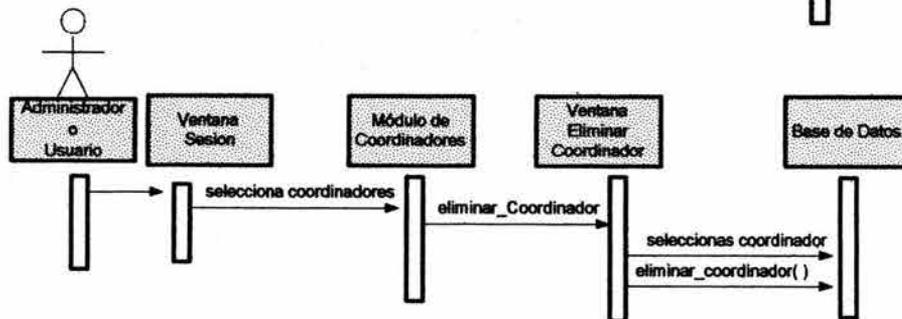
Caso de Uso: 2.1 Alta de Coordinador  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1



Caso de Uso: 2.2 Modificar Coordinador  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1



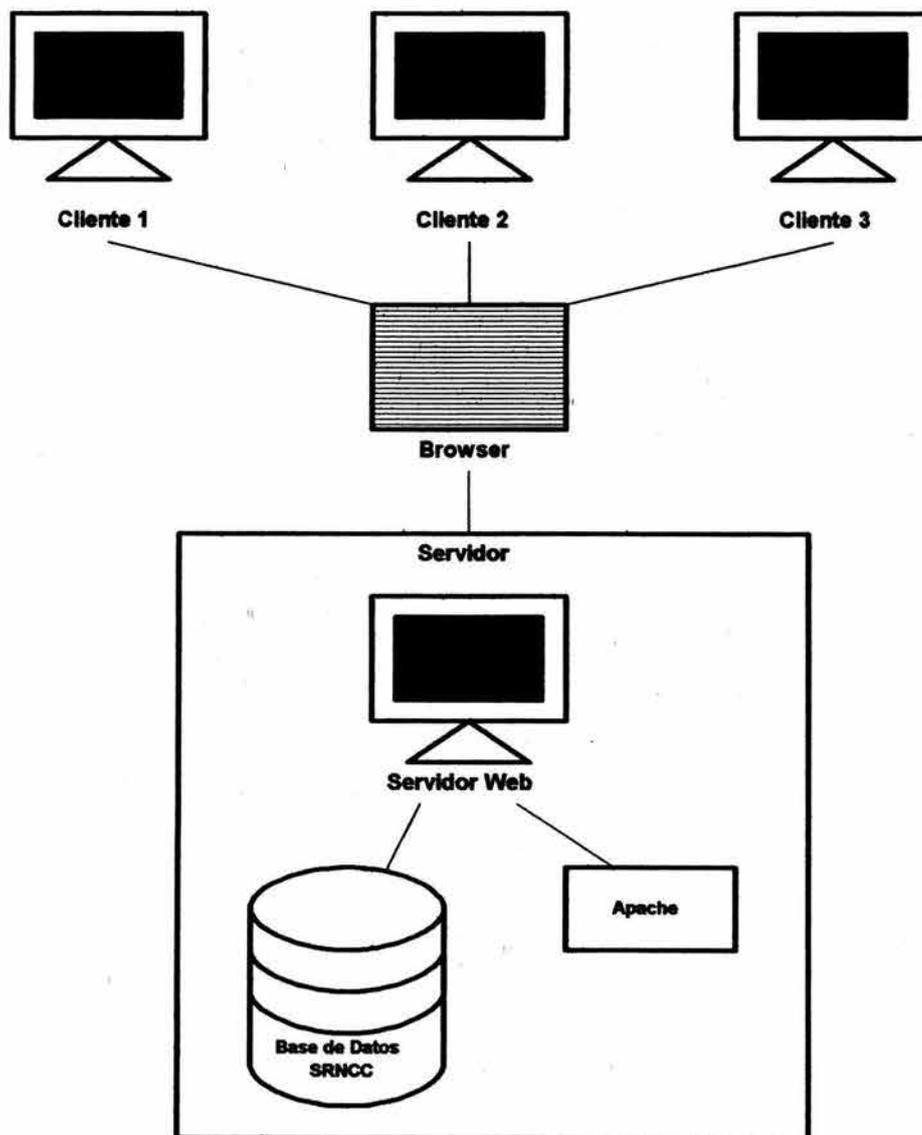
Caso de Uso: 2.3 Eliminar Coordinador  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1



### 3. DIAGRAMA DE COMPONENTES

Este diagrama tiene como objetivo mostrar los componentes de instalación que contendrá el sistema, el diagrama se divide básicamente en tres partes:

- 1) Las máquinas que se llaman clientes.
- 2) El browser por donde accedan los clientes.
- 3) El servidor donde esta la base de datos y apache que se encargará de la información persistente.



#### **4. PLAN DE PRUEBAS DE INTEGRACIÓN**

Las pruebas de integración tienen como objetivo verificar que el sistema está siendo construido apropiadamente con respecto a los requerimientos en todas sus partes de manera que funcionen en conjunto. Estas pruebas se realizarán en cada ciclo de vida del sistema.

- Se revisará el código antes de compilarlo de manera que se puedan detectar los errores de sintaxis y después se hará la corrección a través de los resultados de la compilación.
- En cuanto a los nuevos elementos que se lleguen agregar se deberá desarrollar la forma de control de cambios y en base a eso se añadirá el nuevo elemento. De la misma forma se revisará el código antes de compilarlo.

La documentación a través de las pruebas se llevará a cabo en cada momento del sistema. En el caso del primer prototipo para evaluar se hará uso de uno de los usuarios como prueba para detectar cualquier tipo de error que llegase a suscitar y que los desarrolladores no lograsen detectar.

**FORMA REGD**

Nombre: Leif Oswaldo Alduncin Gil Fecha: 17/02/03  
 Equipo: SISTCAP Ciclo: I  
 Fecha Número Tipo Inyectado Eliminado Tiempo de Defecto corregido  
 corrección

22/04/03	1	1	Dis	Dis	1 min	X
----------	---	---	-----	-----	-------	---

Descripción: Había un error de ortografía en la portada de la fase de Diseño

22/04/03	2	2	Dis	Dis	1 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Había un error de ortografía en Arquitectura 3 capas

22/04/03	3	3	Dis	Dis	1 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Había 3 errores de ortografía en Capa de presentación

22/04/03	4	4	Dis	Dis	1 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Se eliminaron unas comillas en el subtítulo de cada cuartilla en la capa de presentación (con un total de seis cuartillas)

22/04/03	5	5	Dis	Dis	1 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Se eliminaron unas comillas en el subtítulo de Capa lógica de aplicación y se removieron dos faltas ortográficas

22/04/03	6	6	Dis	Dis	1 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Se eliminaron tres faltas ortográficas en el Diagrama de Dependencias

22/04/03	7	7	Dis	Dis	4 min.	X
----------	---	---	-----	-----	--------	---

Descripción: Se eliminaron faltas ortográficas en nombramiento y así mismo se agregaron nuevas s funciones no antes consideradas, se actualizó el estándar de reuso y se corrigió una falta ortográfica en el Plan de Pruebas de Integración

# SISTCAP

---

## CAPITULO 15

### FASE DE IMPLEMENTACIÓN

#### Productos

- **Estándar de codificación.**
- **Dependencia entre módulos.**
- **Detalle de los módulos.**
- **Gráfica de archivos.**
- **Solicitudes de cambios.**
- **Funciones del sistema.**
- **Funciones de reuso.**
- **REGD**

## ESTÁNDAR DE CODIFICACIÓN

### Tabla de Contenido

- 1. Nombre de archivo.**
  - 1.1. Sufijos
  - 1.2. Nombre comunes de archivos.
- 2. Organización de archivos**
  - 2.1. Archivos Fuente postgres
    - 2.1.1. Archivos Fuente JavaScript
    - 2.1.2. Archivos Fuente Html
    - 2.1.3. Comentarios de archivos postgres
    - 2.1.4. Comentarios de archivos JavaScript
    - 2.1.5. Comentarios de archivos Html.
  - 2.2. Importación de archivos.
  - 2.3. Declaración de clases en postgres
- 3. Indentación**
  - 3.1. Longitud de línea
  - 3.2. Agrupación de líneas.
- 4. Comentarios.**
  - 4.1. Comentarios de fin de línea.
  - 4.2. Comentarios en bloque.
- 5. Declaraciones**
  - 5.1. Número por línea.
- 6. Sentencias**
  - 6.1. Sentencia if o if else
  - 6.2. Sentencia for
  - 6.3. Sentencia while.
  - 6.4. Sentencia switch.
- 7. Convenciones de Nombres.**

## ESTÁNDAR DE CODIFICACIÓN

### 1.Nombre de archivo.

#### 1.1 Sufijos

El software utiliza los siguientes sufijos:

Tipo de archivo	Sufijo
Fuente de postgres	.php
Fuente de JavaScript	.js
Imagen	.gif o .jpg
Fuente de HTML	.htm o html

#### 1.2 Nombres comunes de archivos

Nombre del archivo	Uso
ejemplo de un nombre	El nombre para los archivos, siempre en minúsculas
Directorio	El nombre de un directorio, siempre en minúsculas

### 2.Organización de Archivos

- Un archivo consiste de secciones que se deben separar entre líneas en blanco y comentarios en el encabezado, los comentarios a lado del código (opcional).
- Los archivos muy grandes son incómodos, se recomienda no pasar más de 200 líneas de código.

#### 2.1 Archivos Fuente postgres

Los archivos fuente postgres deben de tener solo una clase y puede involucrar a más de una interfaz.

Los archivos fuente de postgres tienen el siguiente orden:

- Comentarios en el encabezado que de una breve explicación de lo que se encarga el archivo.
- Declaraciones de las variables y/o include, métodos POST o GET.
- Declaración de clase o interfaz.

##### 2.1.1 Archivos Fuente JavaScript

Los archivos fuente de JavaScript siguen el siguiente orden:

- Comentarios en el encabezado que de una breve explicación de lo que se encarga el archivo.
- Declaración de funciones.

## ESTÁNDAR DE CODIFICACIÓN

### 2.1.2 Archivos Fuente Html

- Comentarios en el encabezado que de una breve explicación de lo que se encarga el archivo.
- Declaraciones de apertura, titulo, body, etcétera.

### 2.1.3 Comentarios de archivos postgres

- Todos los comentarios deben comenzar con dos diagonales invertidas al inicio  
//Esto es un ejemplo

### 2.1.4 Comentarios de archivos JavaScript

- Todos los comentarios deben comenzar con dos líneas al inicio  
//Esto es otro ejemplo

### 2.1.5 Comentarios de archivos Html

- Todos los comentarios deben comenzar con un picoparéntesis y a continuación un signo de admiración y para finalizar el comentario de debe cerrar el picoparéntesis.  
<! Este es un comentario Html >

### 2.2 Importación de archivos

- Para el caso de postgres se puede utilizar la sentencia include y require
- Para Html la sentencia <FORM></FORM>

### 2.3 Declaración de clases en postgres

- Para la declaración de una clase debe ser el nombre con minúsculas  
class ejemplo  
{  
  
}

## 3. Indentación

La indentación se debe dar con la tecla tab [← →] o tabulaciones (no es específica), también pueden ser cuatro a ocho espacios según sea el caso.

### 3.1 Longitud de línea

Se deben evitar líneas con más de 100 caracteres, pues se puede hacer ilegible el código, excepto el código HTML por sus características propias.

## ESTÁNDAR DE CODIFICACIÓN

### 3.2 Agrupación de líneas

Cuando la expresión no puede ser insertada en una misma línea, se pueden dividir bajo las siguientes principios:

- División después de una coma.
- División después de un operador.
- Alinear la nueva línea con el principio de la expresión en el mismo nivel de la línea anterior.
- Si lo anterior pudiese causar alguna confusión en la lectura del código se puede usar la indentación

### 4. Comentarios

Los comentarios que se pueden llegar a tener (ver puntos 2.1.3, 2.1.4, 2.1.5) que se utilizan según el tipo de archivo nos permiten agregar algo más al lado del código. No siempre es necesario agregar comentarios pues puede prestarse a confusiones en lugar de ayudar por lo cual se sugiere comentar de manera moderada y sólo si es necesario.

#### 4.1 Comentarios de fin de línea

En esta parte los comentarios podrán ir en la parte superior o inmediatamente después del código.

//Este es un ejemplo del comentario del código de abajo.

```
$conn=new conecta();//Objeto de la clase conecta
```

```
$conn->abre_conexion();
```

#### 4.1.1 Comentarios en bloque

Los comentarios en bloque se utilizan para proporcionar descripciones de archivos, métodos, etcétera. Y serán de la siguiente forma:

```
/*
```

```
Este un ejemplo de un comentario de bloque
```

```
*/
```

### 5. Declaraciones

#### 5.1 Número por línea

Se debe hacer una declaración por línea para poder dar el espacio a comentar.

```
$variable1 //Esto es una variable1
```

```
$variable2 //Esto es una variable2
```

## ESTÁNDAR DE CODIFICACIÓN

### 6. Sentencias

Las sentencias pueden ser ya sea simples (una en cada línea) o compuestas (dos o mas en la misma línea, en el caso de las sentencias compuestas estas deben estar bien diferenciadas ya sea por una asignación, paréntesis, etcétera.

#### 6.1. Sentencia if o if else

La sentencia if debe se seguida de un paréntesis y debajo de ella abrir las llaves, se dan las declaraciones y después cerrar las llaves abiertas

```
if(condición)
{
declaraciones;
}
else
{
declaraciones;
}
```

#### 6.2. Sentencia for

La sentencia for debe se seguida de un paréntesis y debajo de ella abrir las llaves, se dan las declaraciones y después cerrar las llaves abiertas.

```
for(inicialización; condición; actualización;)
{
declaraciones;
}
```

#### 6.3 Sentencia while

Una sentencia while debe tener la siguiente forma:

```
while(condición)
{
declaraciones;
}
y una sentencia vacía while:
while(condición);
```

## ESTÁNDAR DE CODIFICACIÓN

### 6.4 Sentencia switch

La sentencia switch tiene la siguiente forma:

#### swich(condición)

```
{  
case a:  
    declaraciones;  
    break;  
}
```

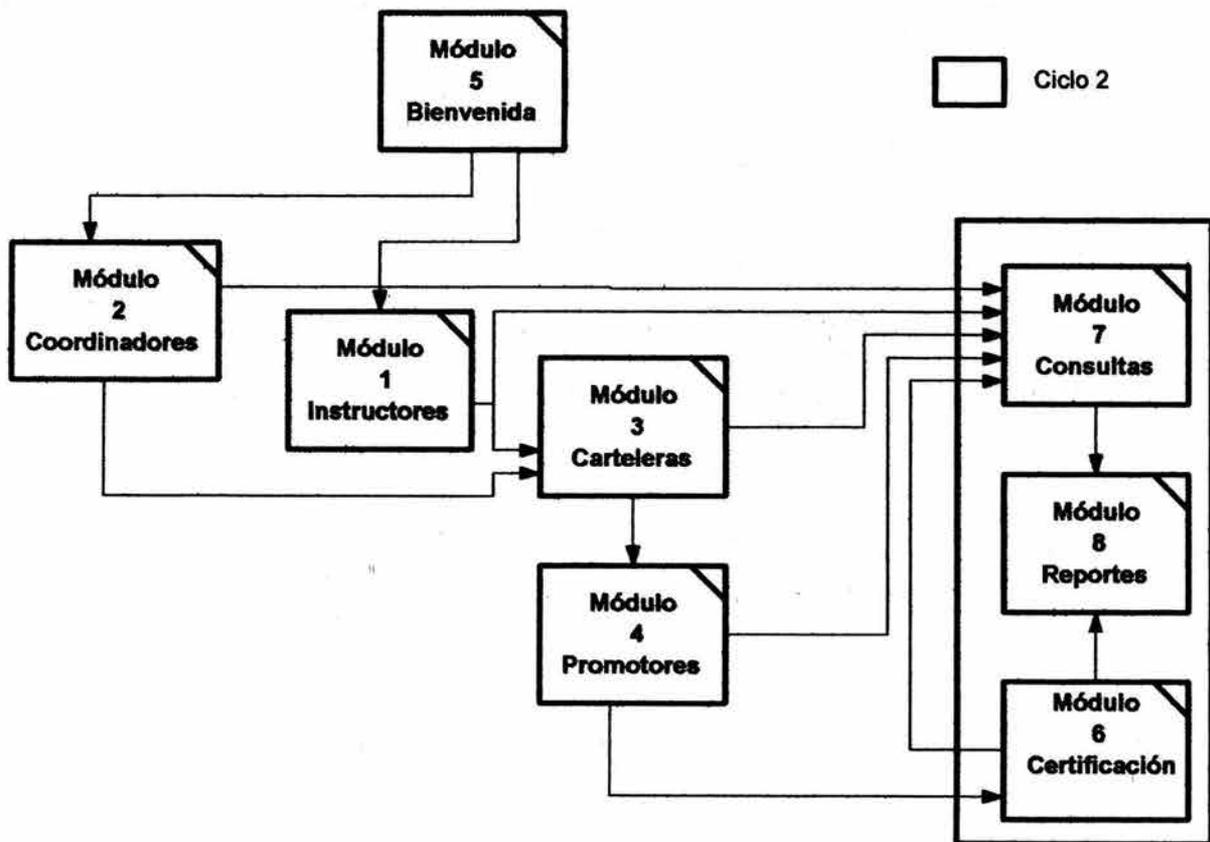
### 7. Convenciones de Nombres

Tipo de identificador	Reglas para nombrar	Ejemplos
Clases	Los nombres de las clases deben ser en minúsculas y no se debe poner palabras abreviadas, ni recortadas.	class cartelera class promotor
Métodos	Los Métodos deben ir en minúsculas y si es muy largo pueden ir con guión bajo	insertar_cartelera() eliminar_coordinador()
Variables	Las variables no tiene restricción en cuando en la definición del tiempo conjugado, estas de preferencia deben ser de nombre completo, tratar de ser claros y con definiciones con respecto en lo que se va a utilizar. Se pueden definir con guiones bajos. Si es una variable global debe ser antecedida por la palabra reservada "global". No usar variables con nombres reservadas por el sistema	\$soon; \$busqueda_cartelera; global \$actual;

## DEPENDENCIA ENTRE MODULOS

En esta sección se definen los módulos que conforman el sistema, a partir de este modelo se harán las Pruebas de Integración.

El número del módulo que se le asignó no corresponde al nivel de dependencia, sino como se especificó en el modelo de Casos de Uso General (Ver Fase de Requerimientos)



## DETALLE DE LOS MÓDULOS

Estas tablas nos permiten conocer a nivel de detalle cada módulo del sistema:

MÓDULO	NOMBRE	CONTENIDO	ESPECIFICACIONES
1	Instructores	lanza_ventanas.js	Encargada de mostrar las ventanas del menú intermedios del módulo Instructores
		errores.js	Encargada de mandar los mensajes de error del módulo de Instructores
		valida.js	Encargada de la validación de la información en datos del instructor
		sub_menú_instructor.php	Encargada de mostrar el menú del módulo de Instructor
		pide_curp_instructor.php	Encargada de requerir el curp en el módulo de Instructores
		valida_instructor.php	Encargada de la validación del curp del instructor
		datos_instructor.php	Encargada de mostrar la interfaz de los datos personales del instructor y dar de alta a una instructor en la base de datos
		elige_instructore.php	Interfaz encargada de desplegar a los instructores por orden alfabético modificar sus datos
		elige_instructore.php	Interfaz encargada de desplegar a los instructores por orden alfabético eliminar sus datos
		modifica_instructor.php	Interfaz encargada de modificar los datos personales de los instructores
		update_instructor.php	Encargada de llamar a la función que se encargará de hacer la actualización en la base de datos
		elimina_instructor.php	Encargada de mandar a llamar la función de elimina_instructor a la base de datos
		clase_instructor.php	Encargada de definir las funciones que se llamarán en las distintos archivos del módulo Instructores
		Directorio catalogos	Ver detalle de módulos catálogos
		clase_conexion.php	Encargada de la conexión a la base de datos

**DETALLE DE LOS MÓDULOS**

<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
2	Coordinadores	lanza_ventanas.js	Encargada de mostrar las ventanas del menú intermedio del módulo Coordinador
		errores.js	Encargada de mandar los mensajes de error del módulo de Coordinador
		valida.js	Encargada de la validación de la información en datos de la Coordinador
		sub_menú_cartelera.php	Encargada de mostrar el menú del módulo de Coordinador
		valida_coordinador.php	Encargada de la validación del curp del Coordinador para verificar si ya se encuentra capturado en la base de datos o es uno nuevo entre otras.
		datos_coordinador.php	Encargada de mostrar la interfaz de los datos personales de un Coordinador
		elige_coordinadorm.php	Encargada de hacer un filtro por la primera letra del apellido del coordinador para poder modificar sus datos
		elige_coordinadote.php	Encargada de hacer un filtro por la primera letra del apellido del coordinador para poder eliminar sus datos
		inserta_coordinador.php	Encargada de dar de alta un coordinador llamando las funciones de la clase coordinador
		clase_coordinador	Encargada de definir las funciones de un coordinador en el sistema
		pide_curp_coordinador	Encargada de solicitar el curp del coordinador para después llamar a valida_curp.php
		Directorio catalogos	Ver detalle de módulos catálogos
		clase_conexion.php	Encargada de la conexión a la base de datos

**DETALLE DE LOS MÓDULOS**

<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
3	Carteleras	update_cartelera.php	Encargada de hacer la inserción de las modificaciones de la cartelera en la base de datos
		carteleram.php	Encargada de mostrar los eventos para la selección a modificar
		cartelerae.php	Encargada de mostrar los eventos para la selección a eliminar
		elige_cartelera_s.php	Encargada de mostrar un mapa de la república para poder mostrar las sesiones
		cartelera_s.php	Encargada de traer las sesiones para poder tomar una dependiendo el Estado de la República
		datos_sesion.php	Encargada de mostrar la interfaz de captura de una sesion
		inserta_sesion.php	Encargada de dar de alta una sesion llamando las funciones de la clase cartelera
		elige_sesion.php	Mapa que permite seleccionar un Estado para hacer un filtro en las sesiones a seleccionar.
		sesiones.php	Encargada se seleccionar el evento de la República Mexicana
		modifica_sesion.php	Interfaz de captura
		update_sesion.php	Encargada de llamar a las funciones de la clase de cartelera para actualizar una sesión

**DETALLE DE LOS MÓDULOS**

<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
3	Cartelera	lanza_ventanas.js	Encargada de mostrar las ventanas del menú intermedio del módulo Cartelera
		errores.js	Encargada de mandar los mensajes de error del módulo de Cartelera
		valida.js	Encargada de la validación de la información en datos de la Cartelera
		sub_menú_cartelera.php	Encargada de mostrar el menú del módulo de Cartelera
		valida_cartelera.php	Encargada de la validación del curp del Coordinador para dar de alta una cartelera
		datos_cartelera.php	Encargada de mostrar la interfaz de los datos de una Cartelera
		elige_ano.php	Interfaz encargada de desplegar la opción de año, puesto que existe un folio que se le asigna a la Cartelera por año.
		inserta_cartelera.php	Encargada de dar de alta una cartelera llamando las funciones de la clase cartelera
		clase_cartelera.php	Encargada de definir las funciones que se encargaran de definir a una cartelera en el sistema
		Directorio catalogos	Ver detalle de módulos catálogos
		clase_conexion.php	Encargada de la conexión a la base de datos
		mexico1.gif	Imagen de un mapa de la República Mexicana
		elige_carteleram.php	Encargada de mostrar un mapa de la República para seleccionar y hacer un filtro según el Estado de tal manera que se puedan modificar sus datos
		elige_cartelerae.php	Encargada de mostrar un mapa de la República para seleccionar y hacer un filtro según el Estado de tal manera que se puedan eliminar sus datos
		modifica_cartelera.php	Interfaz que muestra los campos para modificar la información de cartera seleccionada

**DETALLE DE LOS MÓDULOS**

<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
4	Promotores	alta_promotor.php	Ventana encargada de pedir el curp del promotor
		submenú_promotor.php	Encargada de desplegar una ventana con el submenú del módulo de promotores
		valida_instructor.php	Encargada de validar el curp del promotor.
		datos_promotor.php	Encargada de desplegar la ventana de captura de los datos del promotor.
		inserta_promotor.php	Encargada de guardar las variables para insertar en la base de datos.
		update_promotor.php	Encarada de hacer la modificación de los datos del promotor en la base de datos
		modifica_promotor.php	Encargada de mostrar los datos para ser modificados
		elige_promotorm.php	Encargada de mostrar una ventana para seleccionar la letra del apellido del promotor y hacer una búsqueda para después modificar los datos del promotor.
		elige_promotore.php	Encargada de mostrar una ventana para seleccionar la letra del apellido del promotor y hacer una búsqueda para después eliminar los datos del promotor.
		promotorm.php	Encargado de hacer la búsqueda por la letra por apellido para la modificación de la información
		promotore.php	Encargado de hacer la búsqueda por la letra por apellido para hacer la eliminación en cascada
		elimina_promotor.php	Encargado de eliminar a un promotor.
		asigna_curso_promotor.php	Encargada de seleccionar el curp, la cartelera y asigna un curso al promotor
		elige_cartelera.php	Encargada de seleccionar una cartelera por estado en un mapa
		elige_promotor.php	Encargada de seleccionar a un promotor por orden alfabético
		valida_asignación.php	Encargada de validar la asignación de un promotor
		inserta_asigna_curso	Encargada de asignar en la base de datos un promotor a un curso.

**DETALLE DE LOS MÓDULOS**

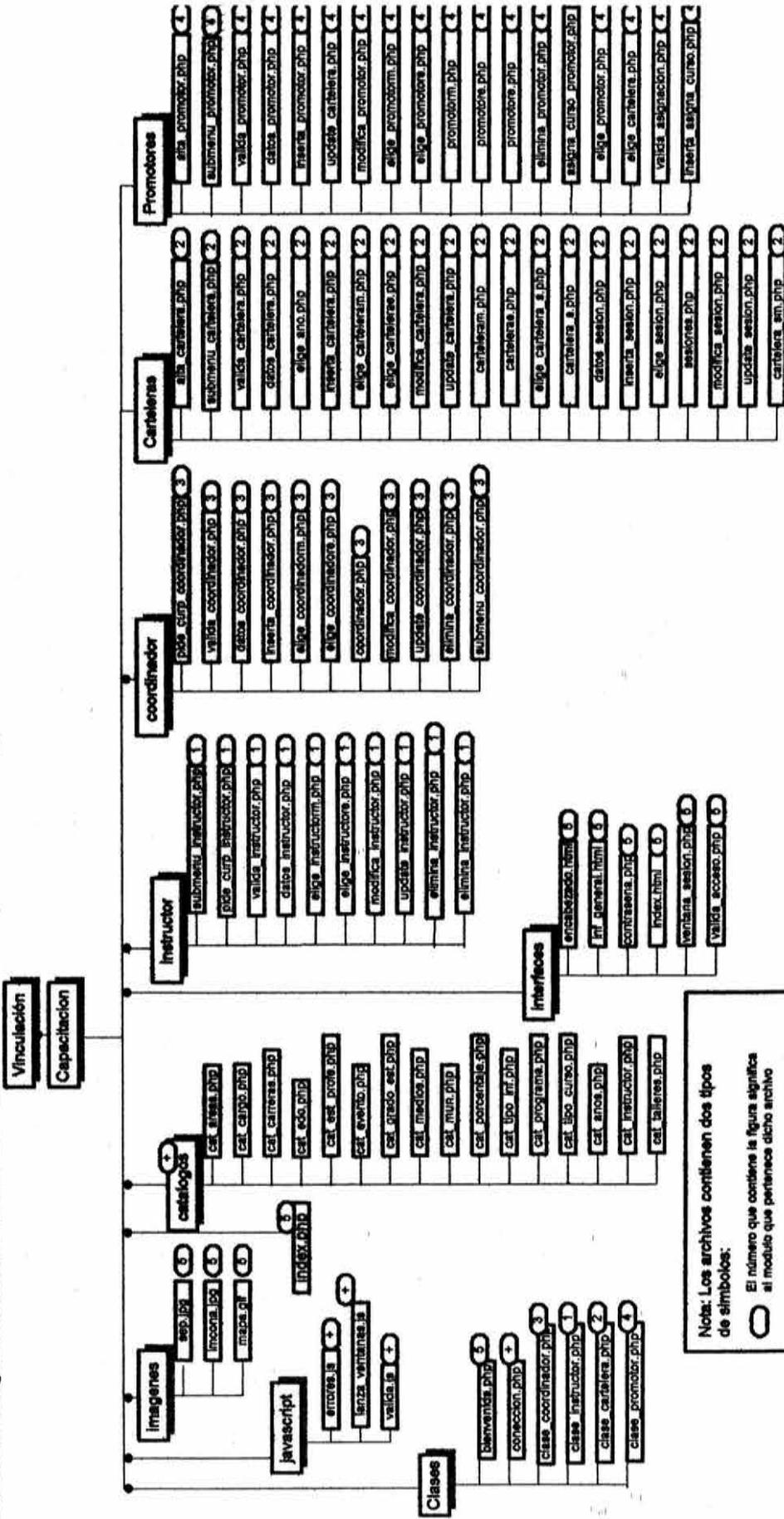
<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
5	Bienvenida	index.php	Encargada de mostrar la pagina principal de acceso al sistema
		errores.js	Encargada de mandar los mensajes de error del módulo de Bienvenida
		encabezado.html	Encargada de mostrar el encabezado en la pagina de Bienvenida
		inf_general	Encargada de mostrar la información de la pagina de Bienvenida (index.html)
		index.html	Encargada de mostrar la parte central de la pagina de Bienvenida
		contrasena.php	Encargada de mostrar los campos de usuario y de password
		valida_acceso	Encargada de validar el acceso del usuario en el login y password
		clase_conexion.php	Encargada de la conexión a la base de datos
		ventana_sesion.php	Encargada de diferenciar el acceso dependiendo el usuario

**DETALLE DE LOS MÓDULOS**

<b>MÓDULO</b>	<b>NOMBRE</b>	<b>CONTENIDO</b>	<b>ESPECIFICACIONES</b>
1,2,3,4	Catálogos	cat_areas.php	Encargada de mostrar todas las áreas de trabajo definidas en la base de datos
		cat_cargo.php	Encargada de mostrar todas los cargos definida en la base de datos
		cat_carreras.php	Encargada de mostrar todas las carreras definidas en el sistema. (Fuente obtenida de la U.N.A.M.)
		cat_edo.php	Encargada de mostrar todos los estados de la Republica Mexicana (Fuente: INEGI)
		cat_edo_profesion.php	Encargada de mostrar todos los estados profesión definidos para el sistema.
		cat_evento.php	Encargada de mostrar todos los tipos de eventos que puedan hacer en el sistema
		cat_grado_est.php	Encargad de mostrar todos los grados de estudios definidos para el sistema
		cat_medios.php	Encargada de mostrar todos los medios de comunicación definidos para el sistema
		cat_mun.php	Encargada de mostrar todos los municipios de la Republica Mexicana (Fuente: INEGI)
		cat_porcentaje.php	Encargada de mostrar el porcentaje acreditado según definido para el sistema
		cat_programa.php	Encargada de mostrar los tipos de programas definidas para el sistema
		cat_tipo_curso.php	Encargada de mostrar los deferentes tipos de cursos que hay definidos para el sistema
		cat_anos.php	Encargada de mostrar solo una tabla con los años, comienza según se definió para el sistema, a partir del año 2000.
		clase_conexion.php	Encargada de la conexión a la base de datos

## GRAFICA DE ARCHIVOS

El objetivo de la gráfica de archivos es poder conocer a detalle cada uno de los archivos que se generan en el sistema. Así mismo esta gráfica consta con un número en cada archivo, este corresponde al módulo al que es referido.



