

94  
29



UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

FACULTAD DE INGENIERIA

DIVISION DE INGENIERIA ELECTRICA

DISEÑO Y DESARROLLO DEL SISTEMA  
AUTOMATIZADO PARA LA ADMINISTRACION  
HOSPITALARIA DEL I. S. S. S. T. E. (SAAH )

T E S I S

PARA OBTENER EL TITULO DE:  
INGENIERO EN COMPUTACION  
P R E S E N T A N :

CLAUDIA	PANIAGUA	DOMINGUEZ
GUSTAVO	FRANCO	VALENZUELA
ALFREDO	TORRES	VAZQUEZ
ROBERTO	VALDES	ORTIZ

DIRECTOR DE TESIS:

M. I. LAURO SANTIAGO CRUZ



MEXICO, D. F.

1996

TESIS CON  
FALLA DE ORIGEN

TESIS CON  
FALLA DE ORIGEN



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



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

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

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

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

## **AGRADECIMIENTOS**

A nuestra máxima casa de estudios, la Universidad Nacional Autónoma de México.

A la Facultad de Ingeniería y sus profesores, por los conocimientos adquiridos de ellos.

Al Instituto de Seguridad y Servicio Social de los Trabajadores del Estado (I.S.S.T.E.), por el apoyo y facilidades otorgadas para la realización del presente trabajo.

Al Ing. Miguel Eduardo González Cárdenas, a la Ing. Elsa Elena Barón Mayo y al Laboratorio de Computadoras y Programación por las facilidades brindadas para la realización de este trabajo.

A Claudia Erika, por su apoyo incondicional.

Un agradecimiento especial para nuestro asesor de tesis M.I. Lauro Santiago Cruz.

A todos ellos, **GRACIAS.**

Claudia, Alfredo, Gustavo y Roberto.

A mis padres les dedico este trabajo de todos corazón y de quienes estaré agradecida por su apoyo y confianza durante todo este tiempo. Gracias por darme la vida.

Recuerden que su esfuerzo no fue en vano.

A Armando y Conny por los consejos recibidos durante mi carrera.  
Gracias por ser mi hermano y tu mi gran cuñada.

Saben ....son mi ejemplo a seguir.

A mis tíos y primos que siempre me alentaron para terminar la carrera y este trabajo.

A mis amigos quienes me apoyaron y creyeron en mí.

**GRACIAS, Claudia**

A mis padres y hermanos, por el amor que les tengo.

Alfredo

A Dios que me dió la vida  
le dedico este trabajo con todo cariño y respeto  
porque has sido tú quien me ha llevado a conseguirlo  
iluminándome y protegiéndome cada día...Gracias Padre.

A mi compañera de cada día  
A la persona que me apoyó y me motivó  
a sacar adelante cada día de mi carrera  
A la persona que me ha entregado todo su cariño  
A la persona que día a día tendré a mi lado  
Angelica te dedico este trabajo con todo mi amor.

A mis hijos Gustavo y Alan por ser gran parte de  
mi corazón y por ser una bendición que Dios me dió  
Los amo con todo mi alma  
Protegelos siempre Señor.

Dedico con todo mi amor y cariño  
a las personas que desde el primer día de mi vida  
supieron darme un gran ejemplo, apoyo, cuidado,  
paciencia y cariño con todo su corazón  
Papá, Mamá los llevo muy presentes  
en todo momento de mi vida.  
Los llevo siempre...en mi corazón.

A mis hermanos Olga, Alma y Alfredo  
a quienes siempre les he de agradecer  
su apoyo y estímulo para el logro de mis  
anhelos y por ser los mejores hermanos  
del mundo...Los quiero mucho.

**GRACIAS, Gustavo**

A mi esposa Silvia, por su amor, apoyo y comprensión en todo momento.

A mis hijas Ana Silvia y Laura Ivonne, todo este esfuerzo es por y para ellas.