Nota: Los archivos contienen dos tipos de símbolos:  
 El número que contiene la figura significa al módulo que pertenece dicho archivo  
 Si contiene un símbolo "+" significa que pertenece a los módulos: 1, 2, 3 y 4

**SISTCAP**Responsable: Administrador de Desarrollo  
Fase: Implementación  
Ciclo: 1

Fecha: 05/05/03

## SOLICITUDES DE CAMBIOS

Nombre	Leif Oswaldo Alduncin Gil	Fecha	05/05/03
Equipo	SISTCAP	Ciclo	1

**Información del componente**

Nombre: "Catálogo Años" Dueño: SISTCAP

Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: Se requiere una tabla tipo catálogo que contenga los años para una alta de cartelera al sistema

**Beneficio del cambio**

Existe menos probabilidad de que el usuario se equivoque al especificar el año para que se de alta una cartelera dado que se le asigna un número de folio por año.

**Impactos del cambio**

Solo se requirió generar un sub menú para especificar el año de la asignación a la cartelera

**Descripción del cambio**

El usuario selecciona el año en que la cartelera se inicia, por lo cual el usuario no necesita teclear el año sino el sistema se lo proporciona, de tal manera que se le asigna un número de folio automáticamente por año llevando una secuencia.

**Estado del cambio**Aprobado  A **\*\* (Aprobado)** Rechazado  R**Información adicional**

Permitirá mayor control sobre el número del folio sobre las carteleras

**Aprobado por**Dueño del producto SISTCAP  
Adm. de calidad : Luis Hernández Nietofecha: 05/05/03  
fecha: 05/05/03

**FORMA DE SOLICITUD DE CAMBIOS**

Nombre	<u>Leif Oswaldo Alduncin Gil</u>	Fecha	<u>05/05/03</u>
Equipo	<u>SISTCAP</u>	Ciclo	<u>1</u>

**Información del componente**

Nombre: "Login" Dueño: SISTCAP

Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: El sistema contaba con un catalogo que contenía a los usuarios del sistema, pero no se podía proteger la intimidad de éstos pues cualquier usuario que entrará en la página principal podía leer los nombres.

**Beneficio del cambio**

Mayor privacidad y seguridad al sistema

**Impactos del cambio**

Se borró un catálogo con los nombres de los usuarios y la página principal cambio ahora en un text

**Descripción del cambio**

El usuario ahora necesita escribir el login y en consecuencia el password para poder acceder al sistema bajo una validación rigurosa.

**Estado del cambio**Aprobado A **\*\* (Aprobado)****Información adicional**

Mayor seguridad y privacidad

**Aprobado por**

Dueño del producto SISTCAP

fecha: 05/05/03

Adm. de calidad : Luis Hernández Nieto

fecha: 05/05/03

**FORMA DE SOLICITUD DE CAMBIOS**

Nombre	<u>Leif Oswaldo Alduncin Gil</u>	Fecha	<u>05/05/03</u>
Equipo	<u>SISTCAP</u>	Ciclo	<u>1</u>

**Información del componente**

Nombre: "Clase Conexión" Dueño: SISTCAP  
Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: Se requirió una clase que se encargara de la conexión a la base de datos, que era más sencillo estar rehusándola y no estar creando una función cada vez que se fuera el sistema una conexión a la base de datos

**Beneficio del cambio**

Mayor modularidad y ahorro de código

**Impactos del cambio**

Ahorro de código

**Descripción del cambio**

Se creó un clase que consta de sólo dos funciones básicas las cuales son: abre\_coneccion(); y cierra\_coneccion() las cuales son recurrentemente solicitadas por diferentes componentes en el sistema

**Estado del cambio**

Aprobado **A** **\*\* (Aprobado)**

**Información adicional**

Modularidad

**Aprobado por**

Dueño del producto SISTCAP  
Adm. de calidad : Luis Hernández Nieto

fecha: 05/05/03

fecha: 05/05/03

**FORMA DE SOLICITUD DE CAMBIOS**

Nombre	<u>Leif Oswaldo Alduncin Gil</u>	Fecha	<u>05/05/03</u>
Equipo	<u>SISTCAP</u>	Ciclo	<u>1</u>

**Información del componente**

Nombre: "Ventana intermedia con mapa interactivo"      Dueño: SISTCAP  
Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: En el módulo de cartelera se cambio una interfaz que sólo constaba con una tabla que tenía los nombres de los Estados de la República Mexicana, las cuales hacian un filtro para después modificar una cartelera o barrarla.

**Beneficio del cambio**

Es mucho más sencillo y visual el filtro para la modificación y eliminación para una cartelera.

**Impactos del cambio**

Se cambió completamente el submenú el cual era algo confusa.

**Descripción del cambio**

Ahora se agregó un mapa que contiene ligas sobre los estados perfectamente delimitados y una lista con los Estados y permite mayor claridad acerca de lo que requiere el usuario realizar para una modificación o eliminación.

**Estado del cambio**

Aprobado      A      **\*\* (Aprobado)**

**Información adicional**

Mayor visualización en el proceso y sencillez

**Aprobado por**

Dueño del producto SISTCAP  
Adm. de calidad : Luis Hernández Nieto

fecha: 05/05/03  
fecha: 05/05/03

**FORMA DE SOLICITUD DE CAMBIOS**

Nombre Leif Oswaldo Alduncin Gil Fecha 05/05/03  
Equipo SISTCAP Ciclo 1

**Información del componente**

Nombre: "Ventana intermedia con orden alfabético para el módulo de instructores" Dueño: SISTCAP

Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: En el módulo de instructores se agregó un submenú que cuenta con un abecedario.

**Beneficio del cambio**

La información se hace mucho mas visible y ordenada

**Impactos del cambio**

Mayor espacio y mejor administración para una modificación y eliminación, así como mejor visualización para el usuario.

**Descripción del cambio**

Al tener un filtro por abecedario el cual consta de hacer una búsqueda con la primera letra del apellido paterno y sólo regresa los instructores con esas características, de manera tal que nos ahorra espacio en la información.

**Estado del cambio**

Aprobado A **\*\* (Aprobado)**

**Información adicional**

Mayor visualización y rapidez en los procesos de modificación y eliminación.

**Aprobado por**

Dueño del producto SISTCAP  
Adm. de calidad : Luis Hernández Nieto

fecha: 05/05/03  
fecha: 05/05/03

**FORMA DE SOLICITUD DE CAMBIOS**

Nombre	<u>Leif Oswaldo Alduncin Gil</u>	Fecha	<u>05/05/03</u>
Equipo	<u>SISTCAP</u>	Ciclo	<u>1</u>

**Información del componente**

Nombre: "Alta Cartelera" Dueño: SISTCAP  
Dirección del respaldo: Archivos del sistema

**Información del cambio**

Razones para hacer el cambio: Cambio de verificación y captura en la alta de una cartelera y coordinadores

**Beneficio del cambio**

Menor cantidad de código y mejor administración en la información

**Impactos del cambio**

Es más sencillo y claro la alta de una cartelera, se ahorro el código en una ventana.

**Descripción del cambio**

Se hace una validación del curp del coordinador, en caso de existir se permite la captura de la misma, y se agregó un botón en caso de alguna actualización de los datos al coordinador, en caso contrario, (o sea que no se encuentre dado de alta el coordinador en el sistema) se manda llamar a Alta de coordinadores para darlo de alta.

**Estado del cambio**

Aprobado A **\*\* (Aprobado)**

**Información adicional**

Mejor administración del módulo alta cartelera

**Aprobado por**

Dueño del producto SISTCAP  
Adm. de calidad : Luis Hernández Nieto

fecha: 05/05/03

fecha: 05/05/03

## FUNCIONES DEL SISTEMA

### Clase Bienvenida

- Clase que contendrá las funciones del sistema en la fase de bienvenida
- Será la encargada de solicitar el login y de verificarlo.
- Será la encargada de solicitar el password y verificarlo.

### Clase Cartelera

- Clase que contendrá las funciones del sistema de cartelera
- Será la encargada de poder insertar un nuevo número de cartelera junto con otros datos.
- Será la encargada de poder modificar una cartelera dada de alta en el sistema.
- Será la encargada de poder eliminar una cartelera dada de alta en el sistema.
- De la misma forma será la encargada de las sesiones para una cartelera.
- Y será la encargada de poder modificar una sesión.

### Clase Coordinador

- Clase que contendrá las funciones del sistema para coordinadores
- Será la encargada de solicitar un Curp y de verificarlo.
- Será la encargada de poder insertar un nuevo coordinador
- Será la encargada de poder modificar al registro del coordinador
- Será la encargada de poder eliminar un coordinador en el sistema.

### Clase Instructor

- Clase que contendrá las funciones del sistema para instructores
- Será la encargada de solicitar un Curp y de verificarlo.
- Será la encargada de poder insertar un nuevo instructor
- Será la encargada de poder modificar al registro del instructor
- Será la encargada de poder eliminar un instructor en el sistema.
- De la misma forma será la encargada evaluar a los instructores.

### Clase Promotor

- Clase que contendrá las funciones del sistema para promotores
- Será la encargada de solicitar un Curp y de verificarlo.
- Será la encargada de poder insertar un nuevo promotor.
- Será la encargada de poder modificar al registro del promotor.
- Será la encargada de poder eliminar un promotor en el sistema.
- De la misma forma será la encargada de asignar cursos a los promotores.

### Clase Conexión

- Clase encargada de conectar a la Base de Datos
- Encargada también de cerrar la conexión

**FUNCIONES DEL SISTEMA**

<b>FUNCION</b>	<b>RESPONSABILIDAD</b>	<b>VARIABLES</b>	<b>MODULO QUE PERTENECE</b>
alta_cartelera(int cartelera, String)	Encargada de dar de alta una cartelera	local	Módulo 5
alta_cordinador_l (int cartelera, String)	Encargada de dar de alta los datos laborales de un coordinador	local	Módulo 5
alta_cordinador_p (int cartelera, String)	Encargada de dar de alta los datos laborales de un coordinador	local	Módulo 5
alta_instructor_p (curp, String)	Encargada de dar de alta los datos personales de un instructor	local	Módulo 1
alta_promotor_dt	Encargada de insertar los datos culturales del promotor a la base de datos	local	Módulo 4
alta_promotor_g	Encargada de insertar los datos generales del promotor a la base de datos	local	Módulo 4
alta_promotor_l	Encargada de insertar los datos laborales del promotor a la base de datos	local	Módulo 4
alta_promotor_p	Encargada de insertar los datos personales del promotor a la base de datos	local	Módulo 4
asigna_cartelera(int)	Encargada de verificar el año de la cartelera, para después poder asignársele un folio	local	Módulo 3
asigna_curso_promotor(String cart, String curp)	Encargada de asignar un curso a los promotores	local	Módulo 4
asigna_num_nuevo(int)	Encargada de asignar un número nuevo a la copia de una cartelera	local	Módulo 3
asigna_num_sesion(String,String)	Encargada de asignar un número a la sesión	global	Módulo 3
captura_sesion(int cartelera)	Encargada de crear una sesión nueva	local	Módulo 3
compruebacorreos(String)	Encargada de validar una cadena de un correo electrónico	global	Módulo 1,2 ,3 y 4
conexion(String)	Encargada de conectar a la base de datos las clases cuando se soliciten	local	Módulo 5
copiar_cartelera(String cartelera, int nueva)	Encargada de copiar una nueva cartelera y asignar su nuevo número	local	Módulo 3
curp_escrito(String)	Encargada de verificar que el curp este bien tecleado	global	Módulo 1,2 ,3 y 4
datos_sesion(String, int, int)	Encargada de regresar los datos de la información de una sesión la cual se va a modificar en la ventana sesión	global	Módulo 3
elimina_cartelera(String)	Encargada de eliminar una cartelera dada de alta en el sistema	local	Módulo 3
elimina_coordinador(String)	Encargada de eliminar un registro con la información del coordinador dado de alta en el sistema	local	Módulo 2
elimina_instructor_p (String)	Encargada de eliminar un registro con la información del instructor dado de alta en el sistema	local	Módulo 1
elimina_promotor (String)	Encargada de eliminar un registro con la información del promotor dado de alta en el sistema	local	Módulo 4

**FUNCIONES DEL SISTEMA**

<b>FUNCION</b>	<b>RESPONSABILIDAD</b>	<b>VARIABLES</b>	<b>MODULO QUE PERTENECE</b>
ejecuta_num(String)	Encargada de validar una cadena de números	global	Módulo 1,2,3 y 4
ejecuta_palabra(String)	Encargada de validar una cadena de caracteres	global	Módulo 1,2,3 y 4
elimina_promotor(String)	Encargada de eliminar un registro con la información del promotor dado de alta en el sistema	local	Módulo 4
ereg(String, String)	Encargada de comparar una cadena y otra	local	Módulo 2
error_nombre(int)	Encargada de mandar mensajes de error cuando un nombre está mal capturado, ya sea que contenga números.	global	Módulo 1,2,3 y 4
lanza_ventana(int)	Encargada de desplegar las ventanas de los submenús de los diferentes módulos, esta es una función de JavaScript	global	Módulos 1, 2, 3 y 4
marca_inscritos(String)	Encarga de asignar una carácter (*) al registro de los promotores para verificar su documentación si esta completa.	local	Módulo 4
modifica_cartelera(int cartelera, String curp)	Encargada de modificar la información de una cartelera dada de alta en el sistema	local	Módulo 3
modifica_coordinadot_p(String)	Encargada de modificar la información personal del coordinador en el sistema	local	Módulo 2
modifica_coordinadot_l(String)	Encargada de modificar la información laboral del coordinador en el sistema	local	Módulo 2
modifica_instructor_p(String)	Encargada de modificar la información personal de un instructor	local	Módulo 1
modifica_promotor_dt(String curp)	Encargada de modificar la información trabajo cultural que se encuentre en el sistema	local	Módulo 4
modifica_promotor_g(String curp)	Encargada de modificar la información general que se encuentre en el sistema	local	Módulo 4
modifica_promotor_l(String curp)	Encargada de modificar la información laboral que se encuentre en el sistema	local	Módulo 4
modifica_promotor_p(String curp)	Encargada de modificar la información personal que se encuentre en el sistema	local	Módulo 4
modifica_sesion (String sesion)	Encargada de modificar los datos de una sesión	local	Módulo 3
substr(String cadena, int comienzo, int cuantos)	Encargada de obtener los caracteres de una cadena.	local	Módulo 2
valida_curp(String)	Encargada de validar el curp del coordinador	global	Módulo 2
valida_nombre (String nombre)	Encargada de validar los caracteres para los campos nombre o de texto)	local	Módulo 1,2,3 y 4
valida_numero(String)	Encargada de la validación de un número en una cadena	global	Módulo 1,2,3 y 4
valida_password(String, String, String)	Encargada de validar el login y el password de la base de datos	local	Módulo 5

**FUNCIONES DE REUSO**

<b>REUSO</b>	<b>RESPONSABILIDAD</b>
Archivo de Catalogos	En este archivo se encuentran todos los catálogos de la base de datos
calcula_edad(String cadena, int comienzo, int cuantos)	Encargada de obtener los caracteres de una cadena
compruebacorreo(String)	Encargada de validar una cadena de un correo electrónico
coneccion()	Encargada de conectar a la base de datos las clases cuando se soliciten
curp_escrito(String)	Encargada de verificar que el curp este bien tecleado
ejecuta_num(String)	Encargad de validar una cadena de números
ejecuta_palabra(String)	Encargada de validar una cadena de caracteres
error_nombre(int)	Encargada de mandar mensajes de error cuando un nombre está mal capturado, ya sea que contenga números.
lanza_ventana(int)	Encargada de desplegar las ventanas de los submenús de los diferentes módulos, esta es una función de JavaScript
valida_nombre( String nombre)	Encargada de validar los caracteres para los campos nombre o de texto)
valida_numero(String)	Encargada de la validación de un número en una cadena

**FORMA REGD**
 Nombre: Leif Oswaldo Alduncin Gil Fecha: 05/05/03  
 Equipo: SISTCAP Ciclo: 1

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
05/03/03	1	1	Impl	Si	2 min	Si

 Descripción: Se encontraron dos errores ortográficos en la hoja de Estándares de Codificación

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
05/05/03	2	2	Impl	Si	2 min	Si

 Descripción: Se cometió un error en la dependencia entre Módulo 3 y Módulo 1, pues no existe.

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
05/05/03	3	3	Impl	Si	1 min	Si

 Descripción: Se encontró un error en el encabezado y pie de página en Estándares de Reúso

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
05/05/03	4	4	Impl	Si	2 min	Si

 Descripción: S e encontró un error en la tabla años del modelo Entidad - Relación

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
05/05/03	5	5	Impl	Si	2 min	Si

 Descripción: Se encontró un error en la Gráfica de Archivos

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado

 Descripción: \_\_\_\_\_

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado

 Descripción: \_\_\_\_\_

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado

 Descripción: \_\_\_\_\_

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado

 Descripción: \_\_\_\_\_

# SISTCAP

---

## CAPITULO 16

### FASE DE PRUEBAS DE INTEGRACIÓN Y DEL SISTEMA

#### Productos

- Pruebas de integración.
- Pruebas del sistema.
- Lista de verificación.
- Forma REGD.

## PRUEBAS DE INTEGRACIÓN

Las pruebas consistieron en la integración de los módulos que se mencionarán a través de la dependencia entre éstos (Ver dependencia de módulos), en la culminación de programación de cada módulo se espera ver un resultado (valor esperado) y así se inicia la integración del siguiente módulo hasta concluir con los módulos definidos para el ciclo 1.

NÚMERO MODULO	NOMBRE MÓDULO	VALOR ESPERADO	VALOR PROBADO	COMENTARIOS
5	Bienvenida	Entrar al sistema con login y password correcto	login y password	El módulo funcionó adecuadamente, ya que se probó con todos los usuarios y sus respectivas claves de acceso (password), así mismo se inicio adecuadamente la entrada al sistema
3	Coordinadores	Alta de un coordinador, modificación de sus datos y eliminaciones	Alta de un coordinador a través del curp, modificación a través del curp y su eliminación también a través del curp	El módulo funcionó adecuadamente puesto que se logró altas, modificaciones y eliminaciones sin ningún problema
1	Instructores	Alta de un instructor, modificación de sus datos y eliminaciones	Alta de un instructor a través del curp, modificación a través del curp y su eliminación también a través del curp	El módulo funcionó adecuadamente puesto que se logró altas, modificaciones y eliminaciones sin ningún problema
2	Cartelera	Alta, modificación y eliminación, alta de una sesión, y modificación de una sesión	Con el curp de un coordinador dado de alta en el sistema poder dar alta una cartelera, así mismo asigna un folio automáticamente, modificar la cartelera a través del curp el número que se le asignó, la alta de una sesión a través del número de cartelera y la modificación de la misma.	El módulo fue satisfactorio pues se dio de alta una cartelera a través del curp de un coordinador y se asignó un folio automáticamente, por otro lado se dio de alta una sesión a partir del número de cartelera. Se pudo modificar la cartelera a través de su número de y de el estado al que pertenece, al igual que la modificación de una sesión a partir de un número de cartelera y por último la eliminación de una cartelera por su número de asignación fueron también llevadas a cabo sin ningún problema.

**PRUEBAS DE INTEGRACIÓN**

<b>NÚMERO MÓDULO</b>	<b>NOMBRE MÓDULO</b>	<b>VALOR ESPERADO</b>	<b>VALOR PROBADO</b>	<b>COMENTARIOS</b>
4	Promotores	Alta, modificación y eliminación, alta de una sesión y modificación de una sesión	Alta de un coordinador a través del curp, modificación a través del curp y su eliminación también a través del curp	El módulo funcionó adecuadamente puesto que se logró altas, modificaciones y eliminaciones sin ningún problema

## PRUEBAS DEL SISTEMA

La finalidad de este documento es mostrar la verificación de las pruebas del sistema, además de mostrar el funcionamiento del sistema en el caso de querer ampliar y/o dar mantenimiento al proyecto.

La información está organizada en una tabla por cada paso natural en la ejecución del sistema indicando que método es el que se manda llamar, aquel que hace las operaciones en la base de datos. Aparte se muestran por módulos las pruebas de stress que se generaron y los resultados de los mismos.

CASO DE USO	METODO	ERROR	OPERACIONES	COMENTARIOS
5.1	pgconnect()	Ventana de error	Conexión a la base de datos	Este proceso se refiere a la conexión a la base de datos para la validación posterior del curp y las pruebas de estos fueron completas y satisfactorias
5.2	Ninguno	del Browser	Liga a la página acerca de información del sistema	El sistema mediante html despliega información acerca del sistema con las pruebas completas y satisfactorias
5.3	Ninguno	del browser	Liga a cerca del modo de Consulta al sistema	El sistema mediante html despliega información acerca del modo de consulta con las pruebas completas y satisfactorias
1.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión a la base de datos</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión a la base de datos</li> <li>▪ Obtención de variables</li> </ul>	Este proceso incluye la conexión a la base de datos, la ejecución de un comando de SQL de inserción y al finalizar un cierre de la conexión
1.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	Este proceso no requiere de nuevo la conexión a la base de datos pues dado que se encuentra en la misma clase (clase instructor) solo se requiere una vez, lo mismo pasa para el caso de el cierre de conexión, en el caso de una modificación a la base de datos se requiere de nuevo el método pgexec(), las pruebas fueron completas y satisfactorias.

CASO DE USO	METODO	ERROR	OPERACIONES	COMENTARIOS
1.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la eliminación de registros en la base de datos de manera satisfactoria.
2.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión</li> <li>▪ Obtención de variables</li> </ul>	Se abre una conexión nueva para la alta de un coordinador al igual se cierra en el proceso que contiene el proceso de coordinadores, también se ejecuta una inserción y la obtención de variables de la interfaz, las pruebas se realizaron completas y de manera satisfactoria.
2.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la modificación de registros en la base de datos de manera satisfactoria.
2.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la eliminación de registros en la base de datos de manera satisfactoria.
3.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión</li> <li>▪ Obtención de variables</li> </ul>	Se abre una conexión nueva para la alta de una cartelera al igual se cierra en el proceso que contiene la cartelera, también se ejecuta una inserción y la obtención de variables de la interfaz, las pruebas se realizaron completas y de manera satisfactoria.
3.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la modificación de registros en la base de datos de manera satisfactoria.
3.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la eliminación de registros en la base de datos de manera satisfactoria.
3.4	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión</li> <li>▪ Obtención de variables</li> </ul>	Se abre una conexión nueva para la alta de una sesión al igual se cierra en el proceso que contiene la sesión, también se ejecuta una inserción y la obtención de variables de la interfaz, las pruebas se realizaron completas y de manera satisfactoria.

CASO DE USO	METODO	ERROR	OPERACIONES	COMENTARIOS
3.5	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la modificación de registros en la base de datos de manera satisfactoria.
4.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión a la base de datos</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión a la base de datos</li> <li>▪ Obtención de variables</li> </ul>	Este proceso incluye la conexión a la base de datos, la ejecución de un comando de SQL de inserción a la base de datos y al finalizar un cierre de la conexión
4.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	Este proceso no requiere de nuevo la conexión a la base de datos pues dado que se encuentra en la misma clase (clase promotor) solo se requiere una vez, lo mismo pasa para el caso de el cierre de conexión, en el caso de una modificación a la base de datos se requiere de nuevo el método pgexec(), las pruebas fueron completas y satisfactorias.
4.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la eliminación de registros en la base de datos de manera satisfactoria.
4.4	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Asignación</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la de asignación de registros en la base de datos de manera satisfactoria.
4.5	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Inserción</li> <li>▪ Obtención de variables</li> </ul>	Se prueba en el proceso la de inserción de registros en la base de datos con otro numero de secuencia de manera satisfactoria.

## PRUEBAS DEL SISTEMA

### Pruebas de Stress

MODULO	NOMBRE	PROBLEMAS	COMENTARIOS
<b>1</b>	<b>Instructor</b>	Validaciones	Se realizaron diferentes capturas de la información y se encontró problemas con la validación
	<b>DETALLES</b>	<b>CAMPO</b>	<b>COMENTARIOS A DETALLE</b>
	Alta Instructor	Curp	No valida adecuadamente dado que permite la entrada incorrecta de un curp incompleto
		Primer nombre	No valida el primer nombre sino se introduce o se deja en blanco
		Segundo nombre	No valida el segundo nombre pues permite números y caracteres al mismo tiempo.
	Modificar Instructor	Apellido paterno	No valida el nombre, pues permite la entrada de números con caracteres
		Primer nombre	No valida el nombre, pues permite la entrada de números con caracteres
		Fecha de nacimiento	Al momento de regresar la información para llevar a cabo la modificación la fecha la regresa de formato inglés y no latinoamericana, como se da en la fase de captura o alta.
<b>3</b>	<b>Cartelera</b>	Validaciones	Se realizaron diferentes capturas de la información y se encontró problemas con la validación
	<b>DETALLES</b>	<b>CAMPO</b>	<b>COMENTARIOS A DETALLE</b>
	Alta Cartelera	Curp	Al momento de ingresar un curp que ya se encontraba dado de alta en el sistema la ventana de error posterior no regresa al submenú.
		Ventana de captura	Se decidió hacer la ventan grande de manera que se ajuste al tamaño del monitor
		Fecha concluye	Existe un error en la validación adecuada en la fecha de concluye, dado que permite introducir las fechas de manera incompleta
	Modificar Cartelera	Fecha inicio	Al momento de regresar la información para llevar a cabo la modificación la fecha la regresa de formato inglés y no latinoamericana, como se da en la fase de captura o alta.
		Fecha concluye	Al momento de regresar la información para llevar a cabo la modificación la fecha la regresa de formato inglés y no latinoamericana, como se da en la fase de captura o alta.