A mis padres Rafael y Ma. Elena, por el amor y apoyo que siempre me han brindado.

A mis hermanos Eduardo y Ma. Elena.

A mi tío José, por su interés para la conclusión de ésta etapa de mi educación.

A mis compañeros de tesis, Claudia, Alfredo y Gustavo, por el apoyo, conocimientos y esfuerzo para lograr este objetivo.

Al Ing. Víctor Escalante Hutrón, por todo el apoyo que siempre me ha brindado.

**A TODOS ELLOS, GRACIAS**

**Roberto**

## **ÍNDICE**

• <b>Introducción</b>	<b>1</b>
• <b>Capítulo 1 Fase de Análisis del S.A.A.H.</b>	<b>4</b>
• Definición del Sistema	4
• Definición del Problema	4
• Justificación del Sistema	5
• Objetivos	6
• Alcances	6
• Consulta Externa	6
• Admisión Hospitalaria	7
• Relaciones Públicas	7
• Hospitalización	7
• Farmacia	7
• Beneficios que se Proporcionarán	8
• Riesgos	9
• Estrategia de Solución	9
• Plan del Proyecto	13
• Ciclo de Vida	13
• Estructura de Administración del Proyecto	14
• Herramientas y Técnicas que se emplearán	14
• Metodología Entidad - Relación	15
• Técnica Arriba - Abajo (TOP-DOWN)	16
• Planificación del desarrollo mediante Gráficas de Gantt	17
• Lenguaje de Programación	17
• <b>Capítulo 2 Fase de Diseño del S.A.A.H.</b>	<b>20</b>
• Diseño Estructural	20
• Marco Teórico	20
• Diagrama de Contexto	21
• Diagrama Jerárquico Funcional	21
• Diagrama de Flujo de Datos	22
• Diagrama Entidad - Relación	22
• Marco de Aplicación	23
• Diagrama Jerárquico Funcional General	23
• Consulta Externa	24
• Admisión Hospitalaria	28
• Hospitalización	32
• Relaciones Públicas	36

• Farmacia	40
• Estándares del Sistema	44
• Tablas y Catálogos	44
• Variables	45
• Archivos	45
• Formas	46
• Teclas de Ayuda	47
• Reportes	48
• Documentación	48
• Diseño de Menús	49
• Diseño de Formas y Reportes	52
• Diseño Detallado	56
• Diccionario de Datos	56
• Definición de Tablas del S.A.A.H.	60
• Especificación de Programas	67
• <b>Capítulo 3 Fase de Desarrollo del S.A.A.H.</b>	<b>75</b>
• Programación, Documentación y Pruebas	76
• Pruebas del Sistema	97
• Pruebas de Integración	98
• Pruebas de Aceptación	99
• <b>Capítulo 4 Fase de Implantación del S.A.A.H.</b>	<b>101</b>
• Plan de capacitación a usuarios	102
• Control de Recursos	102
• Conversión de datos	104
• Operación del Sistema por el usuario	104
• Conversión del nuevo Sistema	104
• Liberación final del Sistema	105
• <b>Resultados y Conclusiones</b>	<b>106</b>
• <b>Bibliografía</b>	<b>109</b>
• <b>Apéndices</b>	
• <b>Apéndice A.</b> ¿Porqué INFORMIX?	
• <b>Apéndice B.</b> Diseño de Formas, Reportes y Programación	
• <b>Apéndice C.</b> Glosario de Términos	

# INTRODUCCIÓN

El Instituto de Seguridad y Servicio Social de los Trabajadores del Estado (I.S.S.S.T.E.), como parte integrante del Sector Salud, tiene como uno de sus objetivos principales el proporcionar atención médica a los trabajadores al servicio del Estado y sus familiares, cuyo número aproximado asciende a 9 millones de derechohabientes. Por lo que los centros hospitalarios del I.S.S.S.T.E. se han caracterizado por brindar atención a un gran número de pacientes, con la consiguiente necesidad de contar con sistemas eficientes. En virtud de que el aparato administrativo de los hospitales, por una parte desarrolla las funciones y actividades propias de cualquier órgano administrativo, y por otra, maneja toda la información que se genera por la prestación de servicios médicos, los hospitales son centros de trabajo de delicada operatividad y compleja administración.