## PRUEBAS DEL SISTEMA

### Pruebas de Stress

MODULO	NOMBRE	PROBLEMAS	COMENTARIOS
2	Coordinador	Validaciones	Se realizaron diferentes capturas de la información y se encontró problemas con la validación
	<b>DETALLES</b>		<b>CAMPO</b>
	Alta Coodinador	Primer nombre	No valida el primer nombre sino se introduce o se deja en blanco
		Apellido paterno	No valida el apellido paterno
		Extensión	No valida la extensión dado que permite letras
	Modifica Coordinador	Segundo nombre	No valida el segundo nombre, pues permite la entrada de números con caracteres
		Primer nombre	No valida el nombre, pues permite la entrada de números con caracteres
		Curp	Falta la creación de un error en el caso de que al modificar el curp se establezca uno que ya se encuentre en la base de datos
		Apellido materno	No valida el apellido materno
		Fax	Permite la entrada de caracteres
Extensión		No valida la extensión dado que permite letras	
<b>DETALLES</b>		<b>CAMPO</b>	<b>COMENTARIOS A DETALLE</b>
4	Promotor	Validaciones	Se realizaron diferentes capturas de la información y se encontró problemas con la validación
	<b>DETALLES</b>		<b>CAMPO</b>
	Alta Promotor	Primer nombre	No valida el nombre, pues permite la entrada de números con caracteres y no valida campo en blanco
		Segundo nombre	No valida el segundo nombre, pues permite la entrada de números con caracteres
		Apellido paterno	No valida el apellido paterno
		Apellido materno	No valida el apellido materno
		Otra carrera	No debe hacer la validación de un campo en blanco y éste lo realiza
		Años sector	No valida la cantidad de años y permite introducir mas de 100 años
		Alta sesiones	Fecha inicial
	Modificar Sesiones	Fecha concluye	No valida la fecha
Fecha inicial		Al momento de regresar la información para llevar a cabo la modificación la fecha la regresa de formato inglés y no latinoamericana, como se da en la fase de captura o alta	
Fecha concluye		Al momento de regresar la información para llevar a cabo la modificación la fecha la regresa de formato inglés y no latinoamericana, como se da en la fase de captura o alta	
Pruebas de conexión al servidor	conexiones	Numero horas	No valida las horas dado que permite la entrada de caracteres
		Ningún problemas	No se encontró ningún problemas en la conexión al sistema de manera simultánea desde varias computadoras al mismo tiempo.

## LISTA DE VERIFICACIÓN

Los pasos a realizar durante la inspección del sistema son:

1. Revisar el código elaborado en la fase anterior. Se debe revisar que:
  - 1.1. Que cumpla con el estándar de codificación.
2. Verificación de Sintaxis.
  - 2.1. Que no existan errores de sintaxis, tales como faltas de puntos y comas, falta de paréntesis o llaves, etcétera.
  - 2.2. Que estén completas las excepciones que se pueden generar en cada caso de uso.
3. Revisar que se estén realizando todos los métodos descritos en los diagramas de clases elaborados en la fase de diseño.
  - 3.1. Revisar que cada caso de uso (durante la ejecución) concuerde con los descritos en la fase de Requerimientos.
  - 3.2. Revisar que cada caso de uso haga lo que tenga que hacer.
  - 3.3. Revisar que se realicen bien las secuencias de ejecución en cada caso de uso con respecto a las descritas en los diagramas de secuencia en la fase de Requerimientos.
4. Revisar por módulo cada interfaz de manera que :
  - 4.1. Verificación de errores.
  - 4.2. Verificación de ejecución.
  - 4.3. Verificación de validación.
5. Revisar en los documentos
  - 5.1. Revisar que se cumpla con el estándar de documentación especificado en la fase de Lanzamiento.
  - 5.2. Revisar que el documento corresponda al título.
  - 5.3. Revisar que el documento no tenga errores de ortografía.

**FORMA REGD**

Nombre:	Leif Oswaldo Alduncin Gil			Fecha:	19/05/03		
Equipo:	SISTCAP			Ciclo:	1		
Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de corrección	Defecto corregido	
19/05/03	1	1	Pru	Pru	1 min	X	
Descripción: Había un error de ortografía en la primera página de las pruebas del sistema							
Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de corrección	Defecto corregido	
19/05/03	2	2	Pru	Pru	1 min.	X	
Descripción: Había un error de ortografía en 1.1							
Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de corrección	Defecto corregido	
19/05/03	3	3	Pru	Pru	1 min.	X	
Descripción: Error en la tabla por diseño faltó un detalle en una celda de "DETALLES" del módulo 2							
Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de corrección	Defecto corregido	
19/05/03	4	4	Pru	Pru	1 min.	X	
Descripción: Error en la tabla por diseño faltó un detalle en una celda de "MODULO" del módulo 4							
Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de corrección	Defecto corregido	
19/05/03	5	5	Pru	Pru	1 min.	X	
Descripción: Error en la tabla por diseño faltó un detalle en una celda de "Pruebas de conexión al servidor"							

# SISTCAP

---

## CAPITULO 17

### FASE POSTMORTEM

#### **Productos**

- **Reporte del ciclo.**
- **Tiempos de Respuesta.**
- **Reporte de roles.**
- **Forma ECC.**

## REPORTE DEL CICLO

### Informe de la configuración al final del ciclo

Elementos de la configuración	Estados de los cambios propuestos	Estado de la implementación
LAN-1	Ningún cambio	Completo
ESTRAT - 1	Aceptado	Completo
PLAN -1	Aceptado	Completo
REQ - 1	Aceptado	Completo
DIS - 1	Aceptado	Completo
IMP - 1	Aceptado	Completo
PRU - 1	Ningún cambio	Completo
PM -1	Ningún cambio	Completo

### Productos

Productos	Tamaños	Unidades
Lanzamiento	7	Páginas
Estrategia	7	Páginas
Planeación	8 / 1	Páginas/ Archivo tipo pj (Microsoft Project)
Requerimientos	48	Páginas
Diseño	27	Páginas
Implementación	29/105	Páginas/ Archivos del sistema
Pruebas	10	Páginas
Postmortem	6	Páginas

### Tiempos de respuesta

CASO DE USO	METODO	ERROR	OPERACIONES	HORA INICIO	HORA FINAL	TIEMPO TOTAL
5.1	pgconnect()	Ventana de error	Conexión a la base de datos	9:00 am	9:10 am	10 min
5.2	Ninguno	del Browser	Liga a la página acerca de información del sistema	9:12 am	9:14 am	2 min
5.3	Ninguno	del browser	Liga acerca del modo de Consulta al sistema	9:15 am	9:17 am	2 min
1.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión a la base de datos</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión a la base de datos</li> <li>▪ Obtención de variables</li> </ul>	9:20 am	9:40 am	20 min
1.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	9:45 am	9:55 am	10 min
1.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	10:00 am	10:20 am	20 min
2.1	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión</li> <li>▪ Obtención de variables</li> </ul>	10:25 am	10:40 am	15 min
2.2	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Modificación</li> <li>▪ Obtención de variables</li> </ul>	10:45 am	11:00 am	15 min
2.3	<ul style="list-style-type: none"> <li>▪ pgexec()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Eliminación</li> <li>▪ Obtención de variables</li> </ul>	11:05 am	11:25 am	10 min
2.4	<ul style="list-style-type: none"> <li>▪ pconnect()</li> <li>▪ pgexec()</li> <li>▪ pgclose()</li> <li>▪ HTTP_GET_VARS[ ]</li> <li>▪ HTTP_POST_VARS[ ]</li> </ul>	Ventana de error	<ul style="list-style-type: none"> <li>▪ Conexión</li> <li>▪ Inserción</li> <li>▪ Cierre de conexión</li> <li>▪ Obtención de variables</li> </ul>	11:30 am	11:45 am	15 min

**Tiempos de respuesta**

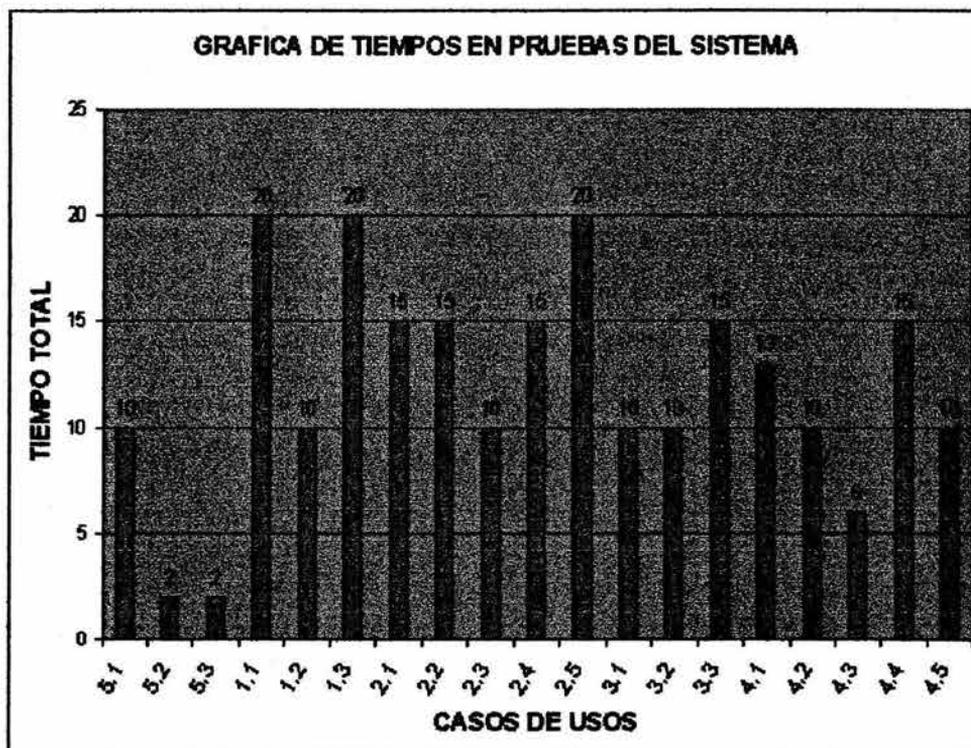
CASO DE USO	METODO	ERROR	OPERACIONES	Hora Inicio	Hora Final	TIEMPO TOTAL
2.5	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Modificación</li><li>▪ Obtención de variables</li></ul>	11:50 am	12:10 pm	20 min
3.1	<ul style="list-style-type: none"><li>▪ pconnect()</li><li>▪ pgexec()</li><li>▪ pgclose()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Conexión</li><li>▪ Inserción</li><li>▪ Cierre de conexión</li><li>▪ Obtención de variables</li></ul>	12:15 pm	12:25pm	10 min
3.2	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Modificación</li><li>▪ Obtención de variables</li></ul>	12:30 pm	12:40 pm	10 min
3.3	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Eliminación</li><li>▪ Obtención de variables</li></ul>	12:45 pm	13:00 hrs	15 min
4.1	<ul style="list-style-type: none"><li>▪ pconnect()</li><li>▪ pgexec()</li><li>▪ pgclose()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Conexión a la base de datos</li><li>▪ Inserción</li><li>▪ Cierre de conexión a la base de datos</li><li>▪ Obtención de variables</li></ul>	13:05 hrs	13:18 hrs	13 min
4.2	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Modificación</li><li>▪ Obtención de variables</li></ul>	13:25 hrs	13:35 hrs	10 min
4.3	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Eliminación</li><li>▪ Obtención de variables</li></ul>	13:39hrs	13:45 hrs	6 min
4.4	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Asignación</li><li>▪ Obtención de variables</li></ul>	13:50 hrs	14:05 hrs	15 min
4.5	<ul style="list-style-type: none"><li>▪ pgexec()</li><li>▪ HTTP_GET_VARS[ ]</li><li>▪ HTTP_POST_VARS[ ]</li></ul>	Ventana de error	<ul style="list-style-type: none"><li>▪ Inserción</li><li>▪ Obtención de variables</li></ul>	14:10 hrs	14:20 hrs	10 min

## PRUEBAS DEL SISTEMA

En la gráfica que a continuación se muestra nos permite verificar los tiempos de respuesta dependiendo el caso de uso.

En el eje X se encuentra los módulos y en el eje Y se encuentra el tiempo de la prueba.

La gráfica nos muestra que existieron algunos casos de uso que nos llevaron los mismos tiempos, lo cual pueden existir algunos procedimientos que llevan el mismo tiempo o contienen consistencia en la captura.



## REPORTE ROLES

Líder del equipo.  
Administrador de Desarrollo.  
Administrador de Configuración.

El hablar desde la perspectiva de tres roles es muy interesante, si alguien tiene la oportunidad de hacerlo no lo dude, claro que es mucha responsabilidad, trabajo y sobre todo paciencia.

El liderazgo es un elemento fundamental del trabajo en equipo, al líder del equipo no se le debe ver en ningún caso, como líder del equipo la experiencia en el proyecto permite crear habilidades a través de la experiencia en diversas responsabilidades, por ejemplo toma de decisiones, mediar con el equipo, etcétera.

A pesar de las desventajas con las que inicie el proyecto aunado a las dificultades técnicas, económicas, burocráticas, que nunca me imaginé, las cuales me dieron una enseñanza incomparable.

De parte del administrador de desarrollo fue un trabajo demoledor dado que solo fuimos dos personas que desarrollaron alrededor de 100 archivos para el sistema, pero ese trabajo tan grande nos llevó a un aprendizaje mayúsculo de muchas cosas que no sabía y que fue necesario tener para darle continuación al proyecto. Yo creo que el único "problema" fue que tal vez no tenía experiencia en algunos casos sobre algunas partes desde el punto de vista de desarrollo, pero también nos da la oportunidad de investigar y conocer como resolver los problemas que suscitaban.

Para el rol de administrador de configuración lo mas importante es la disciplina y la organización de todos los elementos a tiempo y forma, ya que hay muchos cambios en archivos, actualizaciones y te permite indagar en la búsqueda de nuevas alternativas de hardware y software que en este proyecto fueron fundamentales.

Ahora las cosas han cambiado un poco más, ya se cuenta con más equipo, pero creo que definitivamente faltan cosas desde el punto de vista de apoyo que hay que lograr, por ejemplo el servidor no cuenta con una cinta de respaldo, lo cual nos acerca a un gran riesgo para la información del proyecto, lamentablemente en el gobierno o por lo menos del lado de la cultura los recursos son escasos y anuales, por lo cual debemos hacer muy reiterativo este punto para el ciclo dos.

## REPORTE ROLES

**Administrador de Planeación**  
**Administrador de Calidad y Proceso.**

La experiencia que adquirí con la planeación fue muy importante pues dado que nunca había trabajado en proyectos tan largos, lo más importante es que a pesar de que hubo problemas en cuanto al tiempo, por diversas razones técnicas, operativas, burocráticas, familiares y económicas, etcétera; el proyecto se realizó lo más planeado posible. Dado las circunstancias que nos dejó en este ciclo se podrá dar una mejor planeación y más certera para el siguiente ciclo.

Por la calidad creo que el siguiente ciclo nos permitirá obtener medidas mucho más cercanas a comparación de el primer ciclo, pues ya este ciclo adquirí la experiencia en cuanto a tiempos, desarrollo, código, etc. Siento que me faltó más dedicación sin dejar la calidad que se requirió, pero todo esto me sirvió para tener una mejor visión en cuanto a la calidad de desarrollo de software.

Durante este ciclo desde el punto de vista del administrador de calidad y proceso nos permitió descubrir diferentes métodos de calidad los cuales son muy útiles y permiten que un software tenga la menor cantidad de errores posible, sean mucho mas útiles y funciones adecuadamente, la calidad en el desarrollo te permite desarrollar cosas con alto grado de precisión.

**Forma ECC**
 Nombre: Leif Oswaldo Alduncin Gil Equipo: SISTCAP  
 Rol: Líder del equipo No. ciclo: 1 Fecha: 02/06/2003
**Evaluación de cada rol:**

Para cada rol, evalúe el porcentaje de trabajo requerido y la dificultad durante este ciclo		
Rol	Trabajo requerido	Dificultad en el rol
Líder de Equipo	20%	20%
Administrador de Desarrollo	25%	25%
Administrador de Planeación	20%	25%
Administrador de Calidad y Proceso	25%	25%
Administrador de Configuración	10%	5%
Contribución total (100%)	100%	100%

Evalúe la contribución de cada rol. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

Evalúe cada rol de acuerdo al apoyo y ayuda proporcionado. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

Evalúe cada rol de acuerdo a su desempeño. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

**Forma ECC**
 Nombre: Luis Hernández Nieto Equipo: SISTCAP  
 Rol: Administrador de Planeación No. ciclo: 1 Fecha: 02/06/2003
**Evaluación de cada rol:**

Para cada rol, evalúe el porcentaje de trabajo requerido y la dificultad durante este ciclo		
Rol	Trabajo requerido	Dificultad en el rol
Líder de Equipo	20%	20%
Administrador de Desarrollo	25%	25%
Administrador de Planeación	20%	25%
Administrador de Calidad y Proceso	25%	25%
Administrador de Configuración	10%	5%
Contribución total (100%)	100%	100%

Evalúe la contribución de cada rol. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

Evalúe cada rol de acuerdo al apoyo y ayuda proporcionado. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

Evalúe cada rol de acuerdo a su desempeño. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Configuración	1	2	3	4	5

## Conclusiones

Con TSPi el desarrollo del software tiene muchas ventajas. Aunque cuando no se conoce a profundidad puede llegar a ser un poco confuso y podría el equipo desesperarse. Yo creo que cuando uno empieza a manejar a TSPi es bueno que siempre haya una persona que tenga la experiencia para poder dirigir al equipo para evitar las confusiones y no terminar olvidándolo. Algo que es muy importante es que el equipo debe tener bien definidos los roles que se van asumir, dado que hay que estar seguros de conocer nuestras capacidades y no tratar de aventurarnos en algunas tareas en las que no podamos tal vez realizar, esto es algo que TSPi hace mucho énfasis y es muy importante. Cuando se asume un rol es muy importante conocer a fondo las características de cada uno, saber exactamente que resultados que tendrá sobre el software, por ejemplo el Líder del equipo, dado que su trabajo en todas las partes del desarrollo es esencial, sino hay un buen liderazgo el equipo puede desmoronarse.

Otra ventaja en TSPi es que puedes ir conociendo las necesidades que se requieren poco a poco para el software a desarrollar, por ejemplo las juntas semanales las cuales es una función esencial dado que allí surgen las dudas y la comunicación es fluida y de esta manera todos se acercan mucho más al proyecto.

TSPi permite escoger que es lo que más conviene aplicar, tiene una gran serie de procedimientos que están comprobados que si se hacen de la manera adecuada funcionan. También creo que se hay algunas cosas que deberían agregarse como los diagramas de Gantt el cual fue lo que se agregó este trabajo y es una herramienta muy funcional pues permitió trabajar de manera más gráfica y ahorro muchos documentos y pudo dar un seguimiento más sencillo a la planeación y no caer en posibles confusiones y retrasos.

El proyecto SRNCC (Sistema de Red Nacional de Capacitación), es un proyecto que tuvo diversos problemas y satisfacciones, pero al utilizar la metodología de TSPi nos permitió trabajar con responsabilidad y de manera administrada. La experiencia que genera en cada proyecto da la oportunidad de obtener más experiencias y sea más fácilmente superar los problemas que pueda surgir.

El proyecto tiene excelentes oportunidades para extenderse, por ejemplo puede lograrse que los promotores culturales tengan un espacio de discusión, que puedan tener a instructores en línea de cada curso o por tema para asesorar directamente, un lugar en donde todos los promotores puedan relacionarse y contactarse. Mostrar características de contabilidad en el caso de pago a los instructores, los recursos que se asignan en cada curso, un método de inscripción, evaluación en línea, etcétera.

Me permito agregar que durante todo este tiempo que estuve trabajando en el proyecto he aprendido mucho, cuando uno termina la carrera cree que las cosas van a ser muy sencillas en el campo laboral, creo que una de las cosas que me dejó fue que debo cada día enfocar mis habilidades en lo que me gusta trabajar y de esa manera solo voy a aplicar todo aquello que aprendí en la carrera y seguirme preparando para poder ser más competitivo.

Este proyecto permitió conocer a detalle todos los factores en los que se involucra un desarrollo de software de grandes proporciones, la problemática y las ventajas del trabajo en equipo, así mismo la necesidad de abarcar hasta el más mínimo detalle en la participación, comunicación, administración y disciplina de un proyecto.

La experiencia que nos brindó la aplicación de TSPi en un proyecto que involucra ya no la calificación de una materia sino el factor económico, humano, y cooperativo en más de un área de trabajo, donde se maneja mucha información es irremplazable, pues estamos seguros que en otros proyectos nos será de vital utilidad para lograr un mejor desempeño.

## **Bibliografía**

**[CMM, 1994]**

**“CMM, Capability Maturity Model: Guidelines for Improving the Software Process”, Carnegie Mellon University, Software Engineering Series in Software Engineering, Addison-Wesley, 1194.**

**[Davis, 1993]**

**Davis Alan M. “Software Requeriments, Objects, Functions, and States”, Prentice Hall, 1999.**

**[Humphrey, 1997]**

**Humphrey Watts S, “Introduction to the Personal Software Process”, SEI Series in Software Engineering, Addison Wesley, 1997.**

**[Humphrey, 2000]**

**Humphrey Watts S, “Introduction to the Personal Software Process”, SEI Series in Software Engineering, Addison Wesley, 2000.**

**[Pressman, 1997]**

**Pressman Roger S., “Software Engineering: A Practitioner’s Approach”, New York: MacGraw – Hill Companies, Inc., 4a Edición, 1997.**

**[Sommerville and Sawyer, 19997]**

**Sommerville, I., and Sawyer, P., “Requirements Engineering: A Food Practice Guide”, New York: John Wiley & Sons.**

**[Oktaba, 2003]**

**Hanna Oktaba., “Grafica de dependencias entre necesidades”, Apuntes, U.N.A.M, 2003.**

**[Alvarez Sanchez, 2001]**

**Alvarez Sanchez Carmen D., “Aplicación de TSPi en un curso de Ingeniería de Software nivel Licenciatura”, Tesis, UNAM.**

**Carpets de Trabajo de los equipos.**

**[METASOFT, 2001]**

**Carpeta de trabajo del equipo METASOFT, generada durante el curso de Ingeniería de Software a nivel Licenciatura, UNAM**

**[Equipo Redskull Developers, 2001]**

**Carpeta de trabajo del equipo Redskull Developers, generada durante el curso de Ingeniería de Software a nivel Licenciatura, UNAM, 2003**

**Páginas de Internet.**

**<http://www.inegi.gob.mx> - Consulta Marzo 2003**

**<http://SEI.gob> - Consulta Abril 2004**

**<http://ada.fciencias.unam.mx/Ingenieriasoftware/ligas.html> - Consulta Junio 2004**

**PARTE III**

# **Anexo**

## **Anexo**

La finalidad de este documento es complementar la segunda parte de este documento en la sección práctica; sus capítulos de requerimientos y diseño.

### **REQUERIMIENTOS**

Caso de uso 2 "Administración de Coordinadores" .....	155
Caso de uso 3 "Administración de Cartelera" .....	162
Caso de uso 4 "Administración de Promotores" .....	173
Caso de uso 5 "Bienvenida" .....	184
Plan de Pruebas.....	187

### **DISEÑO**

Diagramas de secuencia 3.1 "Alta de Cartelera" .....	193
Diagramas de secuencia 3.2 "Modificar Cartelera" .....	193
Diagramas de secuencia 3.3 "Eliminar Cartelera" .....	194
Diagramas de secuencia 3.4 "Sesiones" .....	194
Diagramas de secuencia 4.1 "Alta Promotor" .....	195
Diagramas de secuencia 4.2 "Modificar Promotor" .....	195
Diagramas de secuencia 4.3 "Eliminar Promotor" .....	195
Diagramas de secuencia 4.4 "Asigna curso" .....	196
Diagramas de secuencia 4.5 "Copiar cartelera" .....	196

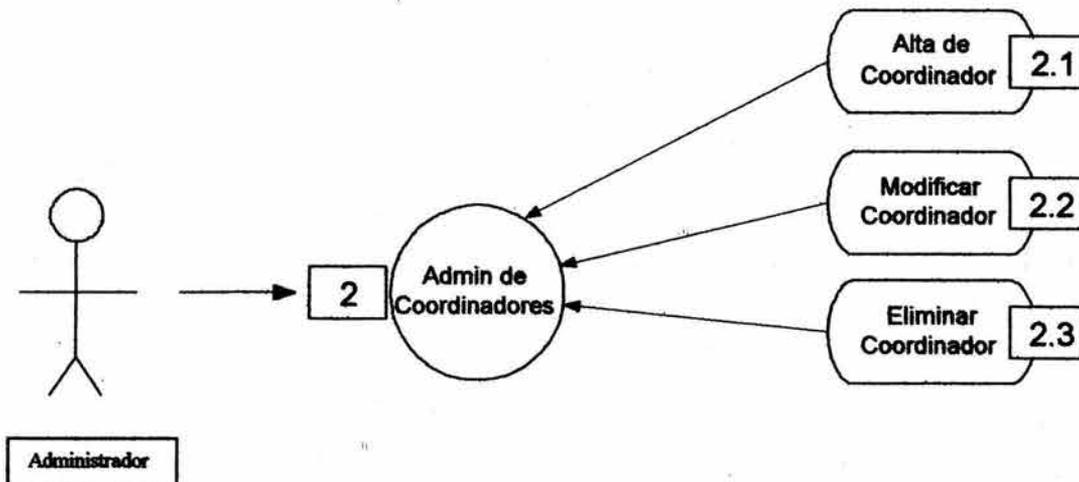
## FASE DE REQUERIMIENTOS

### CASOS DE USO

**Nombre:** 2. Administración de Coordinadores.

**Objetivo:** Administrar la información de un coordinador para darlo de alta, modificar, eliminar, consultar.

**Precondición:** Haber entrado al sistema como administrador.



## 2.1 Alta de Coordinador

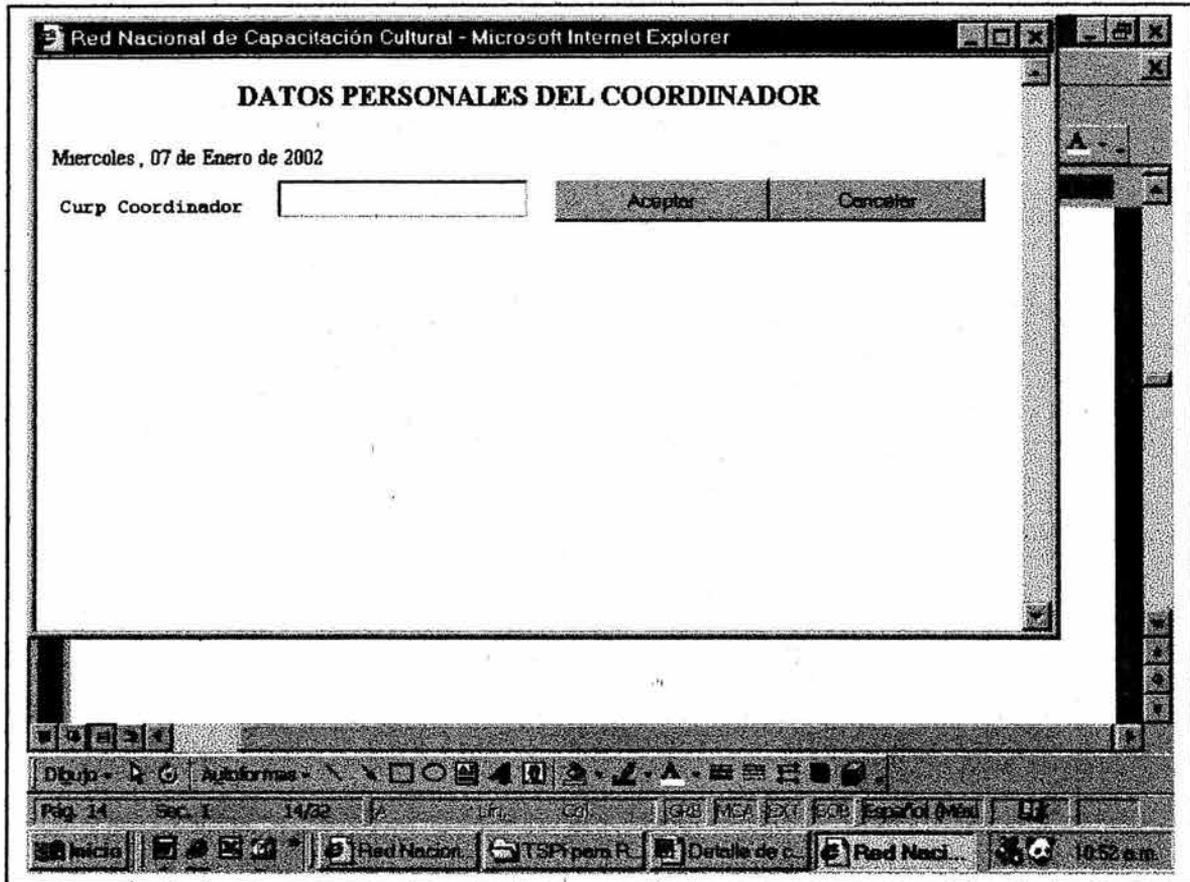
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Coordinadores" en la página de inicio.	2	El sistema abre la página "Módulo de Coordinadores"	
3	El administrador selecciona alta de instructor	4	El sistema pide el Curp del Coordinador. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador teclea el Curp y le da click en el botón aceptar.	6	El sistema valida y revisa el Curp	E 2.1

### Excepciones para 2.1

Id	Descripción	Acción
E 2.1	El administrador no introduce Curp y le da click al botón aceptar	El sistema muestra la leyenda "El Curp no puede ser vacío, favor de corregirlo". Y un botón de regresar. Regresa a la página anterior

**Poscondición:** El sistema permite acceso al modulo de captura para dar de alta a un coordinador.

# PROTOTIPO



## 2.2 Modificar Coordinador

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Coordinadores" en la página de inicio.	2	El sistema abre la página "Módulo de Coordinadores"	
3	El administrador selecciona modifica_coordinador	4	El sistema pide que seleccione el nombre o Curp del Coordinador. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al coordinador y le da click en el botón consultar.	6	El sistema muestra los datos del coordinador	

**Poscondición:** El sistema permite al administrador modificar o consultar los datos del coordinador seleccionado.

# PROTOTIPO



### 2.3 Eliminar Coordinador

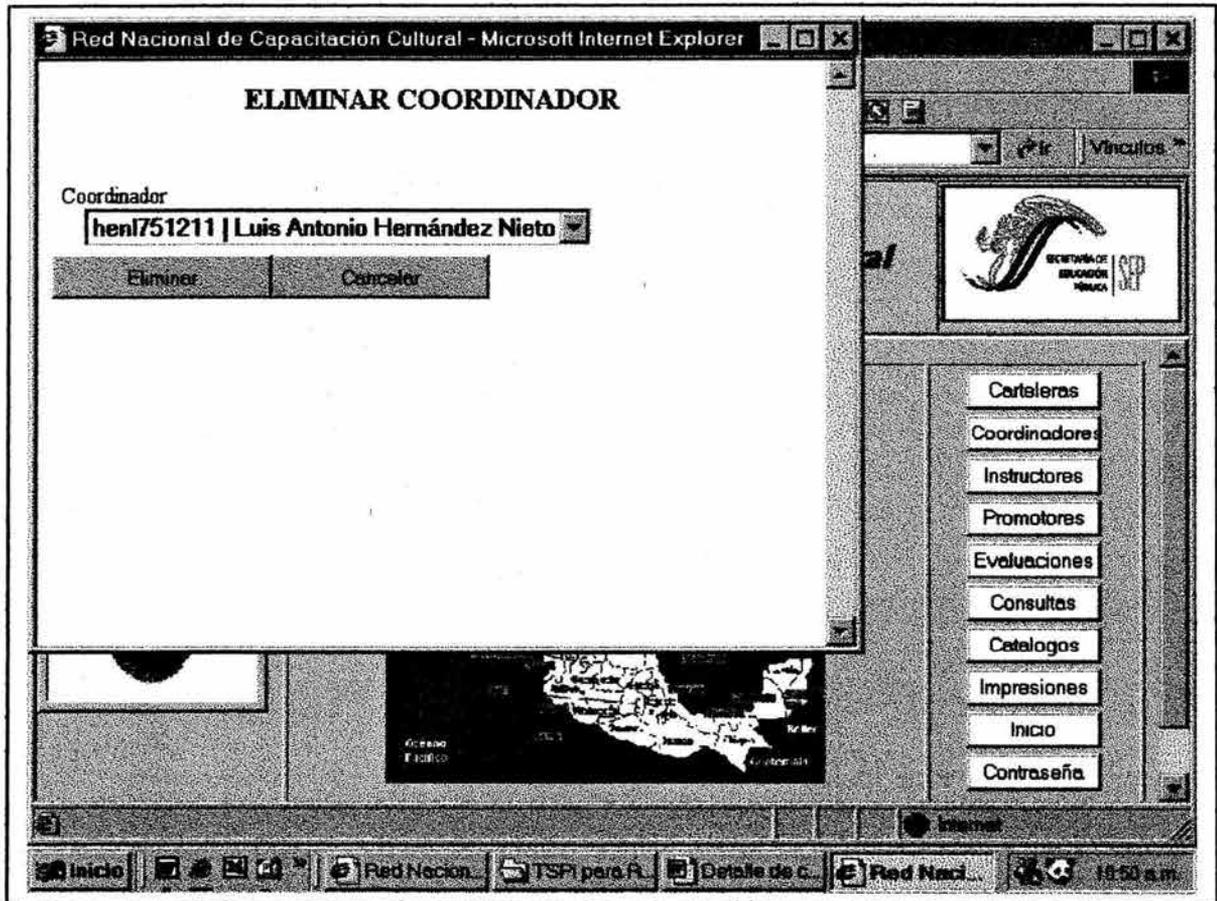
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Coordinadores" en la página de inicio.	2	El sistema abre la página "Módulo de Coordinadores"	
3	El administrador selecciona Eliminar_Coordinador	4	El sistema pide que seleccione el nombre o Curp del coordinador. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al instructor y le da click en el botón Eliminar.	6	El sistema confirma la eliminación.	E 2.3

#### Excepción para 2.3

Id	Descripción	Acción
E 2.3	El administrador decide eliminar al Coordinador o no.	El sistema muestra la pantalla de confirmación, si es afirmativa elimina al Coordinador, en caso contrario regresa al paso 3.

**Poscondición:** El sistema permite al administrador eliminar al coordinador seleccionado.

# PROTOTIPO

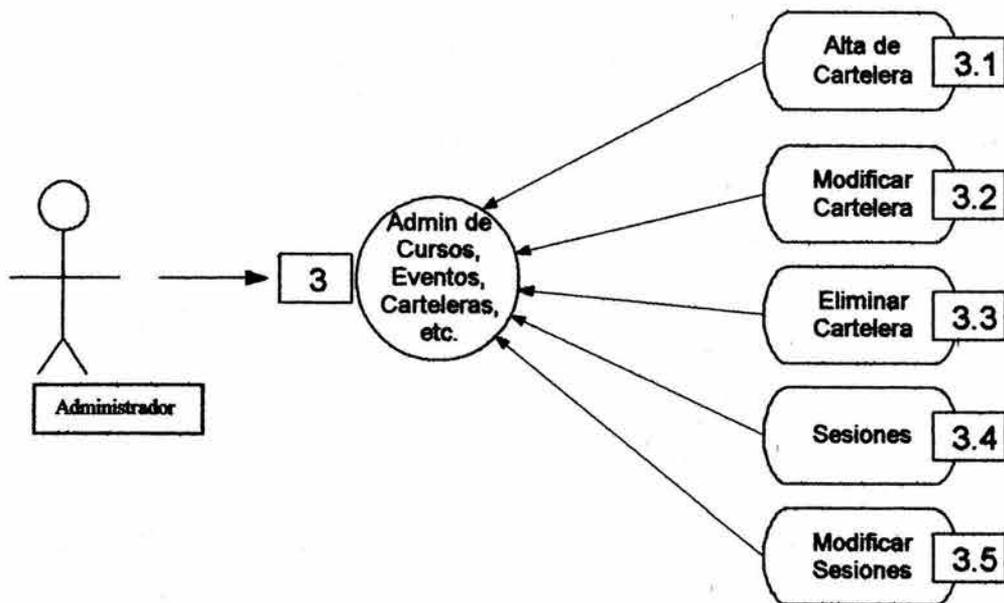


## CASOS DE USO

**Nombre:** 3. Administración de Cursos, Eventos, Carteleras, etcétera.

**Objetivo:** Administrar la información de las carteleras, cursos, eventos, seminarios, módulos, talleres, etcétera.

**Precondición:** Haber entrado al sistema como administrador.



### 3.1 Alta de Cartelera

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Cartelera" en la página de inicio	2	El sistema abre la página "Módulo de Cartelera"	
3	El administrador selecciona alta de cartelera.	4	El sistema proporciona el número de cartelera, que se compone con el número consecutivo y los dos últimos dígitos del año, además pide el curp del coordinador. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador teclea el Curp del coordinador y le da click en el botón enviar.	6	El sistema valida y revisa el Curp del coordinador.	E 3.1

#### Excepción para 3.1

Id	Descripción	Acción
E 3.1	El administrador no introduce curp y le da click al botón aceptar	El sistema muestra la leyenda "El curp no puede ser vacío, favor de corregirlo". Y un botón de regresar. Regresa a la página anterior

**Poscondición:** El sistema permite al administrador dar de alta una cartelera.

# PROTOTIPO



---

### 3.2 Modificar Cartelera

---

<b>Paso</b>	<b>Actor</b>	<b>Paso</b>	<b>Sistema</b>	<b>Excepción</b>
1	El Administrador selecciona la opción "Cartelera" en la página de inicio.	2	El sistema abre la página "Módulo de Cartelera"	
3	El administrador selecciona modificar cartelera.	4	El sistema proporciona las cartelera existentes, para que el usuario elija la cartelera a modificar	
5	El administrador selecciona la cartelera y le da click en el botón Consultar.	6	El sistema toma el número del evento y muestra sus datos para modificar	

**Poscondición:** El sistema permite al administrador modificar datos de la cartelera seleccionada.

# PROTOTIPO



### 3.3 Eliminar Cartelera

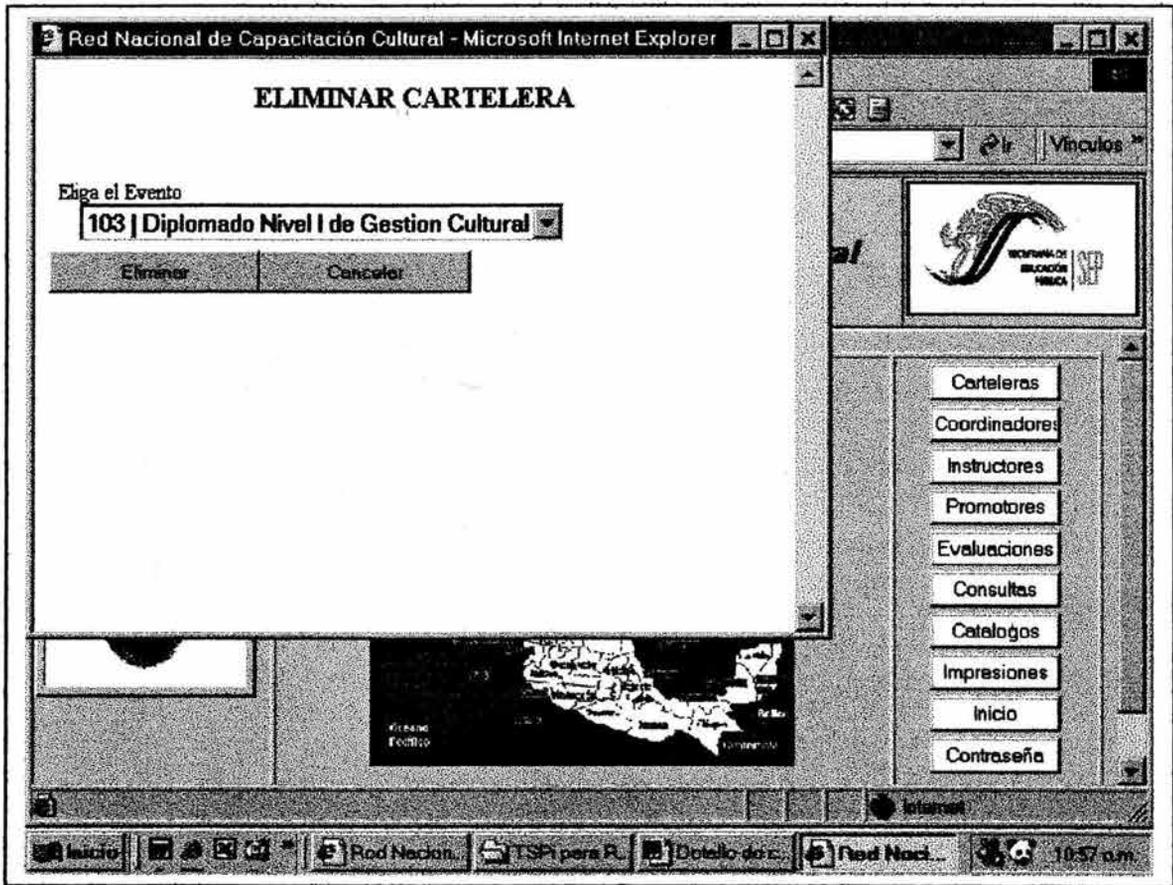
<b>Paso</b>	<b>Actor</b>	<b>Paso</b>	<b>Sistema</b>	<b>Excepción</b>
1	El Administrador selecciona la opción "Cartelera" en la página inicio	2	El sistema abre la página "Módulo de Cartelera"	
3	El administrador selecciona eliminar cartelera.	4	El sistema proporciona las cartelera existentes, para que el usuario elija la cartelera a eliminar	
5	El administrador selecciona la cartelera y le da click en el botón Eliminar.	6	El sistema toma el número del evento y aparece una ventana de confirmación.	E 3.3

#### Excepción para 3.3

<b>Id</b>	<b>Descripción</b>	<b>Acción</b>
E 3.3	El administrador decide eliminar la cartelera o no.	El sistema muestra la pantalla de confirmación, si es afirmativa elimina la cartelera, en caso contrario regresa al paso 3.

**Poscondición:** Elimina la cartelera elegida del sistema.

# PROTOTIPO



---

### 3.4 Sesiones

---

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Cartelera" en la página de inicio.	2	El sistema abre la página "Módulo de Cartelera"	
3	El administrador selecciona botón de sesiones	4	El sistema proporciona las cartelera existentes, para que el usuario elija la cartelera a la que se le se asignarán las sesiones. Si elige el botón de "Cancelar", el sistema regresa al paso 3.	
5	El administrador selecciona la cartelera y le da click en el botón Asignar sesión	6	El sistema toma el número del evento y muestra la ventana de captura para la captura de sesiones	

**Poscondición:** El sistema da de alta una nueva sesión

**Falta página**

**N° 170**

### 3.5 Modificar Sesiones

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Cartelera" en la página inicio.	2	El sistema abre la página "Módulo de Cartelera"	
3	El administrador selecciona botón de modificar sesiones	4	El sistema proporciona las cartelera existentes, para que el usuario elija la cartelera a la que se le se asignarán las sesiones y el número de sesión a modificar. Si elige el botón de "Cancelar", el sistema regresa al paso 3.	
5	El administrador selecciona el evento, el número de sesión y le da click en el botón Modificar sesión	6	El sistema toma el nombre del evento y el número de sesión y muestra la ventana de modificar la sesión.	E 3.5

Excepción para 3.5

Id	Descripción	Acción
E 3.5	El administrador introduce un número de sesión incorrecto.	El sistema muestra la leyenda que esa sesión no existe y muestra un botón para regresar.

**Poscondición:** Muestra los datos de la sesión para su modificación o consulta

# PROTOTIPO



#### 4.1 Alta de Promotor

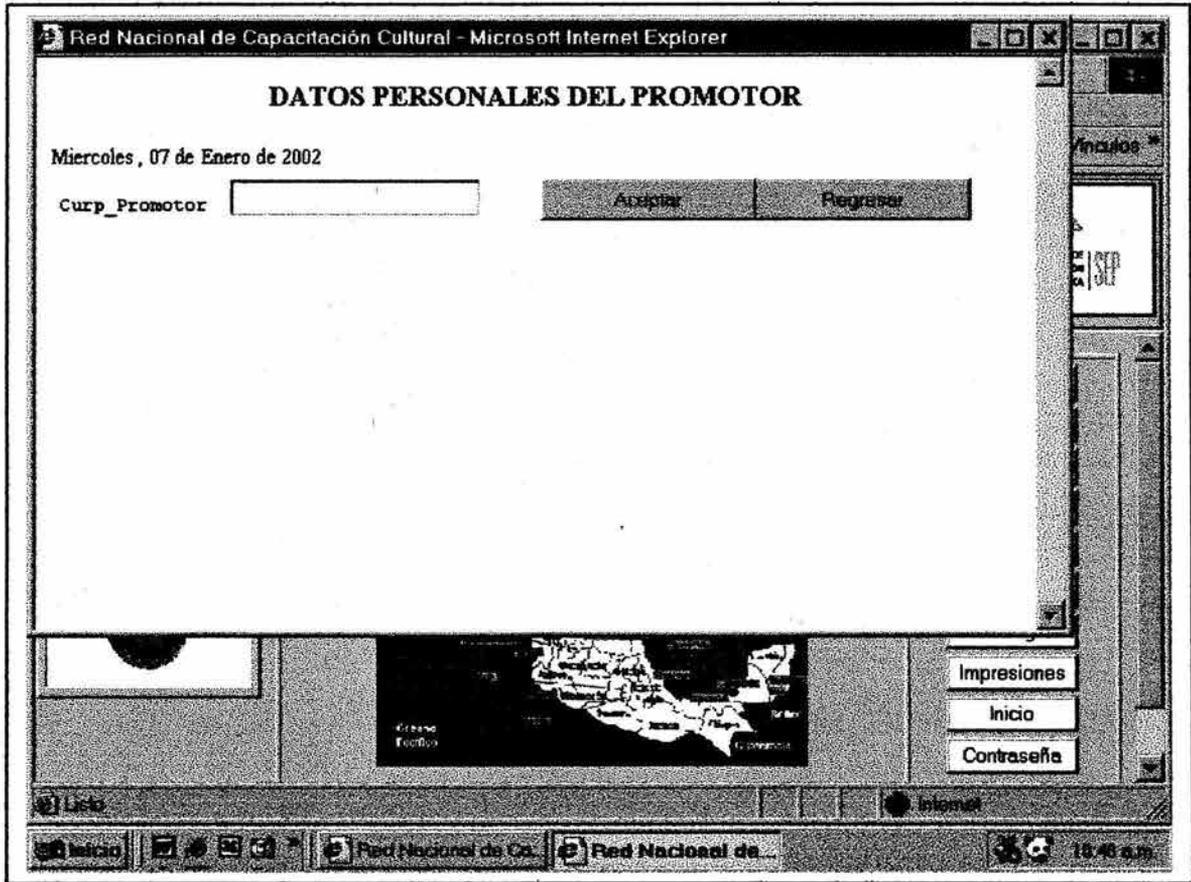
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Promotores" en la página de inicio	2	El sistema abre la página "Módulo de Promotores"	
3	El administrador selecciona alta de promotor	4	El sistema pide el Curp del Promotor. Si el administrador le da click en el botón regresar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador teclea el Curp y le da click en el botón aceptar.	6	El sistema valida y revisa el Curp	E 4.1

#### Excepciones para 4.1

Id	Descripción	Acción
E 4.1	El administrador no introduce Curp y le da click al botón aceptar	El sistema muestra la leyenda "El Curp no puede ser vacío, favor de corregirlo". Y un botón de regresar. Regresa a la página anterior

**Poscondición:** El sistema permite acceso al modulo de captura para dar de alta a un promotor.

# PROTOTIPO



---

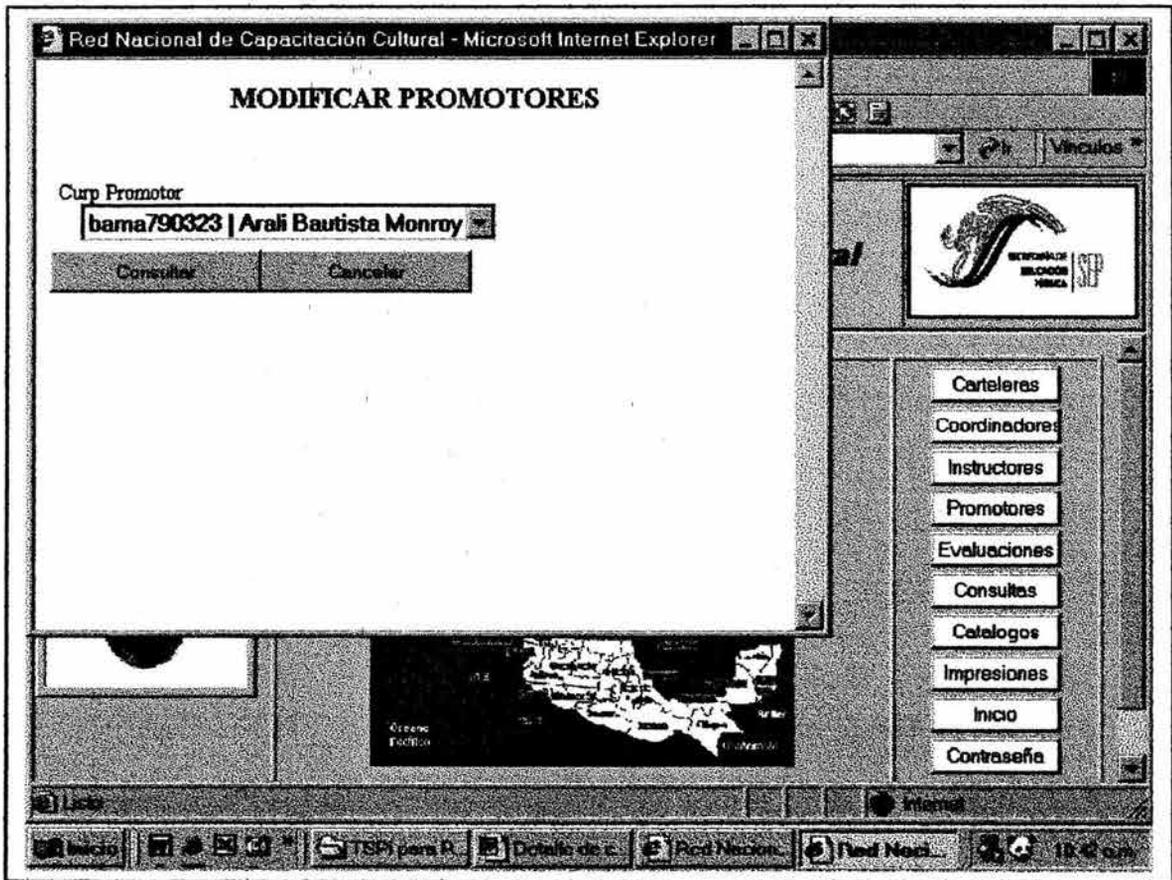
## 4.2 Modificar Promotor

---

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Promotores" en la página de inicio.	2	El sistema abre la página "Módulo de Promotores"	
3	El administrador selecciona modifica_promotor	4	El sistema pide que seleccione el nombre o Curp del Promotor. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al promotor y le da click en el botón consultar.	6	El sistema muestra los datos del promotor	

**Poscondición:** El sistema permite al administrador modificar o consultar los datos del promotor seleccionado.

# PROTOTIPO



### 4.3 Eliminar Promotor

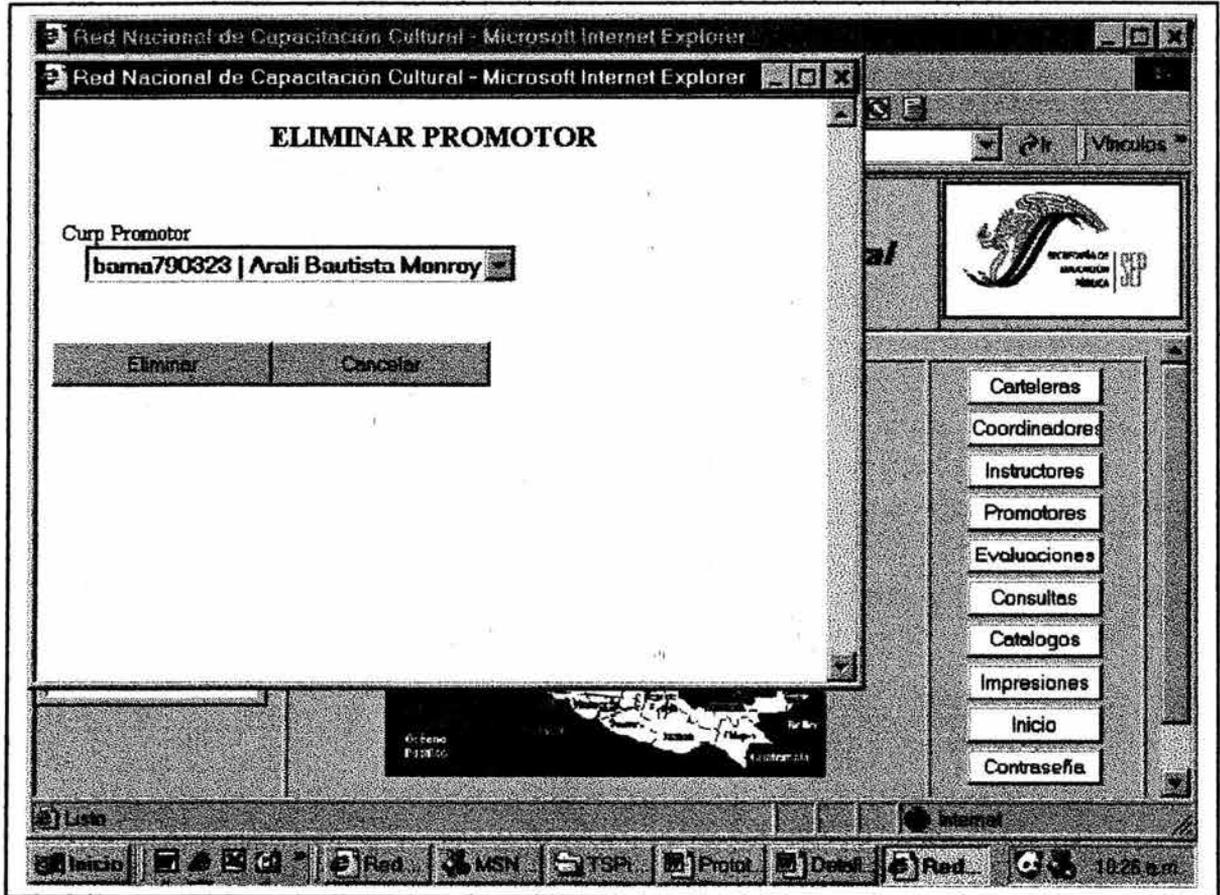
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Promotores" en la página de inicio	2	El sistema abre la página "Módulo de Promotores"	
3	El administrador selecciona Eliminar_Promotor	4	El sistema pide que seleccione el nombre o Curp del Promotor. Si el administrador le da click en el botón cancelar, lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador selecciona al promotor y le da click en el botón Eliminar.	6	El sistema confirma la eliminación.	E 4.3

#### Excepción para 4.3

Id	Descripción	Acción
E 4.3	El administrador decide eliminar al Promotor o no.	El sistema muestra la pantalla de confirmación, si es afirmativa elimina al Promotor, en caso contrario regresa al paso 3.

**Poscondición:** El sistema permite al administrador eliminar al promotor seleccionado.

# PROTOTIPO



---

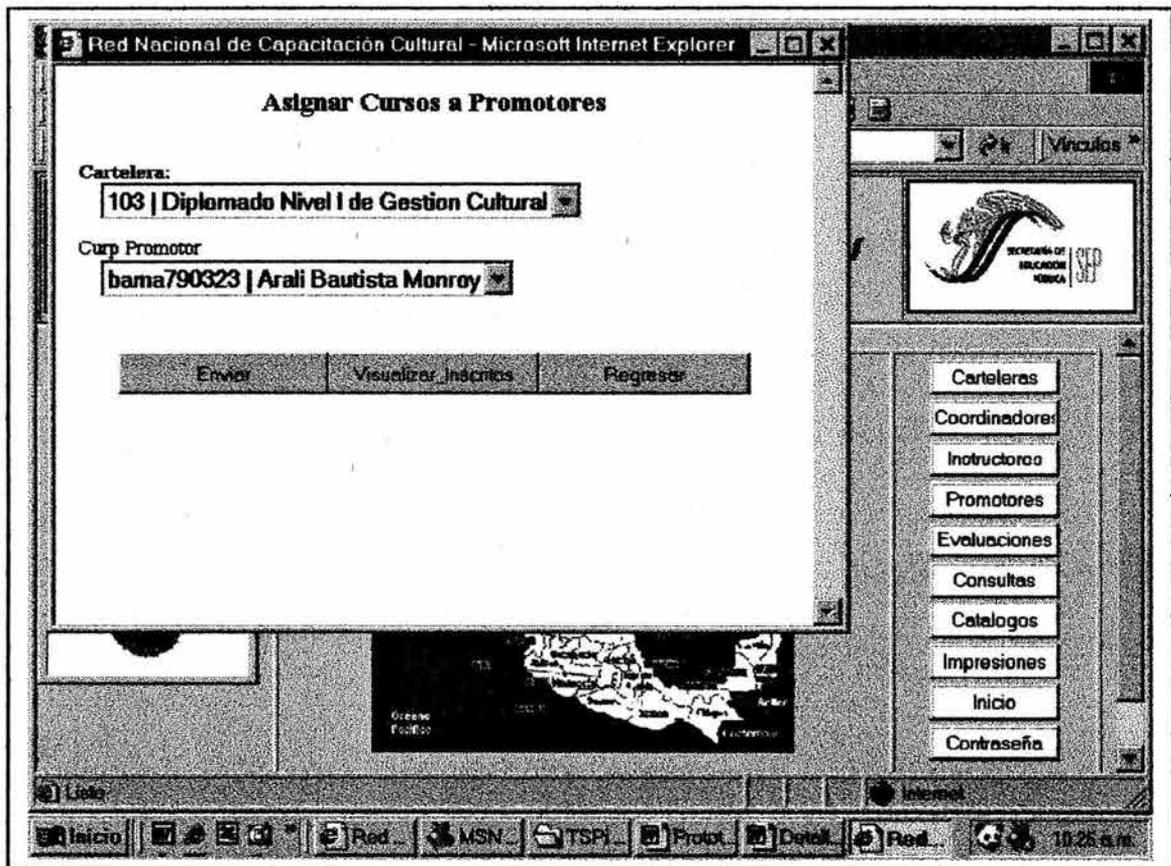
#### 4.4 Asignar cursos

---

Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Promotores" en la página de inicio	2	El sistema abre la página "Módulo de Promotores"	
3	El administrador selecciona Asigna Curso	4	El sistema pide que seleccione el nombre y número de la cartelera nombre y Curp del Promotor. Si el administrador le da click en el botón "regresar", lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador le da click al botón enviar	6	El sistema confirma la asignación,	
7	El administrador tiene la opción de visualizar a los promotores inscritos, le da click en el botón Visualizar inscritos	8	El sistema muestra a los promotores que tiene inscritos con respecto a la Cartelera y al Curp	

**Poscondición:** El sistema permite al administrador asignar cursos al promotor seleccionado.

# PROTOTIPO



## 4.5 Copiar cartelera

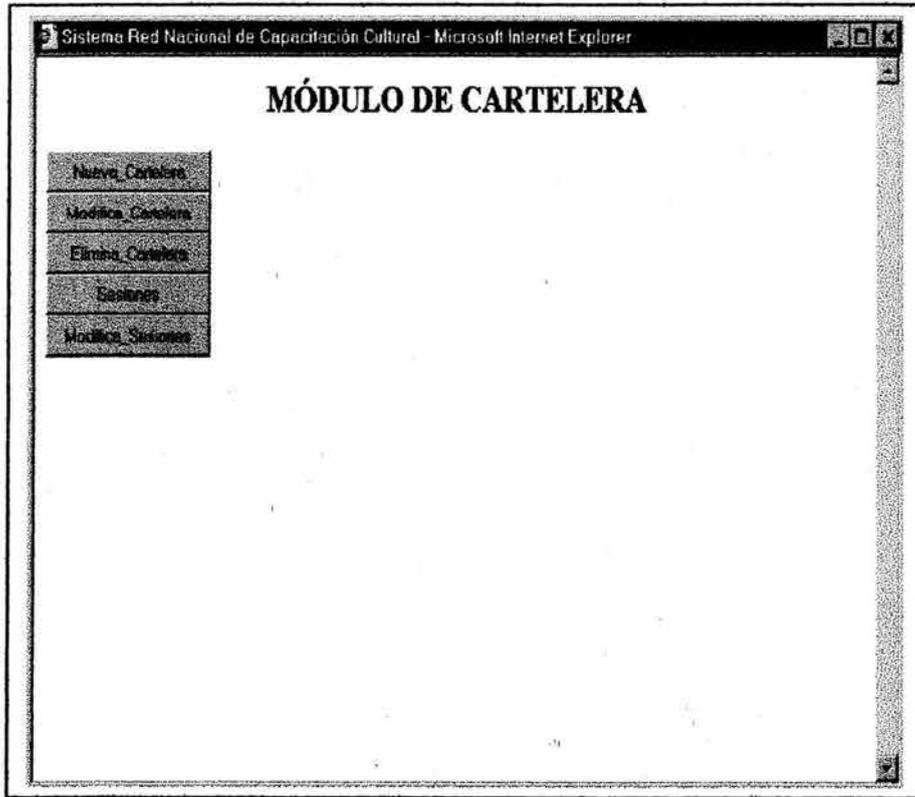
Paso	Actor	Paso	Sistema	Excepción
1	El Administrador selecciona la opción "Promotores" en la página de inicio.	2	El sistema abre la página "Módulo de Promotores"	
3	El administrador selecciona Copiar_cartelera	4	El sistema pide que seleccione el número de la cartelera y escriba el número de la nueva cartelera. Si el administrador le da click en el botón "regresar", lo regresa al paso 2, en caso contrario se realiza el paso 5.	
5	El administrador le da click al botón copiar	6	El sistema confirma la copia,	E 4.5

### Excepción para 4.5

Id	Descripción	Acción
E 4.5	El administrador decide copiar la cartelera con todos los promotores a una nueva.	El sistema muestra la pantalla de confirmación, si es afirmativa copia la cartelera, en caso contrario regresa al paso 3.

**Poscondición:** El sistema permite al administrador una cartelera con todos los promotores asignados a una nueva.

## PROTOTIPO

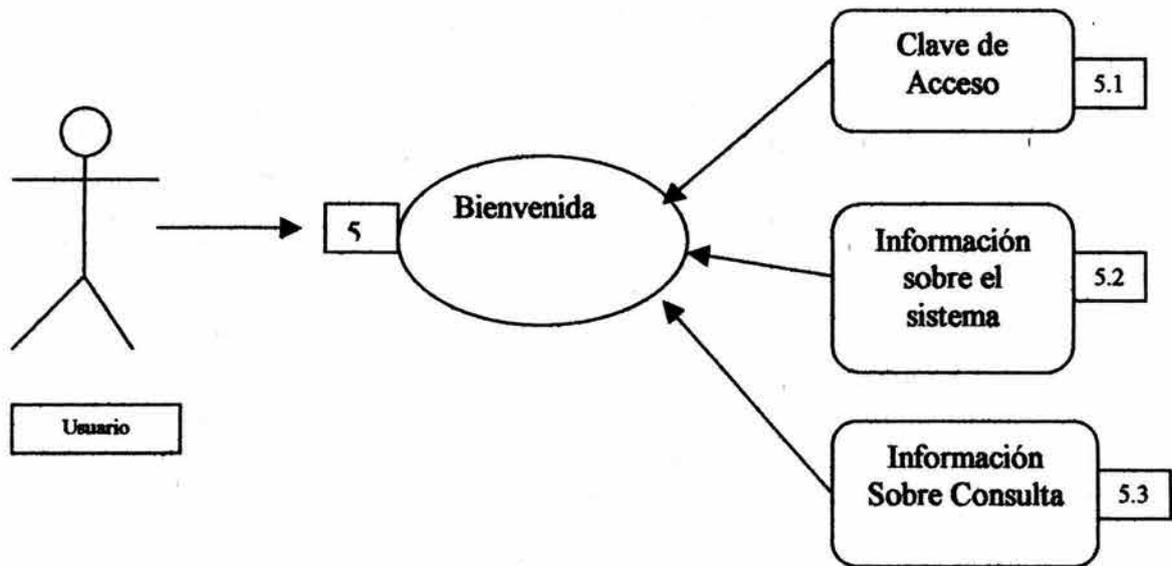


## CASOS DE USO

**Nombre:** 5. Página de Bienvenida

**Objetivo:** Dar acceso a los diferentes usuarios al sistema, al igual permitir información acerca del mismo y sobre sistema de consulta, el cual se especifica para el ciclo 2

**Precondición:** Ser usuario en el sistema con un login y un password y de otro modo solo ser visitante de la parte de documentación (se refiere a información del sistema o de CONACULTA)



---

## 5.1 Clave de Acceso

---

Paso	Actor	Paso	Sistema	Excepción
1	Usuario o el administrador selecciona el "login" para la clave de acceso	2	El sistema despliega una lista de nombres de usuarios	E5.1
3	El administrador o el usuario introduce el "password"	4	El sistema valida el nombre seleccionado y el password	E5.1.1

---

### Excepción para 5.1 y 5.1.1

---

Id	Descripción	Acción
E 5.1	El usuario selecciona un nombre de una lista de usuarios	El sistema puede o no tener al usuario dado de alta, por lo cual puede no desplegar su nombre
E5.1.1	El usuario introduce un "password" al sistema	El sistema valida el "password" del usuario, manda un mensaje de error en caso de que no sea acorde al sistema.

**Poscondición:** El sistema permite al usuario o al administrador ingresar al sistema.

---

## 5.2 Información sobre el sistema

---

Paso	Actor	Paso	Sistema	Excepción
1	Usuario ingresa en el modulo de información del sistema	2	El sistema despliega una ventana con información acerca del sistema	

**Poscondición:** El sistema permite al usuario o al administrador ingresar a la información general acerca del sistema



## PLAN DE PRUEBAS

### 2.1 Alta de Coordinador

Datos Pruebas	Resultado esperado
Dar click en el botón coordinadores	Abrir modulo de coordinadores
Dar click en alta coordinadores	Abrir modulo de alta coordinadores
Introducir el Curp de Coordinador	El sistema da de alta un nuevo coordinador
Introducir el Curp de Coordinador mal escrito	Mensaje de error de Curp mal escrito
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 2.2 Modificar Coordinador

Datos Prueba	Resultado esperado
Dar click en modificar coordinador	Abrir modulo de modificar coordinador
Seleccionar al coordinador de una lista	El sistema no tiene al coordinador dado de alta
Seleccionar al coordinador de una lista	El sistema contiene al coordinador dado de alta
Dar click en continuar con un coordinador en el sistema	El sistema muestra los datos del coordinador para modificar
Dar click en cancelar	El sistema no ejecuta ninguna acción

### 2.3 Eliminar Coordinador

Datos Prueba	Resultado esperado
Dar click en eliminar coordinador	Abrir modulo de eliminar coordinador
Seleccionar coordinador de una lista	El sistema no tiene el coordinador
Seleccionar coordinador de una lista	El sistema lo despliega en una lista
Eliminar coordinador seleccionado	Mensaje de verificación para eliminar
Dar click en cancelar	El sistema regresa a la pantalla anterior

## PLAN DE PRUEBAS

### 3.1 Alta de Cartelera

Datos Prueba	Resultado esperado
Dar click en el botón cartelera	Abrir modulo de carteleras
Dar click en alta cartelera	Abrir modulo de alta cartelera
Introducir el Curp de Coordinador	El sistema da de alta una nueva cartelera
Introducir el Curp de Coordinador mal escrito	Mensaje de error de Curp mal escrito
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 3.2 Modificar Cartelera

Datos Prueba	Resultado esperado
Dar click en modificar cartelera	Abrir modulo de modificar cartelera
Seleccionar el evento	El sistema no tiene el evento registrado
Seleccionar el evento	El sistema lo despliega en una lista
Dar click en continuar con un evento seleccionado	El sistema muestra los datos del evento para modificar la cartelera
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 3.3 Eliminar Cartelera

Datos Prueba	Resultado esperado
Dar click en eliminar cartelera	Abrir modulo de eliminar cartelera
Seleccionar evento	El sistema no tiene evento
Seleccionar el evento	El sistema lo despliega en una lista
Eliminar evento de la cartelera seleccionada	Mensaje de verificación para eliminar
Dar click en cancelar	El sistema regresa a la pantalla anterior

## PLAN DE PRUEBAS

### 3.4 Alta de sesiones

Datos Prueba	Resultado esperado
Dar click en alta sesiones	Abrir modulo de alta sesiones
Seleccionar de la lista un evento	El sistema da de alta una nueva sesión
Seleccionar de la lista un evento	El sistema no tiene la sesión dada de alta
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 3.5 Modificar Sesiones

Datos Prueba	Resultado esperado
Dar click en modificar sesión	Abrir modulo de modificar sesión
Seleccionar el evento	El sistema no tiene el evento registrado
Seleccionar el evento	El sistema lo despliega en una lista
Dar click en continuar con un evento seleccionado e introducir el número de la sesión	El sistema muestra los datos del evento para modificar la cartelera
Dar click en continuar con un evento seleccionado e introducir mal el número de la sesión	Mensaje de error de número de sesión incorrecta.
Dar click en cancelar	El sistema regresa a la pantalla anterior

## PLAN DE PRUEBAS

### 4.1 Alta de Promotor

Datos Prueba	Resultado esperado
Dar click en el botón Promotores	Abrir modulo de promotores
Dar click en alta Promotores	Abrir modulo de alta promotores
Introducir el Curp de Promotor	El sistema da de alta un nuevo promotor
Introducir el Curp de Coordinador mal escrito	Mensaje de error de Curp mal escrito
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 4.2 Modificar Promotor

Datos Prueba	Resultado esperado
Dar click en modificar promotor	Abrir modulo de modificar promotor
Seleccionar al promotor de una lista	El sistema no tiene al promotor dado de alta
Seleccionar al promotor de una lista	El sistema contiene al promotor dado de alta
Dar click en continuar con un promotor en el sistema	El sistema muestra los datos del promotor para modificar
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 4.3 Eliminar Promotor

Datos Prueba	Resultado esperado
Dar click en eliminar promotor	Abrir modulo de eliminar promotor
Seleccionar promotor de una lista	El sistema no tiene el promotor
Seleccionar promotor de una lista	El sistema lo despliega en una lista
Eliminar promotor seleccionado	Mensaje de verificación para eliminar
Dar click en cancelar	El sistema regresa a la pantalla anterior

## PLAN DE PRUEBAS

### 4.4 Asignar Curso

Datos Prueba	Resultado esperado
Dar click en asignar curso	Abrir modulo de asignar curso
Seleccionar Curp del promotor de una lista	El sistema no tiene el promotor registrado
Seleccionar Curp del promotor de una lista	El sistema lo despliega en una lista
Seleccionar cartelera de una lista	El sistema no la cartelera registrada
Seleccionar cartelera de una lista	El sistema lo despliega en una lista
Dar click en enviar con una cartelera y un curp seleccionado de la lista	El sistema asigna un curso
Dar click en visualizar inscritos	El sistema permite ver cuantos promotores inscritos hay por la cartelera y el promotor seleccionados
Dar click en cancelar	El sistema regresa a la pantalla anterior

### 4.5 Copiar Cartelera

Datos Prueba	Resultado esperado
Dar click en copiar cartelera	Abrir modulo de copiar
Seleccionar cartelera de una lista	El sistema no tiene la cartelera registrada
Seleccionar cartelera de una lista	El sistema lo despliega en una lista
Dar click en copiar con una cartelera seleccionado de la lista e ingresar un número de cartelera nuevo y correctamente escrito	El sistema copia a los promotores de una cartelera a una nueva
Dar click en copiar con una cartelera seleccionado de la lista e ingresar un número de cartelera nuevo y correctamente escrito	Mensaje de error por el número de cartelera mal escrito
Dar click en cancelar	El sistema regresa a la pantalla anterior

## PLAN DE PRUEBAS

---

### 5.1 Página de bienvenida

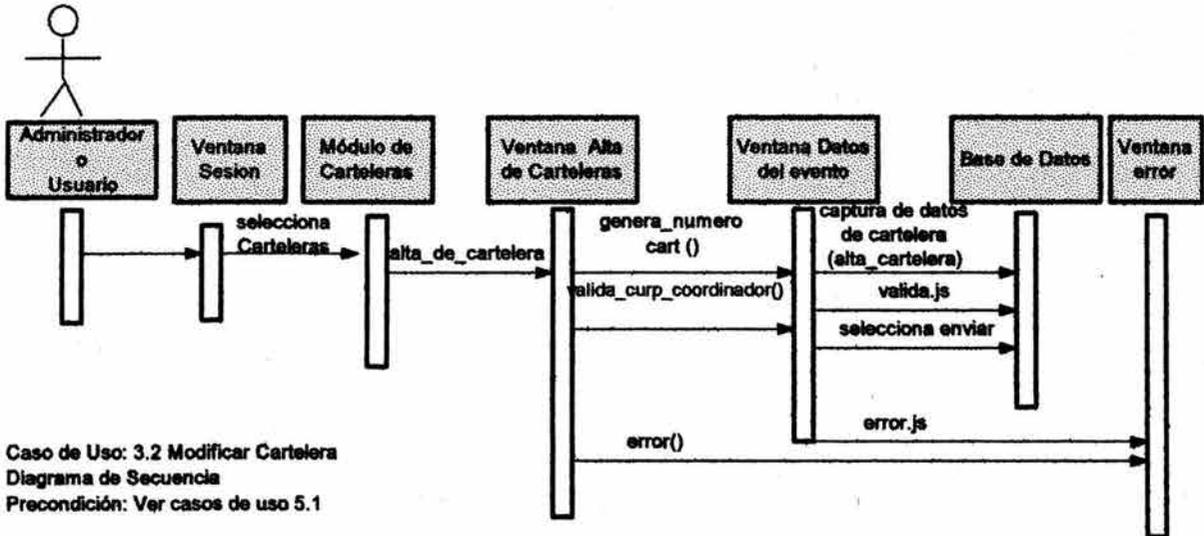
---

<b>Datos Prueba</b>	<b>Resultado esperado</b>
Seleccionar login	Lista desplegable de usuarios
Introducir password	Acepta ingreso
Introducir password incorrecto	Mensaje de error de password incorrecto
No introducir password	Mensaje de error de password incorrecto

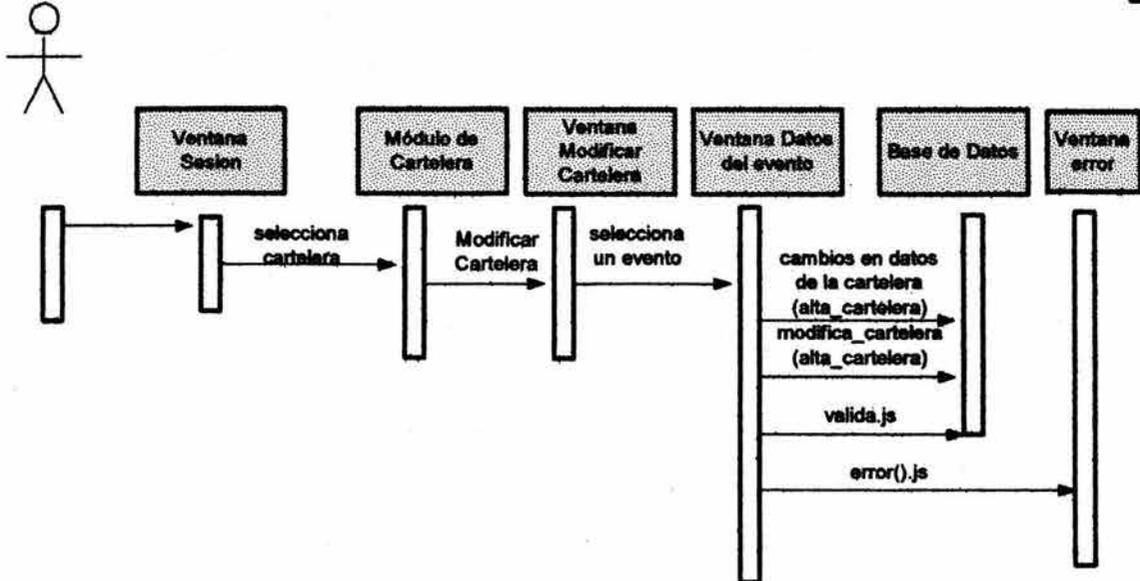
# DISEÑO

## DIAGRAMA DE SECUENCIA

Caso de Uso: 3.1 Alta de Cartelera  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1

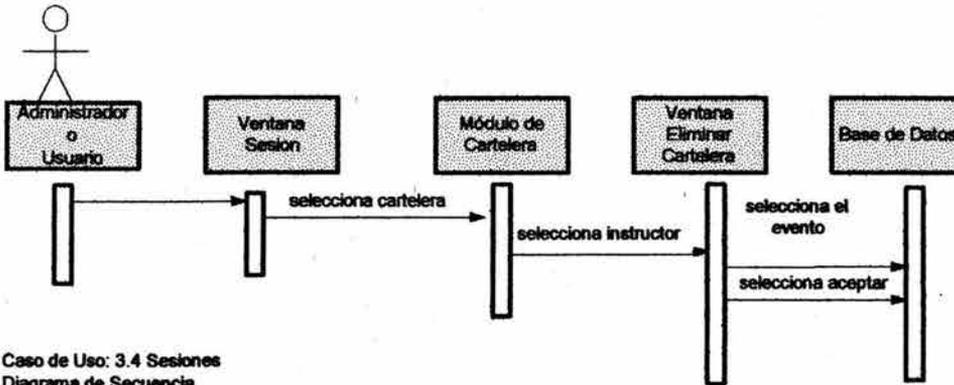


Caso de Uso: 3.2 Modificar Cartelera  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1

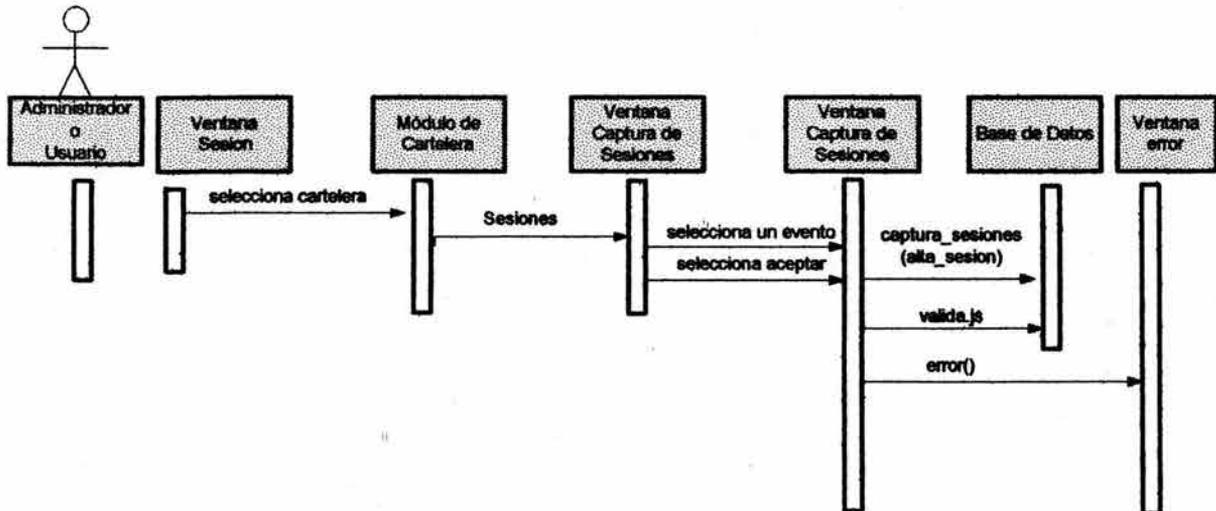


# DIAGRAMA DE SECUENCIA

Caso de Uso: 3.3 Eliminar Cartelera  
 Diagrama de Secuencia.  
 Precondición: Ver casos de uso 5.1

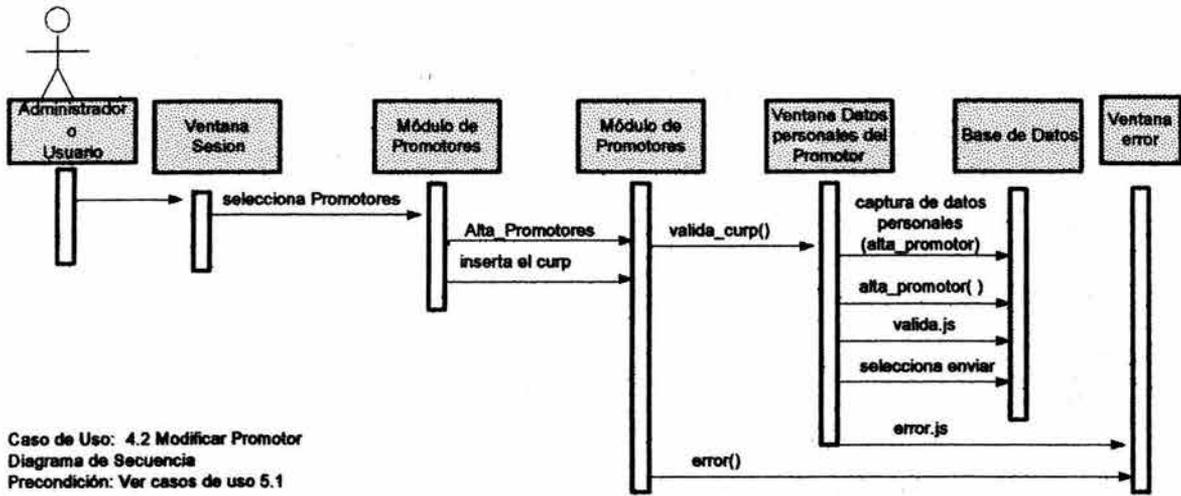


Caso de Uso: 3.4 Sesiones  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1

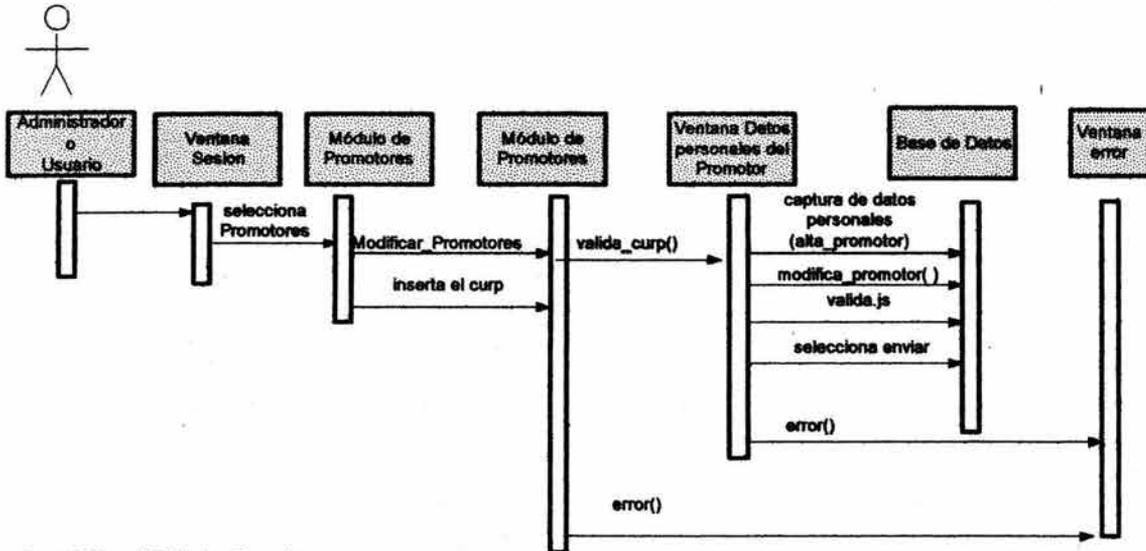


## DIAGRAMA DE SECUENCIA

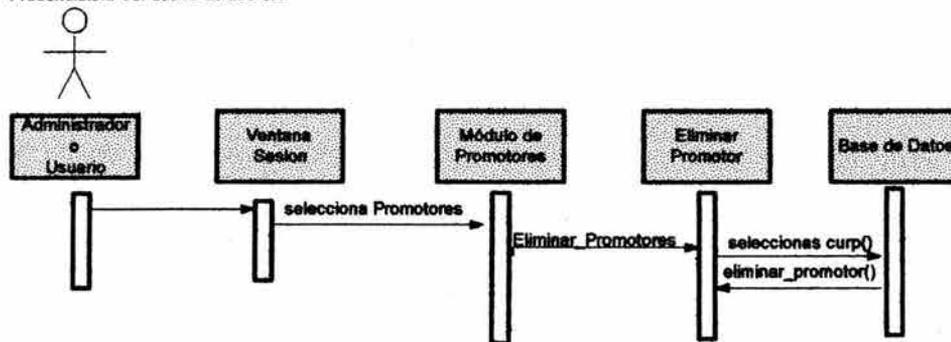
Caso de Uso: 4.1 Alta de Promotor  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1



Caso de Uso: 4.2 Modificar Promotor  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1

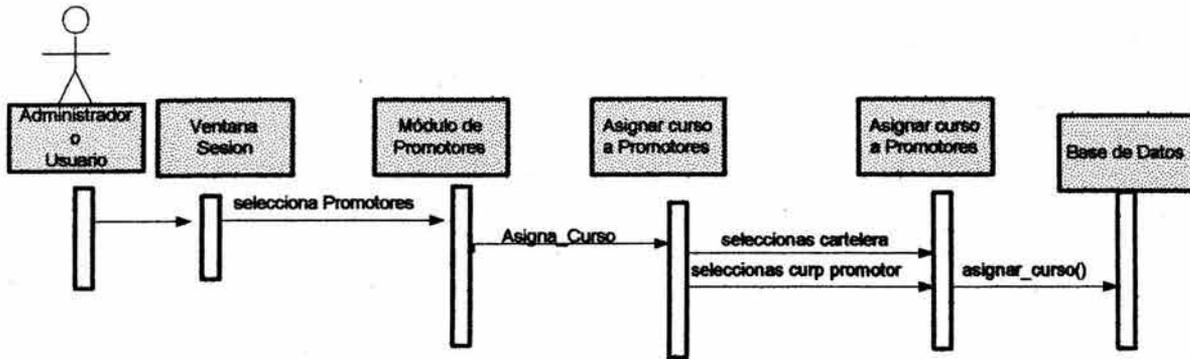


Caso de Uso: 4.3 Eliminar Promotor  
 Diagrama de Secuencia  
 Precondición: Ver casos de uso 5.1

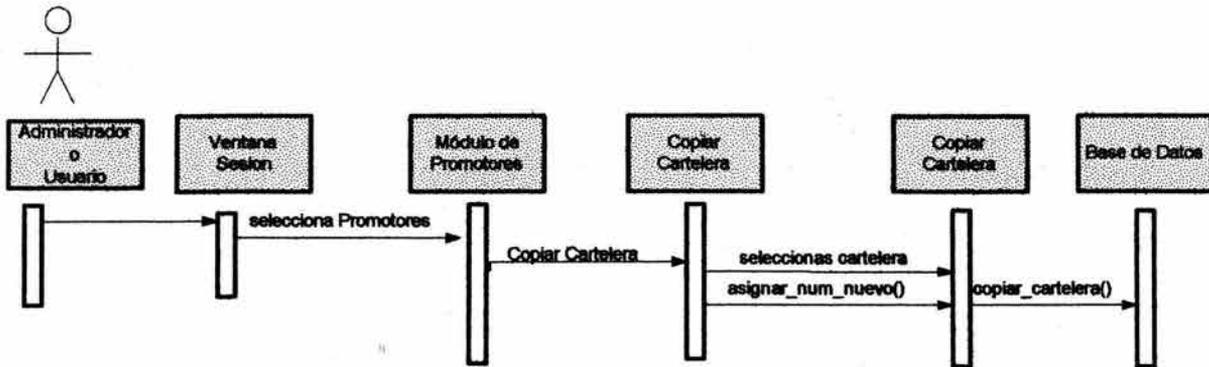


# DIAGRAMA DE SECUENCIA

Caso de Uso: 4.4 Asignar Curso  
Diagrama de Secuencia  
Precondición: Ver uso de casos 5.1



Caso de Uso: 4.5 Copiar Cartelera  
Diagrama de Secuencia  
Precondición: Ver uso de casos 5.1



**MANUAL**

**de usuario**

**SRNCC**

**(Sistema de Red Nacional de Capacitación Cultural)**

## Índice

<b>Introducción .....</b>	<b>1</b>
<b>¿Cómo entrar al sistema SRNCC?.....</b>	<b>2</b>
<b>¿Cómo doy de alta a un instructor? .....</b>	<b>3</b>
<b>¿Cómo modificar la información de un instructor? .....</b>	<b>4</b>
<b>¿Cómo elimino la información de un instructor? .....</b>	<b>5</b>
<b>¿Cómo doy de alta a un coordinador?.....</b>	<b>6</b>
<b>¿Cómo modificar la información de un coordinador? .....</b>	<b>7</b>
<b>¿Cómo elimino la información de un coordinador? .....</b>	<b>8</b>
<b>¿Cómo doy de alta una cartelera?.....</b>	<b>9</b>
<b>¿Cómo puedo modificar una cartelera?.....</b>	<b>11</b>
<b>¿Cómo se elimina una cartelera? .....</b>	<b>12</b>
<b>¿Cómo doy de alta una sesión? .....</b>	<b>13</b>
<b>¿Cómo modificar una sesión? .....</b>	<b>14</b>
<b>¿Cómo dar de alta un promotor?.....</b>	<b>15</b>
<b>¿Cómo modificar la información de un promotor?.....</b>	<b>16</b>
<b>¿Cómo eliminar a un promotor?.....</b>	<b>17</b>
<b>¿Cómo asigno un curso a un promotor?.....</b>	<b>18</b>
<b>¿Cómo cancelo una asignación a un promotor? .....</b>	<b>20</b>
<b>¿Cómo copiar una cartelera .....</b>	<b>21</b>

## INTRODUCCIÓN

El presente documento es una ayuda para el manejo del sistema SRNCC (Sistema de Red Nacional de Capacitación Cultural), el sistema esta basado en diferentes partes entre ellas son Instructores, Coordinadores, Carteleras, Sesiones y Promotores. El manual esta basado en las necesidades de los usuarios y las preguntas más usuales que surgen a partir de ellos.

### ¿Qué es SRNCC?

SRNCC es un sistema que surgió con la necesidad de poder administrar la información de los cursos culturales que se imparten en toda la República Mexicana a través del departamento de Capacitación Cultural en CONACULTA (Consejo Nacional para la Cultura y las Artes).

### ¿Qué es SISTCAP?

SISTCAP es el grupo de trabajo con integrantes universitarios de la facultad de Ciencias de la U.N.A.M. que desarrolla el sistema SRNCC con la más alta tecnología y bajo la metodología TSPi (Team Software Process introduction).

### ¿Cómo está integrado SRNCC?

SRNCC esta conformado actualmente en 5 grandes bloques:

- Instructores.
  - Alta de Instructores.
  - Modificación de Instructores
  - Eliminación de Instructores.
- Coordinadores.
  - Alta de Coordinadores.
  - Modificación de Instructores.
  - Eliminación de Instructores.
- Carteleras.
  - Alta de Carteleras.
  - Modificación de Carteleras.
  - Eliminación de Carteleras.
- Sesiones.
  - Alta de Sesiones.
  - Modificación de Sesiones
- Promotores.
  - Alta de Promotor.
  - Modificación de Promotores.
  - Eliminación de Promotores.
  - Asignación de Cursos a Promotores.
  - Copia de Carteleras.

## ¿Cómo entrar al sistema SRNCC?

Lo primero que necesitas hacer es hablar con el administrador del sistema quien te dará dos claves para entrar al sistema (login y password). Un login es la clave de un nombre y un password es la clave secreta del sistema, éstas dos palabras son únicas y sólo tu debes saberla y nadie más.

Una vez que tienes las claves debes abrir una página de Internet y escribir la siguiente dirección: [http://172.17.9.78/vinculacion/dir\\_cap](http://172.17.9.78/vinculacion/dir_cap) recuerda escribir exactamente como es la dirección. Ya que veas la página debes introducir las claves (login y password). (ver Imagen 1.1).



Imagen 1.1

Después de introducir las claves se abrirá la página de bienvenida. (ver Imagen 1.2).

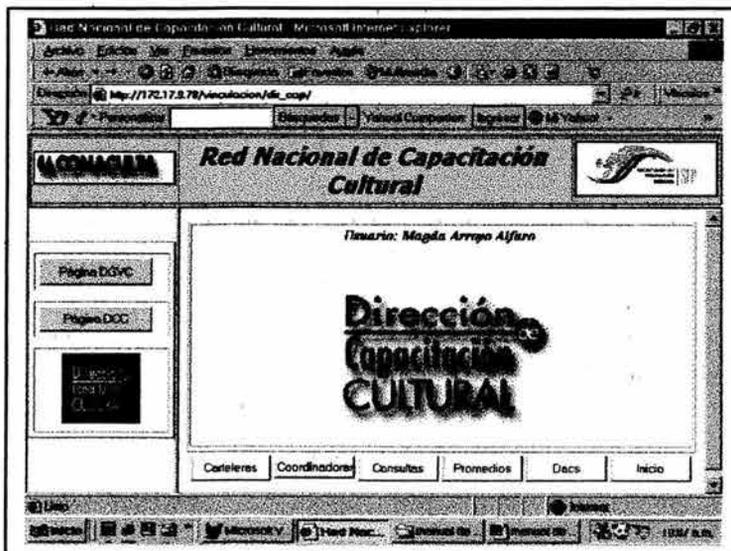
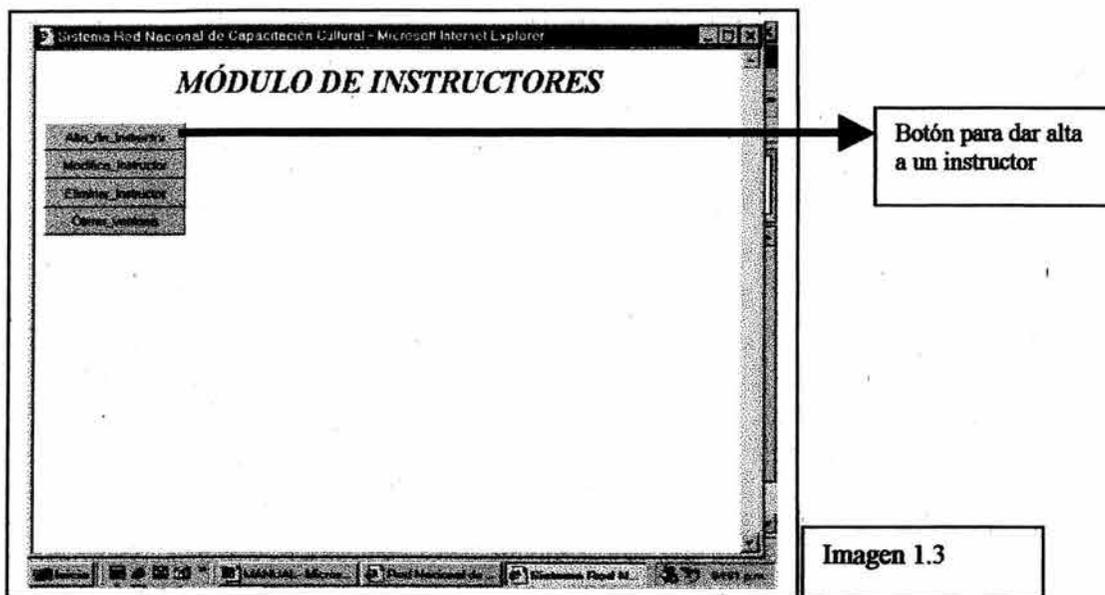


Imagen 1.2

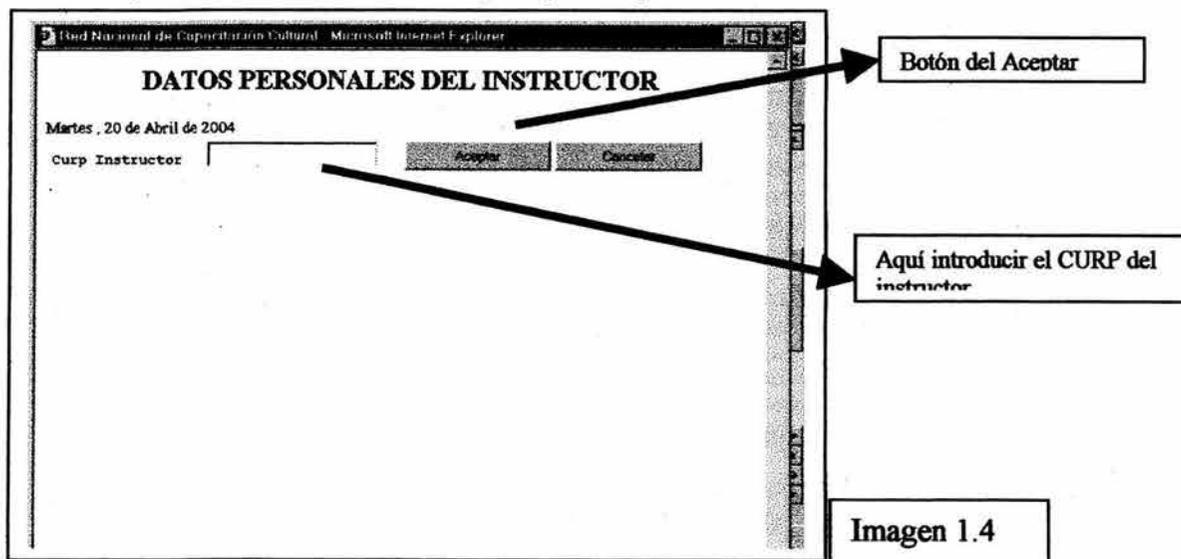
### ¿Cómo doy de alta a un Instructor?

Recuerda que un instructor es aquella persona que imparte los cursos y sesiones así que lo que necesitas es tener su CURP (Clave Única de Registro Personal) y es estar dentro de la página de bienvenida (ver Imagen 1.2) y buscar el botón en la parte inferior que diga Instructores y darle un clic con el botón del mouse y te mostrará una nueva ventana que se llama "Modulo de Instructores" (ver Imagen 1.3).

Encontrarás un botón del lado izquierdo que diga Alta de Instructores, al cual le debes de dar de nuevo un clic para que puedas entrar a otra ventana que se llama DATOS PERSONALES DEL INSTRUCTOR (ver Imagen 1.4.)



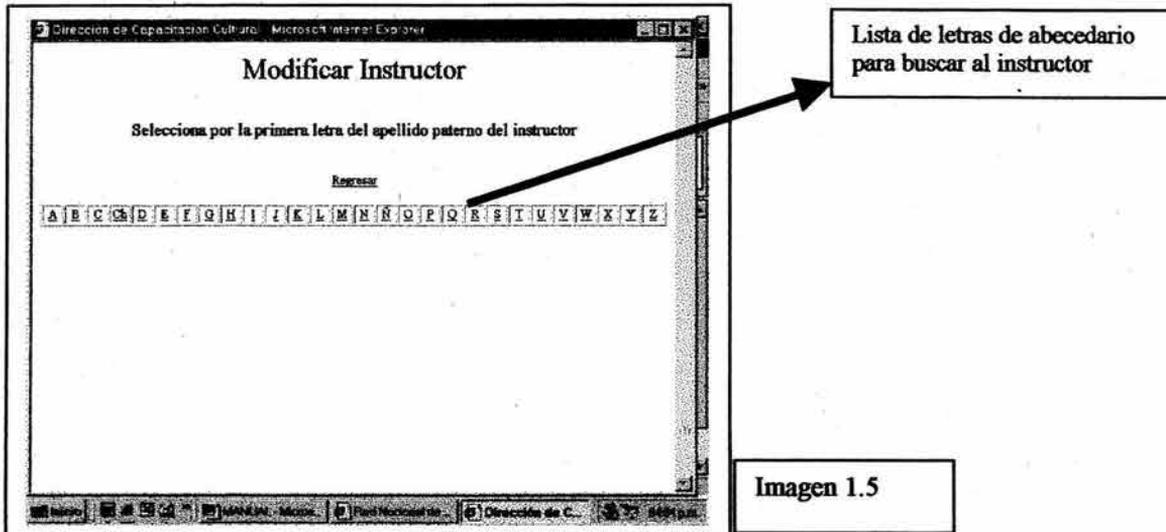
En la ventana de "DATOS PERSONALES DEL INSTRUCTOR" debes introducir el CURP del instructor y darle clic en el botón de Aceptar, para empezar a dar de alta a un instructor.



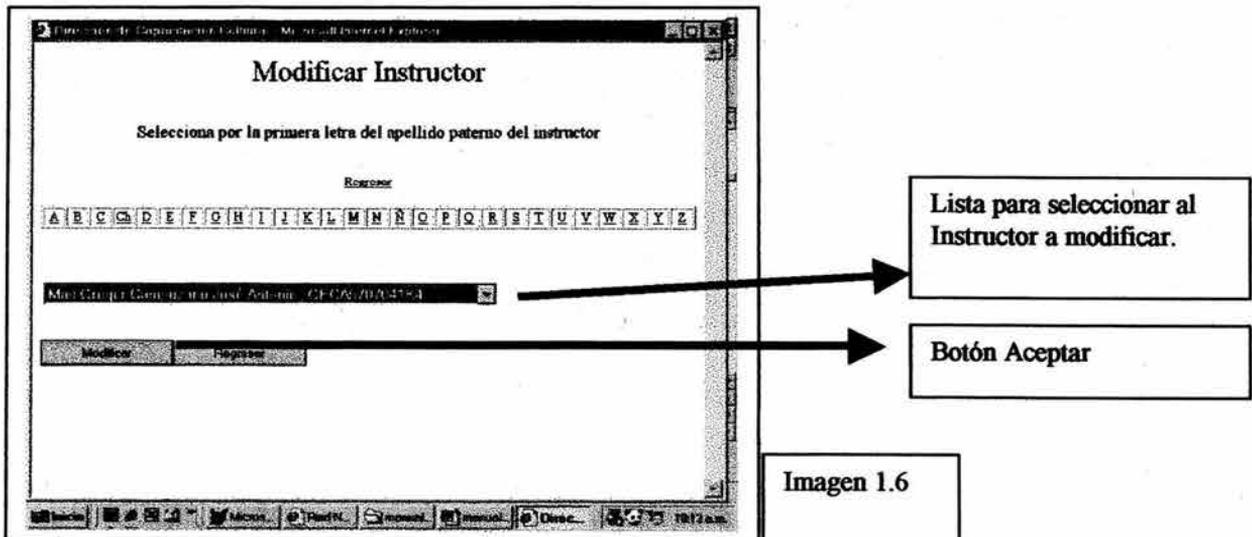
## ¿Cómo modificar la información de un instructor?

Para poder modificar a un instructor tienes que haber llegado hasta la ventana de “MODULO DE INSTRUCTORES”(ver Imagen 1.3). Lo que necesitas es encontrar al Instructor correcto, ahora lo que tienes que hacer es dar clic en el botón que se encuentra del lado izquierdo que se llama “Modifica\_Instructor”.

El sistema debió haber abierto una nueva ventana que se llama “Modificar Instructor” (ver Imagen 1.5) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Instructor para poder iniciar la búsqueda.



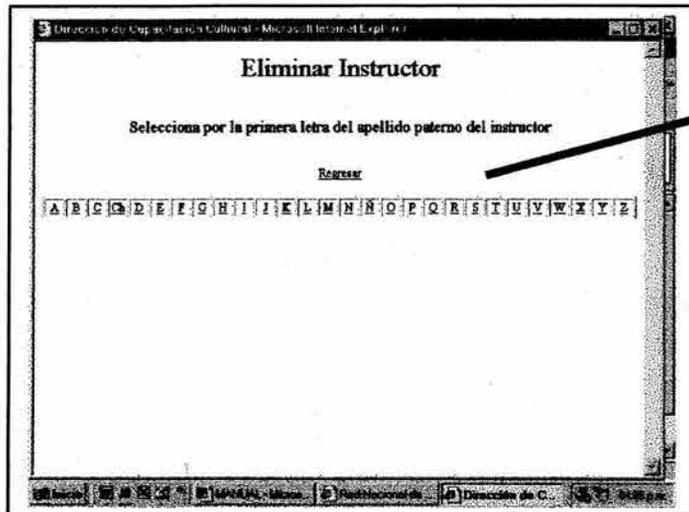
Una vez que se selecciones la primera letra del apellido del Instructor aparece una lista donde seleccionas el nombre del Instructor y le das clic al botón de modificar.(ver Imagen 1.6)



## ¿Cómo elimino la información de un Instructor?

Lo primero que tienes que hacer es encontrar al instructor que deseas eliminar y estar en la ventana de "MODULO DE INSTRUCTORES" (ver Imagen 1.3), ahora solo busca del lado izquierdo el botón de Eliminar\_Instructor.

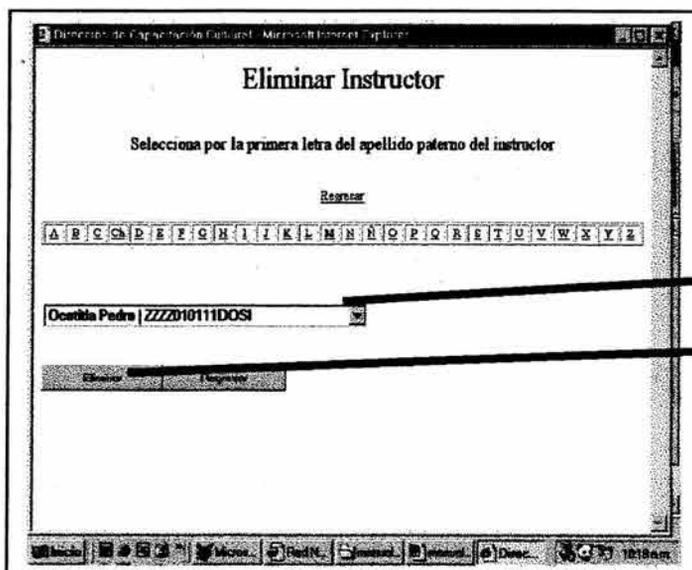
El sistema debió haber abierto una nueva ventana que se llama "Eliminar Instructor" (ver Imagen 1.7) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Instructor para poder iniciar la búsqueda.



Lista de letras del abecedario para buscar al instructor que se desea eliminar

Imagen 1.7

Una vez seleccionado la primera letra del apellido paterno del Instructor aparece una lista donde seleccionas el nombre y le das clic en el botón Eliminar. (ver Imagen 1.8)



Lista para seleccionar al Instructor a modificar.

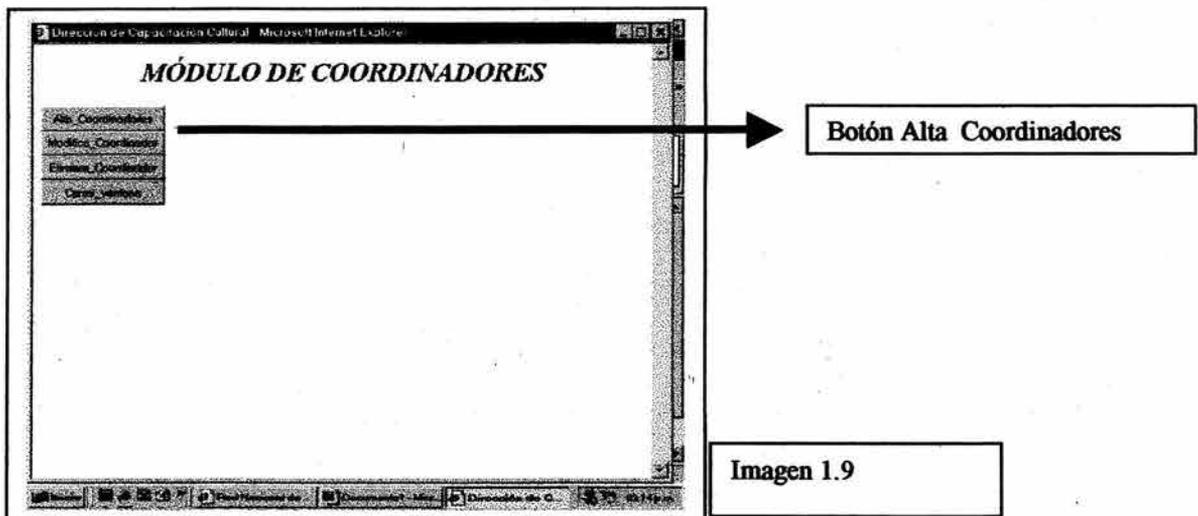
Botón Aceptar

Imagen 1.8

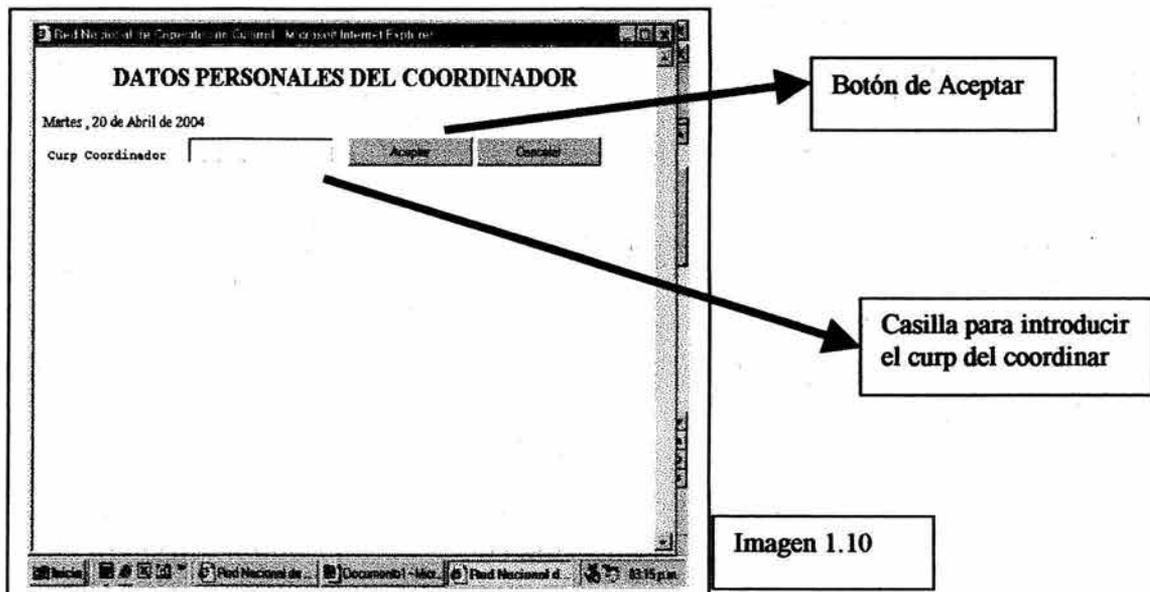
## ¿Cómo doy de alta un coordinador?

Un coordinador es la persona que coordina las carteleras en un curso, así que lo que necesitas es tener su CURP y es estar dentro de la página de bienvenida (ver Imagen 1.2) y buscar el botón en la parte inferior que diga Coordinadores y darle un clic con el botón del mouse y te mostrará una nueva ventana que se llama “MÓDULO DE COORDINADORES” (ver Imagen 1.9).

Encontrarás un botón del lado izquierdo que diga Alta\_Coordinadores, al cual le debes de dar de nuevo un clic para que puedas entrar a otra ventana que se llama DATOS PERSONALES DEL COORDINADOR (ver Imagen 1.10).



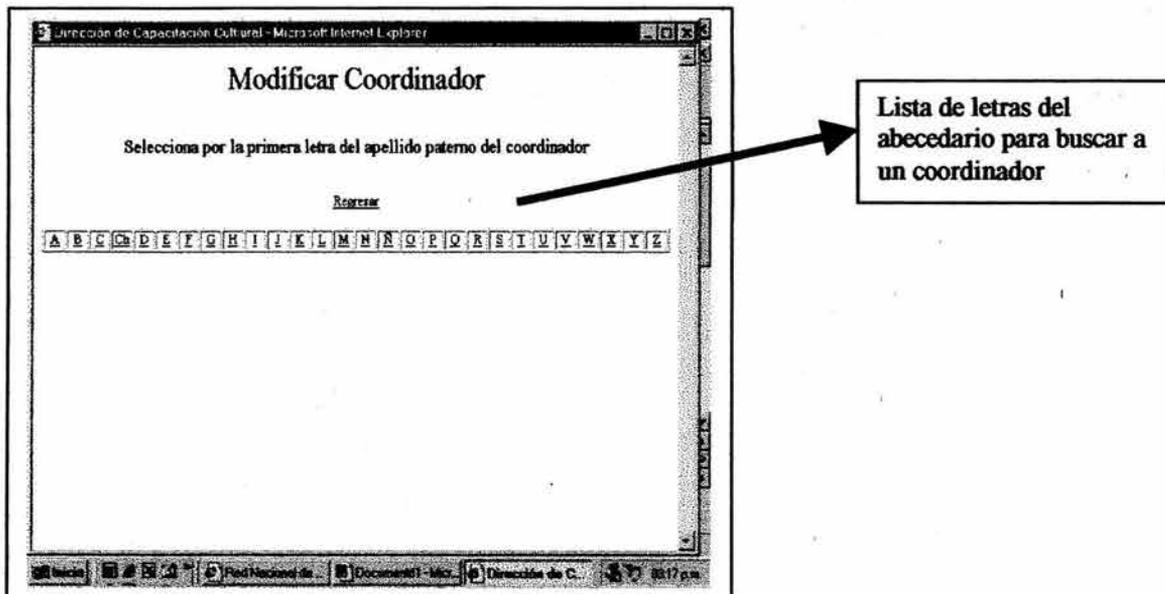
En la ventana de “DATOS PERSONALES DEL COORDINADOR” debes introducir el CURP del coordinador y darle clic en el botón de Aceptar, para empezar a dar de alta a un instructor.



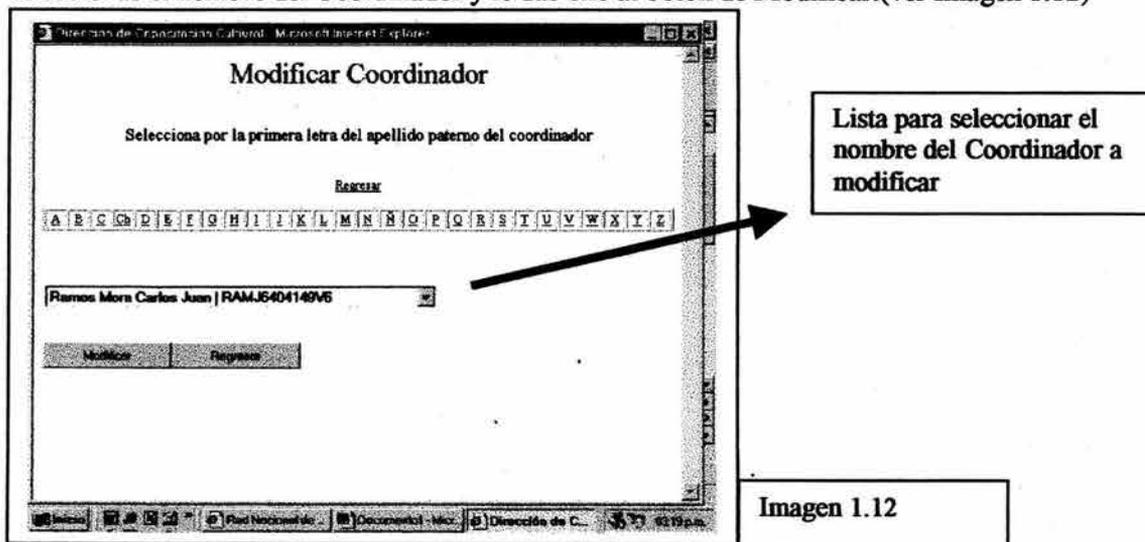
## ¿Cómo modificar la información de un coordinador?

Para poder modificar a un coordinador tienes que haber llegado hasta la ventana de “MODULO DE COORDINADORES”(ver Imagen 1.9). Lo que necesitas es encontrar Coordinador correcto, ahora lo que tienes que hacer es dar clic en el botón que se encuentra del lado izquierdo que se llama “Modifica\_Coordinador”.

El sistema debió haber abierto una nueva ventana que se llama “Modificar Coordinador” (ver Imagen 1.11) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Coordinador para poder iniciar la búsqueda.



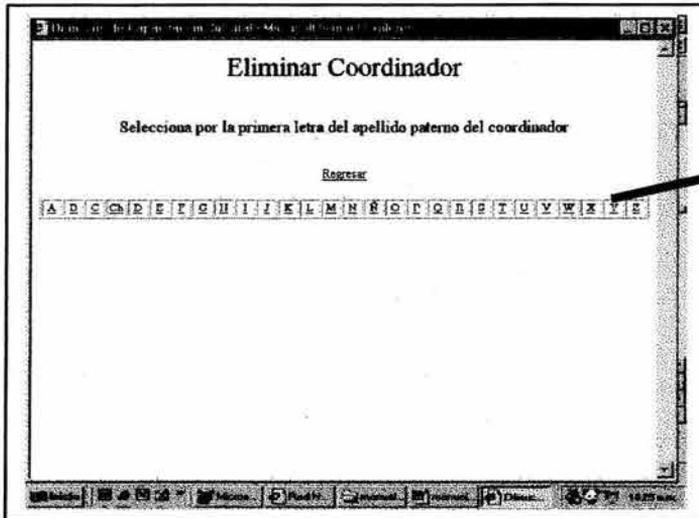
Una vez que se selecciona la primera letra del apellido del Coordinador te mostrará una lista donde seleccionas el nombre del Coordinador y le das clic al botón de Modificar.(ver Imagen 1.12)



### ¿Cómo elimino la información de un Coordinador?

Lo primero que tienes que hacer es encontrar al coordinador que deseas eliminar y estar en la ventana de “MODULO DE COORDINADORES” (ver Imagen 1.9), ahora sólo busca del lado izquierdo el botón de Eliminar\_Coordinador.

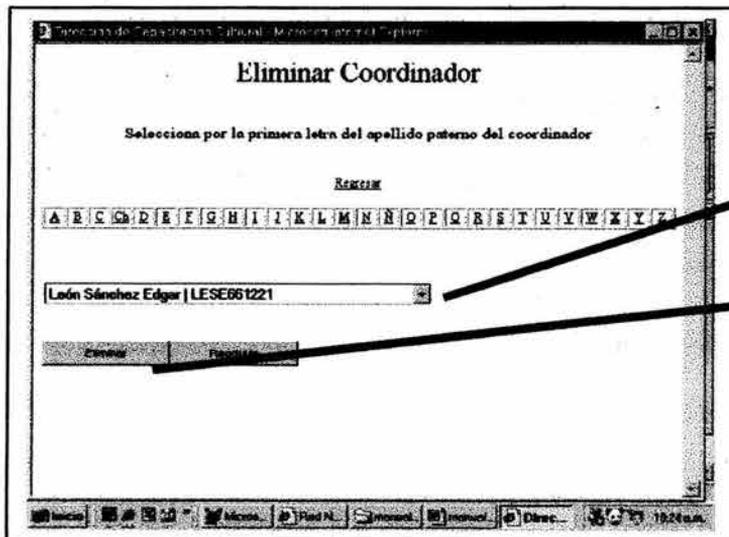
El sistema debió haber abierto una nueva ventana que se llama “Eliminar Coordinador” (ver Imagen 1.13) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Coordinador para poder iniciar la búsqueda.



Lista de letras del abecedario para buscar a un coordinador

Imagen 1.13

Una vez que se seleccionen la primera letra del apellido del Coordinador te mostrará una lista donde seleccionas el nombre del Coordinador y le das clic al botón de Eliminar.(ver Imagen 1.14)



Lista para seleccionar el nombre del Coordinador a modificar

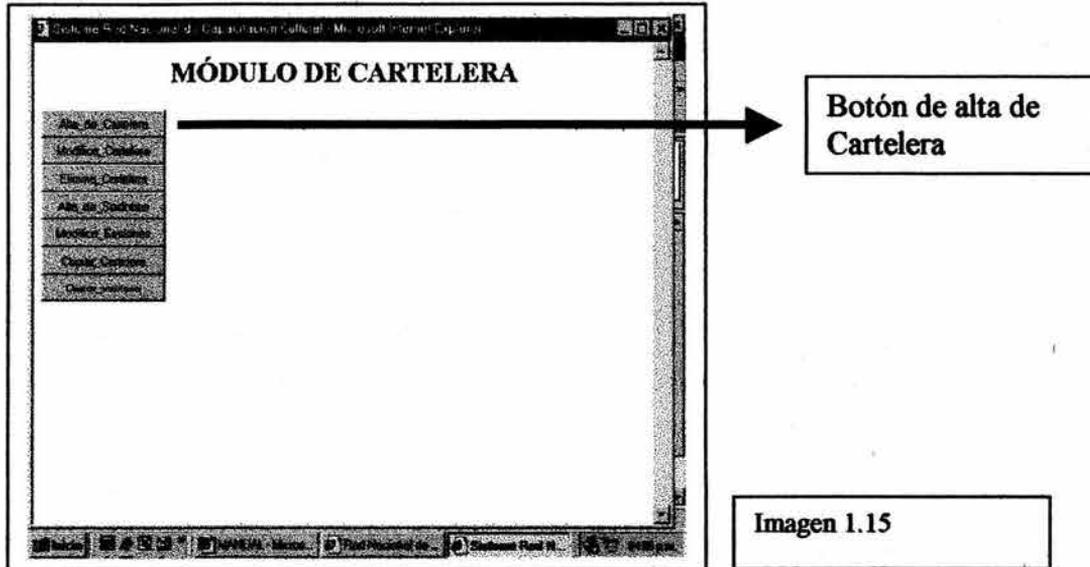
Botón Eliminar

Imagen 1.14

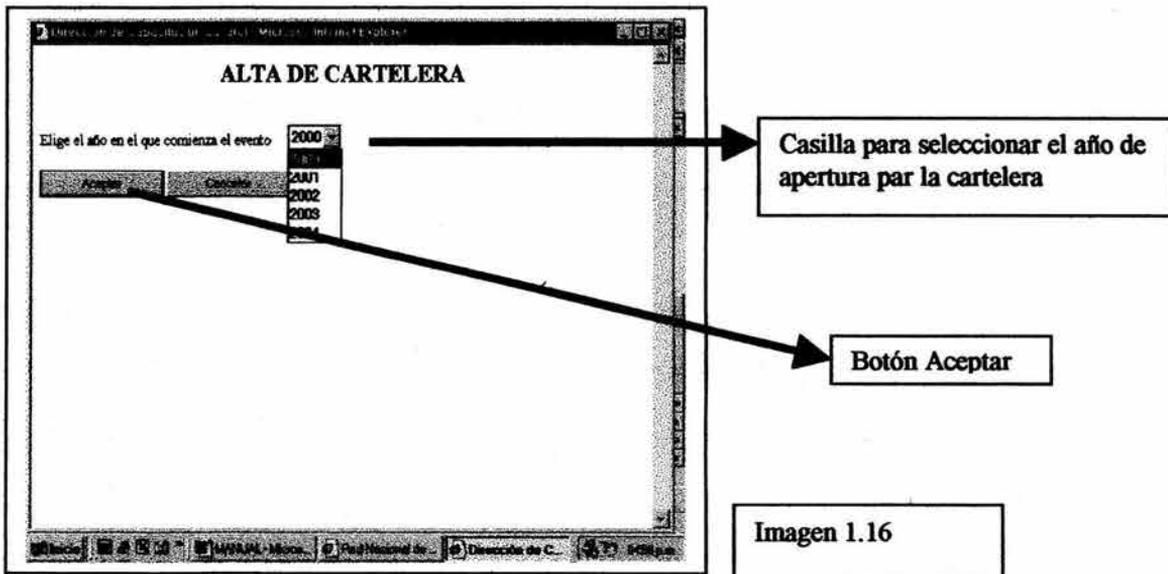
**¿Cómo doy de alta una cartelera?**

Una cartelera es el lugar y datos específicos para iniciar un curso un taller o un módulo. Para dar de alta una cartelera requieres tener al coordinador y al instructor dados de alta antes. Para dar una cartelera debes estar dentro de la página de bienvenida (ver Imagen 1.2) y buscar el botón en la parte inferior que diga Carteleras y darle un clic con el botón del mouse y te mostrará una nueva ventana que se llama “Modulo de Carteleras” (ver Imagen 1.15).

Encontrarás un botón del lado izquierdo que diga Alta\_de\_Carteleras, al cual le debes de dar de nuevo un clic para que puedas entrar a otra ventana que se llama ALTA DE CARTELERA (ver Imagen 1.16).

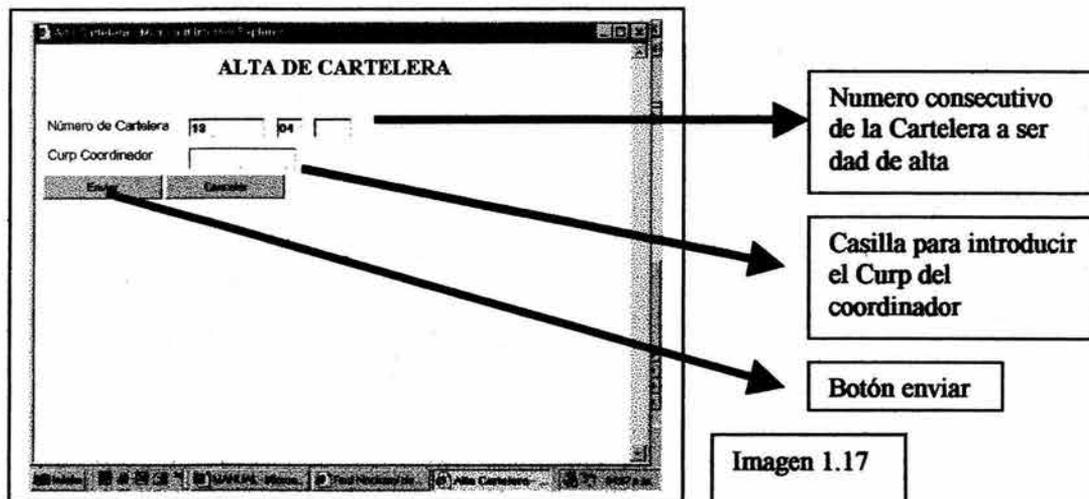


En la ventana ALTA DE CARTELERA debes seleccionar el año en que fue abierta la cartelera y darle clic en el botón de Aceptar (ver Imagen 1.16).

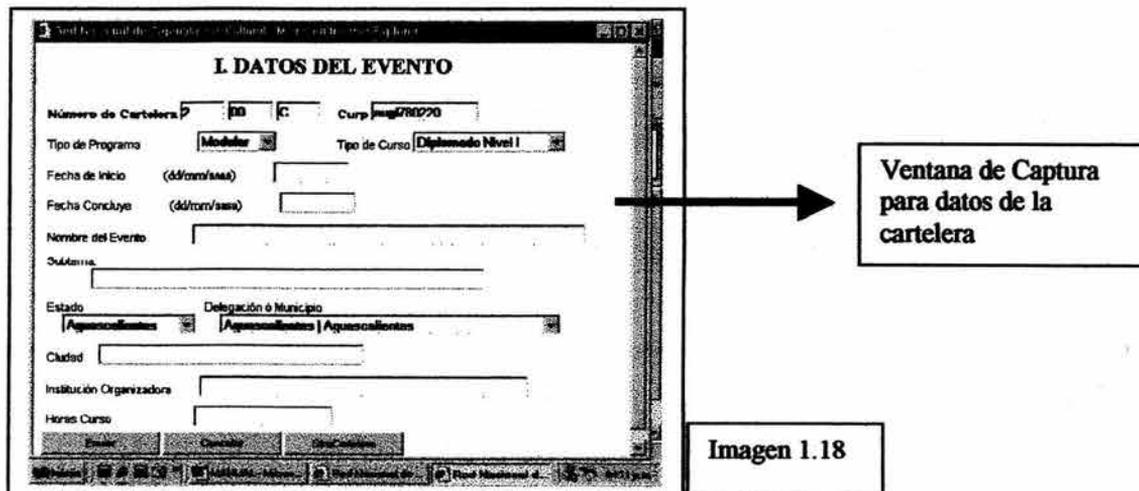


➔ **Continúa ¿Cómo doy de alta una cartelera?**

Una vez que ha sido seleccionado el año abre una nueva ventana ALTA DE CARTELERA en la cual debes introducir el numero consecutivo de cartelera y el CURP del coordinador y darle clic en el botón Enviar (ver Imagen 1.17)



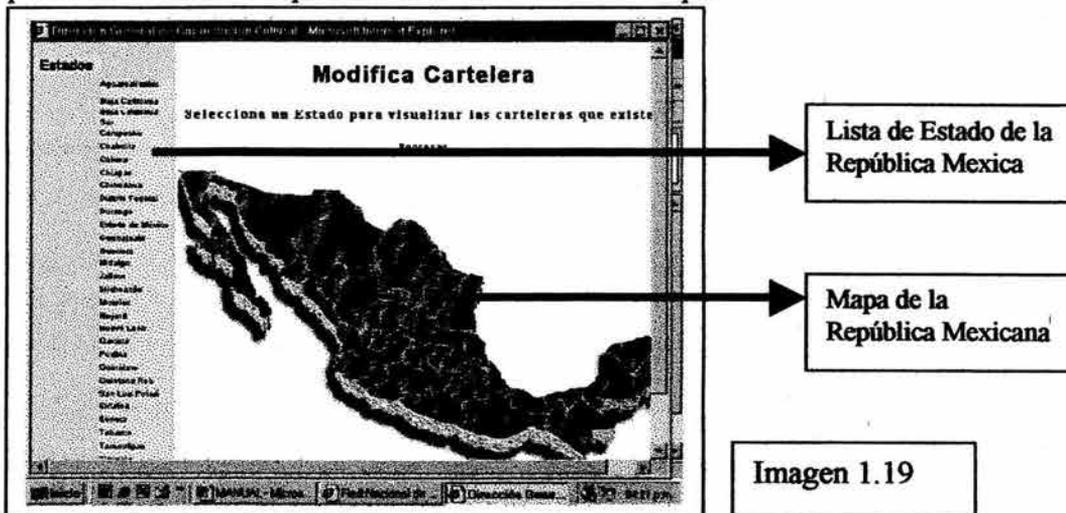
Y aparece una nueva ventana que se llama DATOS DEL EVENTO donde ya se puede capturar la cartelera. (Ver Imagen 1.18)



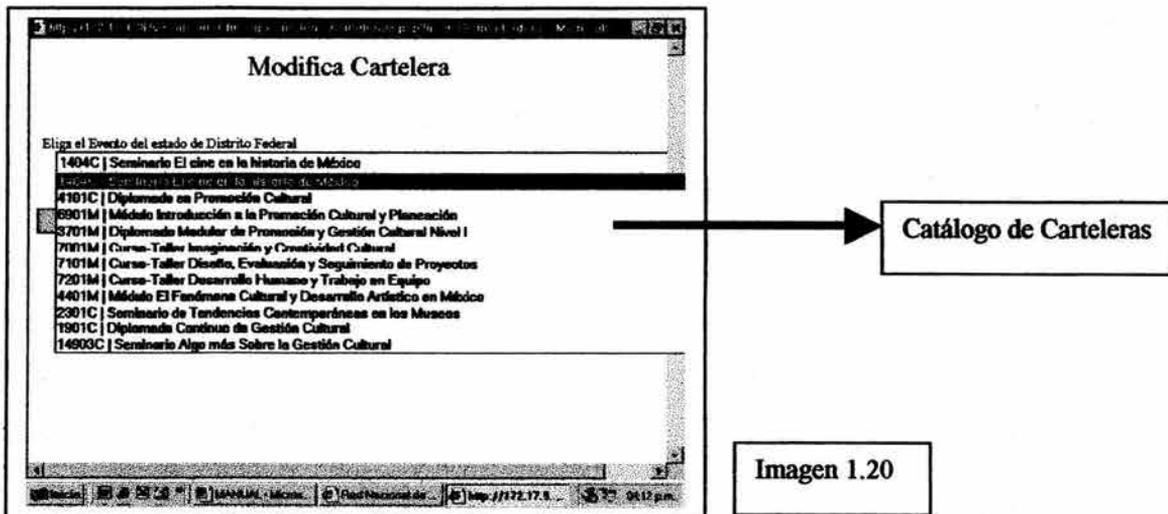
### ¿Cómo puedo modificar una cartelera?

Para poder modificar a una cartelera tienes que haber llegado hasta la ventana de “MODULO DE CARTELERAS”(ver Imagen 1.15). Lo que necesitas Para modificar una cartelera de entre los datos que das de alta a una cartelera necesitas saber en que estado fue dada de alta, ahora lo que tienes que hacer dar clic en el botón que se encuentra del lado izquierdo que se llama “Modificar\_Cartelera”.

El sistema debió haber abierto una nueva ventana que se llama “Modifica Cartelera” (ver Imagen 1.19) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa.



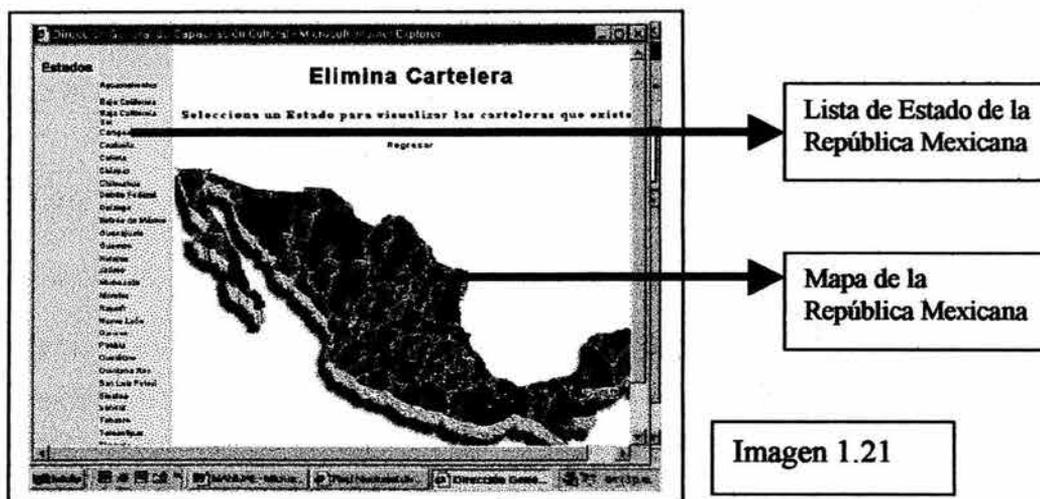
Después se abre una nueva ventana la cual debes seleccionar la cartelera que deseas modificar y darle clic en el botón de Modificar(ver Imagen 1.20).



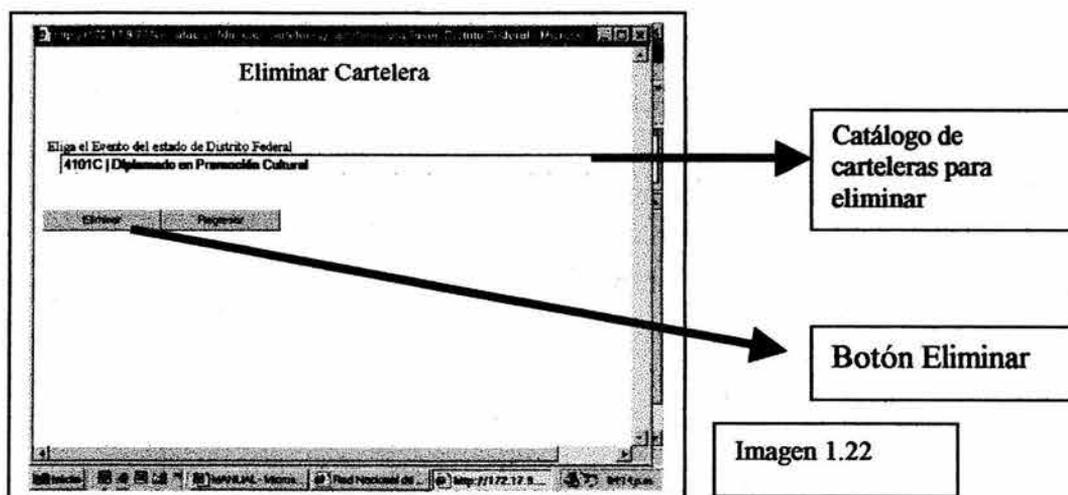
### ¿Cómo se elimina una cartelera?

Para poder eliminar a una cartelera tienes que haber llegado hasta la ventana de “MODULO DE CARTELERAS”(ver Imagen 1.15). Lo que necesitas Para eliminar una cartelera de entre los datos que das de alta a una cartelera necesitas saber en que estado fue dada de alta, ahora lo que tienes que hacer dar clic en el botón que se encuentra del lado izquierdo que se llama “Eliminar\_Cartelera”.

El sistema debió haber abierto una nueva ventana que se llama “Eliminar Cartelera” (ver Imagen 1.21) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa.



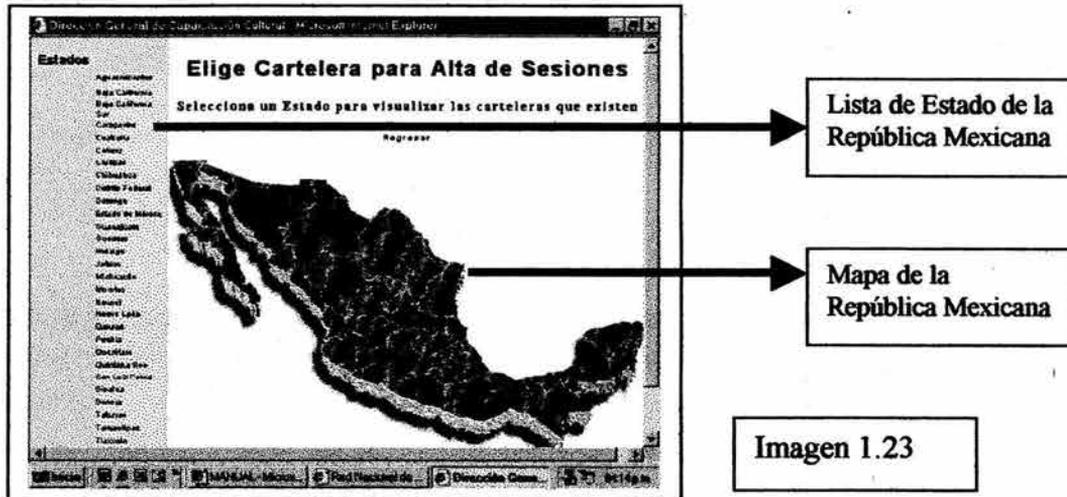
Seleccionar la cartelera que se desea eliminar y dar clic en el botón Eliminar. (ver Imagen 1.22)



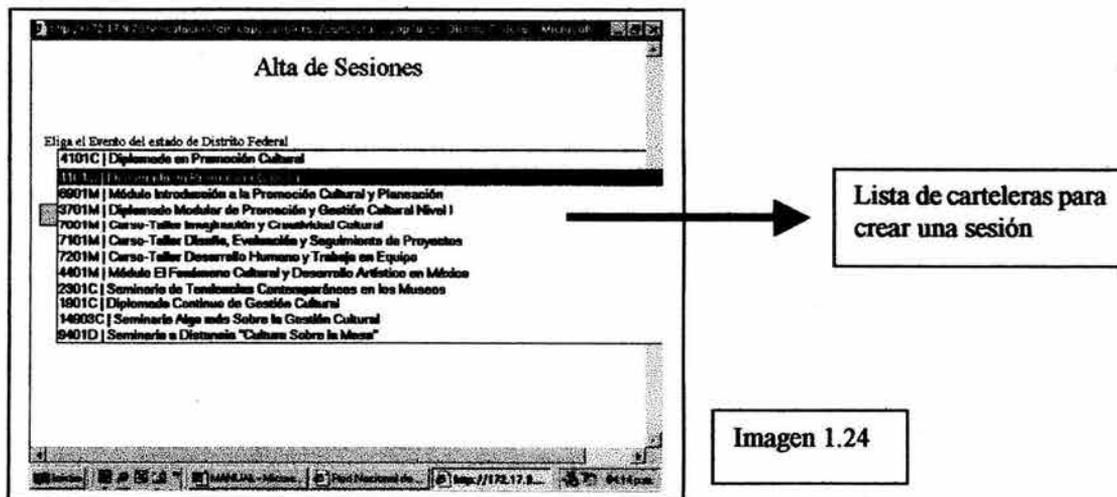
### ¿Cómo doy de alta una sesión?

Para poder eliminar a una cartelera tienes que haber llegado hasta la ventana de “MODULO DE CARTELERAS”(ver Imagen 1.15).Lo que necesitas es ya tener la cartelera para dar de alta una sesión.

El sistema debió haber abierto una nueva ventana que se llama “Elige Cartelera para Alta de Sesiones” (ver Imagen 1.23) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa.



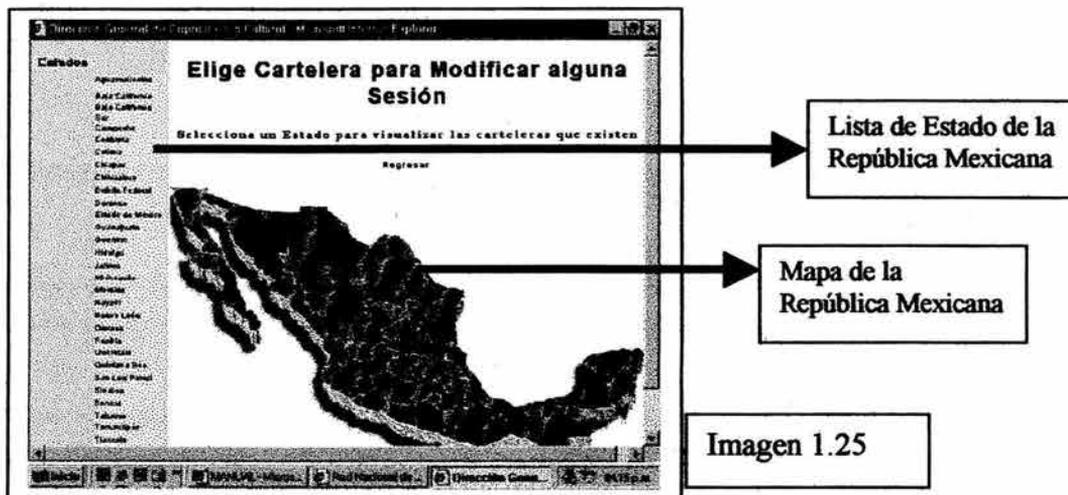
Ahora lo que necesitas es seleccionar el nombre de la cartelera y dar clic en el botón enviar para iniciar el alta de una cartelera donde va serle asignada una número de sesión (ver Imagen 1.24).



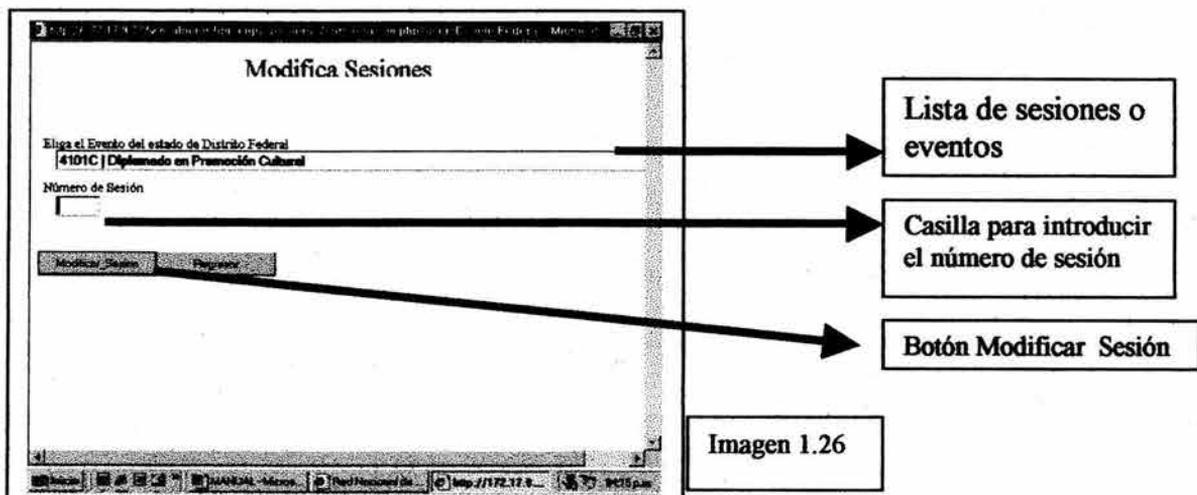
## ¿Cómo modificar una sesión?

Para poder modificar a una sesión tienes que haber llegado hasta la ventana de “MODULO DE CARTELERAS”(ver Imagen 1.15). Lo que necesitas Para modificar una sesión de entre los datos que das de alta a una cartelera necesitas saber en que estado fue dada de alta, ahora lo que tienes que hacer dar clic en el botón que se encuentra del lado izquierdo que se llama “Modificar\_Sesión”.

El sistema debió haber abierto una nueva ventana que se llama “Elige Cartelera para Modificar alguna Sesión” (ver Imagen 1.25) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa



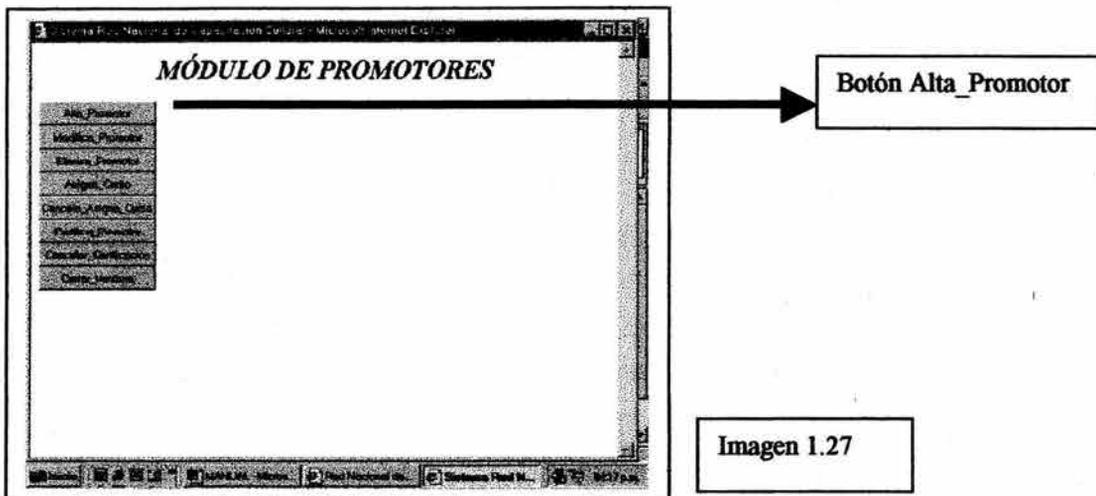
Después se abre una nueva ventana llamada “Modifica Sesiones” la cual debes seleccionar el evento que deseas modificar e introducir el número de sesión y darle clic en el botón de Modificar\_Sesión (ver Imagen 1.26).



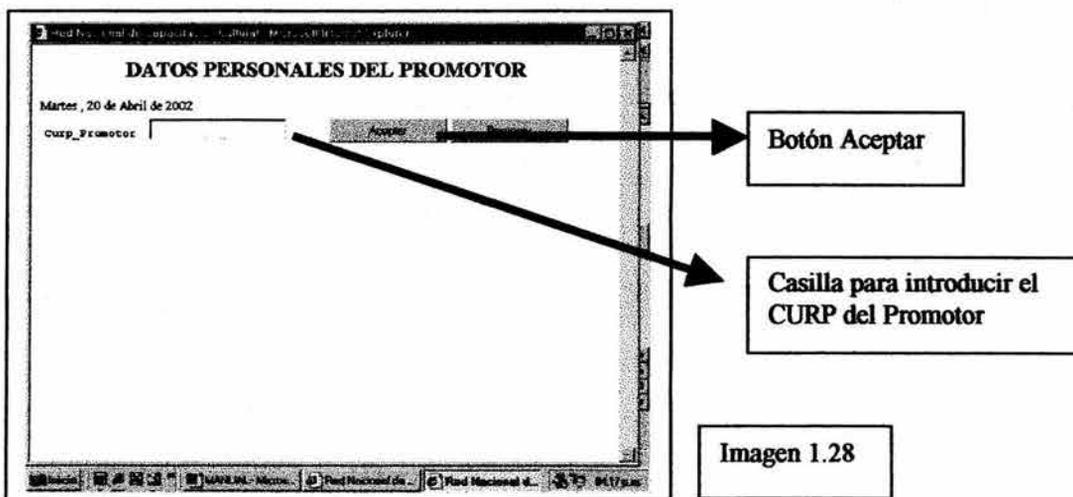
### ¿Cómo dar de alta a un promotor?

Recuerda que un promotor es aquella persona que toma los cursos así que lo que necesitas es tener su CURP (Clave Única de Registro Personal) y estar dentro de la página de bienvenida (ver Imagen 1.2) y buscar el botón en la parte inferior que diga Promotores y darle un clic con el botón del mouse y te mostrará una nueva ventana que se llama "Módulo de Promotores" (ver Imagen 1.27).

Encontrarás un botón del lado izquierdo que diga Alta\_Promotor, al cual le debes de dar de nuevo un clic para que puedas entrar a otra ventana que se llama DATOS PERSONALES DEL COORDINADOR (ver Imagen 1.28.)



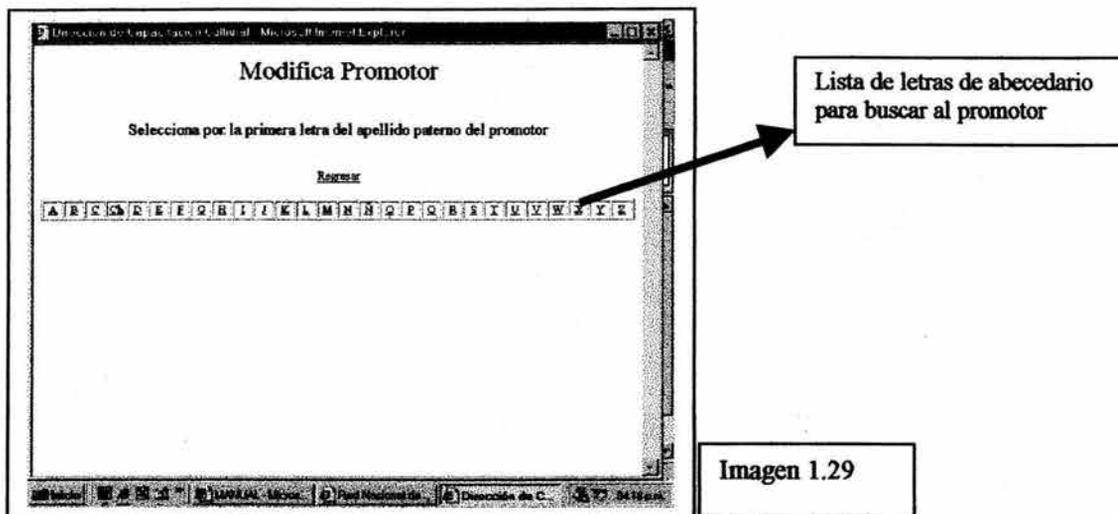
En la ventana de "DATOS PERSONALES DEL PROMOTOR" debes introducir el CURP del instructor y darle clic en el botón de Aceptar, para empezar a dar de alta a un coordinador.



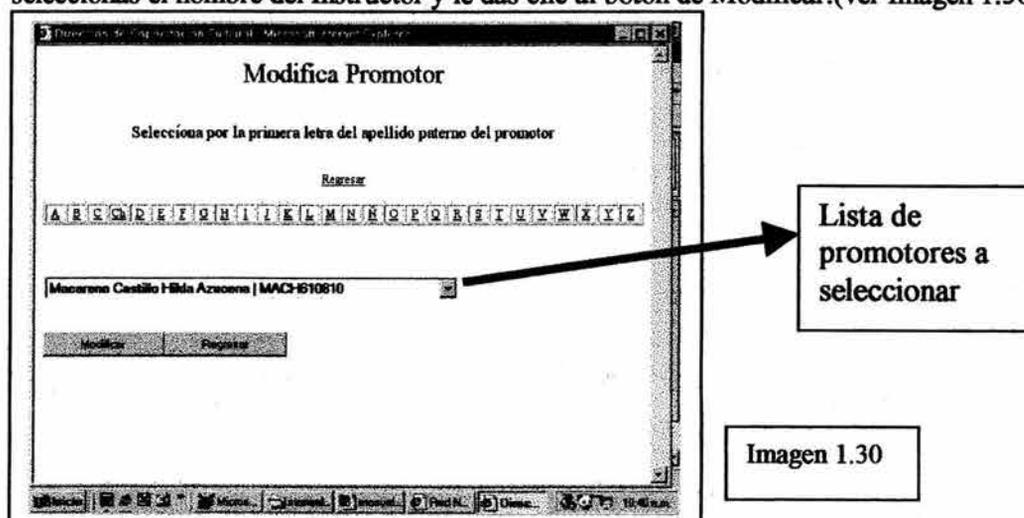
## ¿Cómo modificar la información de un promotor?

Para poder modificar a un instructor tienes que haber llegado hasta la ventana de “MODULO DE PROMOTORES”(ver Imagen 1.27). Lo que necesitas es encontrar al Instructor correcto, ahora lo que tienes que hacer es dar clic en el botón que se encuentra del lado izquierdo que se llama “Modifica\_Promotor”.

El sistema debió haber abierto una nueva ventana que se llama “Modifica Promotor” (ver Imagen 1.29) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Instructor para poder iniciar la búsqueda.



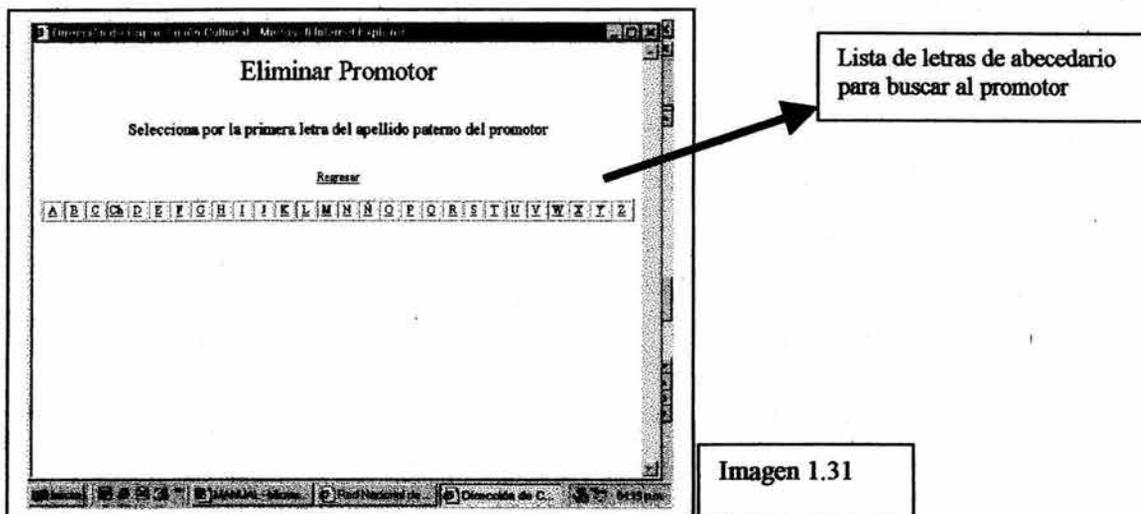
Una vez que se selecciona la primera letra del apellido del Instructor aparece una lista donde seleccionas el nombre del Instructor y le das clic al botón de Modificar.(ver Imagen 1.30)



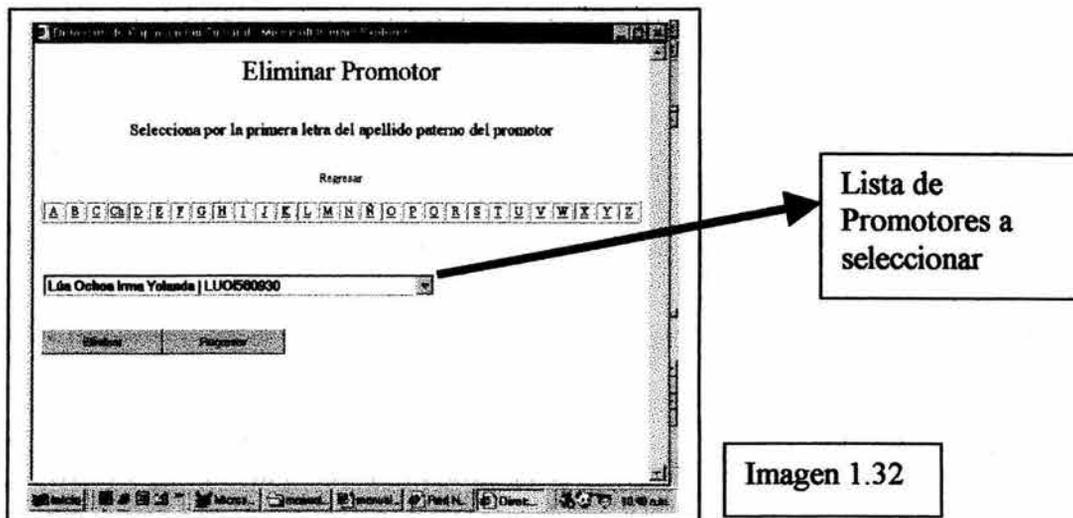
### ¿Cómo eliminar a un Promotor?

Lo primero que tienes que hacer es encontrar al Promotor que deseas eliminar y estar en la ventana de “MODULO DE PROMOTORES” (ver Imagen 1.27), ahora solo busca del lado izquierdo el botón de Elimina\_Promotor.

El sistema debió haber abierto una nueva ventana que se llama “Eliminar Promotor” (ver Imagen 1.31) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Promotor para poder iniciar la búsqueda.



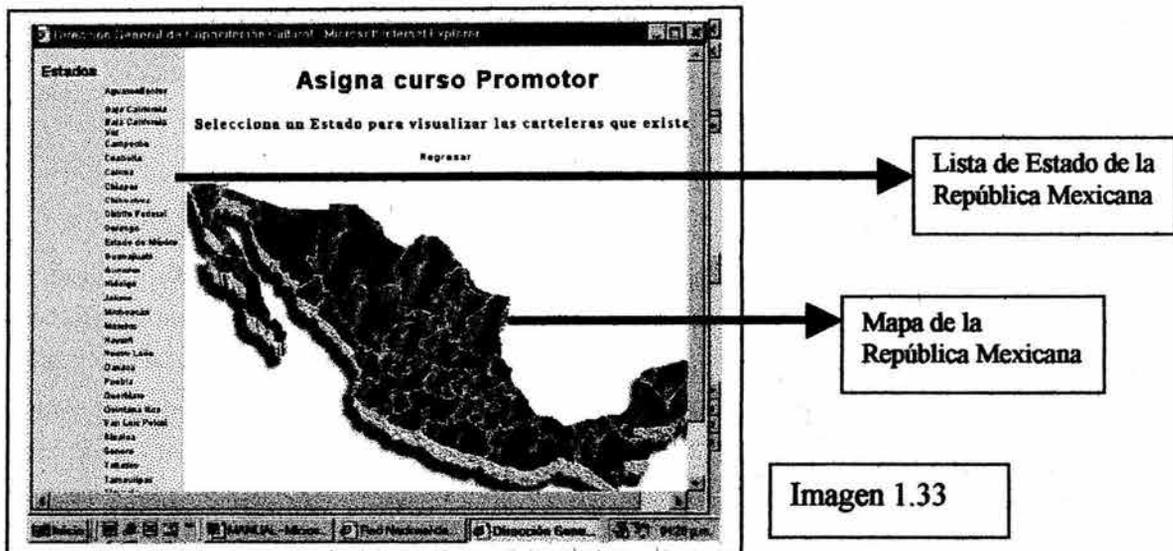
Una vez seleccionado la primera letra del apellido paterno del Promotor aparece una lista donde seleccionas el nombre y le das clic en el botón Eliminar. (ver Imagen 1.32)



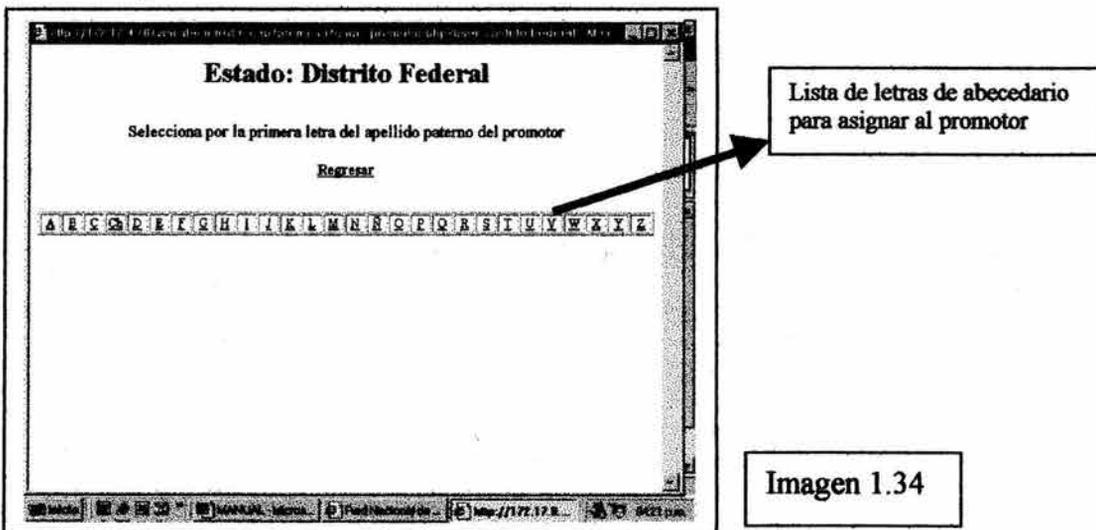
### ¿Cómo asigno un curso a un Promotor?

Lo primero que tienes que hacer es encontrar la cartelera (o curso) y estar en la ventana de "MODULO DE PROMOTORES" (ver Imagen 1.27), ahora solo busca del lado izquierdo el botón de Asigna\_Curso. (ver Imagen 1.28.).

El sistema debió abrir una ventana que se llama "Asigna Curso Promotor" (ver Imagen 1.33) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa

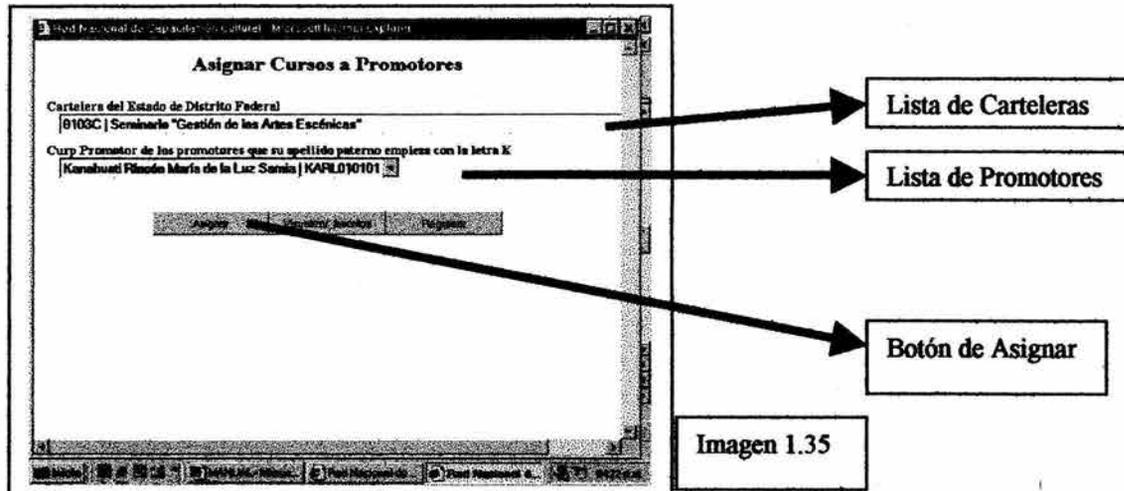


El sistema debió haber abierto una nueva ventana que hace referencia al estado que seleccionaste (ver Imagen 1.34) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Promotor al que quieres asignar el curso.

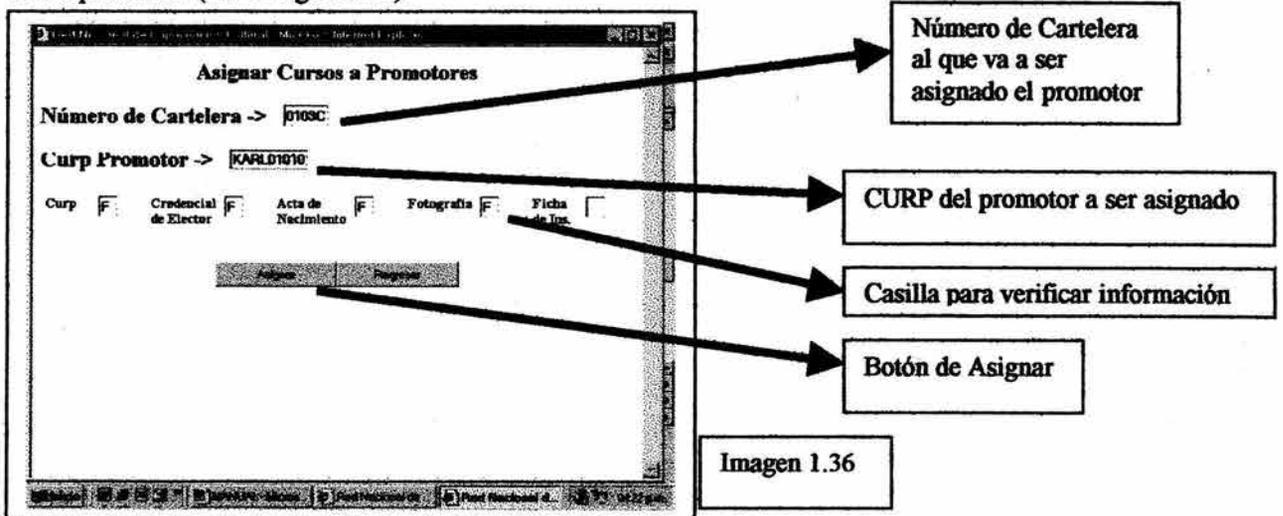


→ Continúa de ¿Cómo asigno un curso a un promotor?

Ahora aparece una ventana donde te da una lista de carteleras de ese estado y una lista de Promotores que corresponde su apellido paterno con la primera letra que seleccionaste. Y una vez que los selecciones a ambos le das clic en el botón Asignar (ver Imagen 1.35).



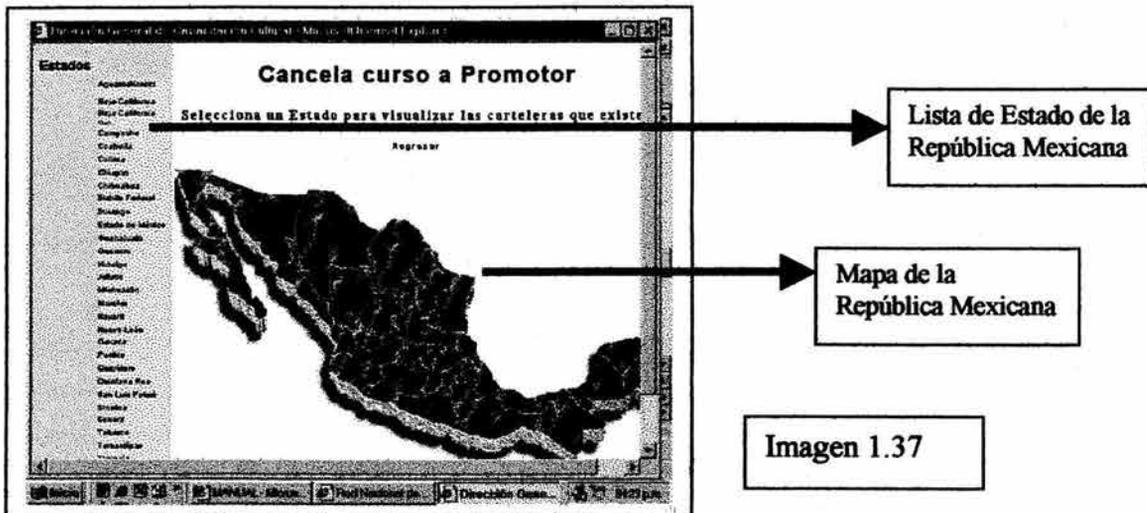
Ahora aparece otra ventana "Asigna Curso a Promotor" donde lo que tienes que hacer es poner una F si le falta documentación y una T si la tiene completa para verificar la información en papel correspondiente (ver Imagen 1.36)



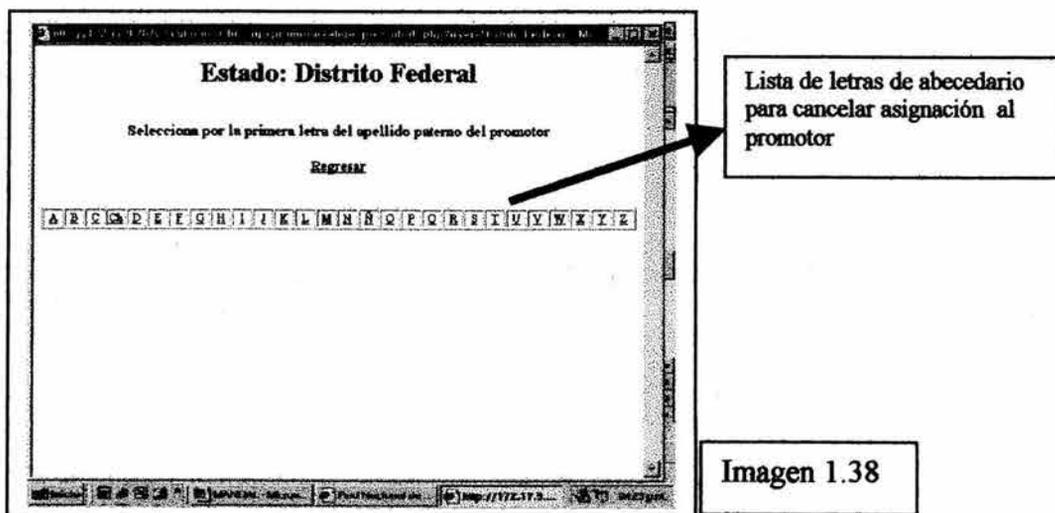
### ¿Cómo cancelo una asignación a un promotor

Lo primero que tienes que hacer es encontrar la cartelera en la que esta asignado el Promotor que deseas eliminar en su asignación y estar en la ventana de “MODULO DE PROMOTORES” (ver Imagen 1.27), ahora solo busca del lado izquierdo el botón de Cancela\_Asignación.

El sistema debió abrir una ventana que se llama “Cancela curso a Promotor” (ver Imagen 1.37) y te muestra un mapa de la República Mexicana en donde seleccionas ya sea por la lista del lado izquierdo el estado o sobre el mapa para visualizar las carteleras por el estado seleccionado.

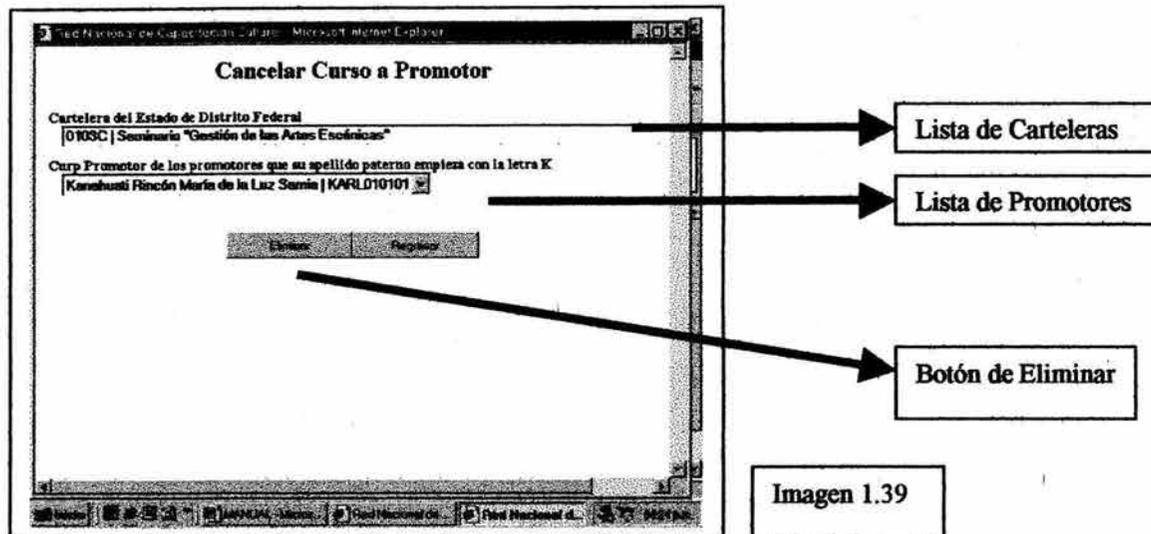


El sistema debió haber abierto una nueva ventana que hace referencia al estado que seleccionaste (ver Imagen 1.38) y te muestra una línea de letras del abecedario. Lo que tienes que hacer es dar clic a la primera letra de apellido paterno del Promotor al que quieres cancelar la asignación.



### Continúa de cancela asignación a un promotor

Ahora aparece una ventana donde te da una lista de carteleras de ese estado y una lista de Promotores que corresponde su apellido paterno con la primera letra que seleccionaste. Y una vez que los selecciones a ambos le das clic en el botón Eliminar (ver Imagen 1.39).



### ¿Cómo copiar una cartelera?

Suele pasar que el mismo grupo de promotores toma varias Carteleras por lo cual sería muy lento tener que capturar a las mismas personas a cada Cartelera. Lo primero que debes hacer es estar en el "MODULO DE PROMOTORES"(ver Imagen 1.27). Lo que necesitas es encontrar cartelera correcta, ahora lo que tienes que hacer dar clic en el botón que se encuentra del lado izquierdo que se llama "Copiar Cartelera".

Ahora el sistema abrirá otra ventana que se llama "Copiar Cartelera" y donde buscas la cartelera a copiar después el sistema le asigna un número consecutivo y le debes de dar clic en el botón Copiar Cartelera (ver Imagen 1.40)