Anteriormente el I.S.S.S.T.E. había utilizado herramientas de computación para apoyar sistemas convencionales, como es el caso de la nómina, contabilidad, presupuestos, almacenes y pensiones entre otros; sin embargo, con la expansión del *hardware* a menor costo y la tecnología de los equipos interactivos y concurrentes, otras áreas que también son de vital importancia pueden ser sistematizadas con éxito, como es el caso de la administración hospitalaria.

En nuestro país, hasta la década de los setentas no se había incursionado en esta área, debido a los altos costos que implicaba instalar equipos en configuración espejo, con sistemas desarrollados en otros países para resolver sus necesidades específicas. Por otro lado, el costo de mantenimiento de esos equipos y su *software* era sumamente elevado.

En la década de los ochentas ocurren cambios significativos en el mundo de la informática, los cuales abren prácticamente todos los ámbitos de la actividad humana a la computación.

En la actualidad, hablar de avances tecnológicos y específicamente de informática, es sin duda alguna, hacer referencia a algo que influye muy de cerca en el bienestar y la calidad de vida de la población.

Los bienes y servicios provenientes de la investigación científica nos han permitido contar con mejores y cada vez más sofisticados equipos biomédicos, que permiten una oportuna detección y un amplio conocimiento del estado físico del ser humano; han fomentado la evolución a pasos agigantados de los equipos de cómputo, abriendo paso a computadoras cada vez menos costosas, más amigables y poderosas; de igual manera, han permitido dar un fuerte impulso a las telecomunicaciones y a muchas otras áreas de importancia.

El desarrollo tecnológico ha permitido reducir el tiempo que transcurre entre el descubrimiento científico y su materialización en herramientas aplicables a la vida diaria. Es de esta manera que cada vez que se habla de avances en materia de informática, se encuentran aplicaciones de cómputo más prácticas y útiles para aplicarse en la vida cotidiana. Específicamente hablando de la atención médica y la administración de hospitales, los resultados están a la vista, ofreciendo el mercado diversas oportunidades, ya que existen varias tecnologías a la vez y cada una de ellas conteniendo características diferentes, mismas que se evaluaron para el desarrollo de un sistema de cómputo integral que atendiera las necesidades y características particulares de los hospitales del I.S.S.S.T.E.

En el presente trabajo se muestra el desarrollo del Sistema Automatizado para la Administración Hospitalaria (S.A.A.H.). En él se describen las actividades que se llevaron a cabo para la realización de dicho sistema, desde su fase inicial hasta el término del mismo, explicando las técnicas y herramientas utilizadas para su desarrollo, así como los resultados obtenidos.

Para el desarrollo del sistema se plantea una fase de análisis, en la cual se hace una definición del sistema, tomando en cuenta la problemática así como su justificación para la realización de éste, definiendo las metas, restricciones, alcances, beneficios y riesgos, del sistema y del proyecto. Cabe mencionar que en esta fase se describirá el modelo del ciclo de vida, los requerimientos preliminares de personal y recursos, lenguajes y herramientas para su desarrollo y por último, la planificación del proyecto mediante gráficas de Gantt.

La siguiente fase es la de diseño, la cual se desarrolla en dos etapas: la primera estructural, conteniendo diagramas de procesos, definición de estándares, formas de entrada y la elaboración de reportes y estadísticas; la segunda será detallada, definiendo las tablas, códigos, parámetros y especificaciones de programación principalmente.

Se continúa con la fase de desarrollo, en la cual se lleva a cabo la programación de aplicaciones y la ejecución de pruebas; así como la realización de la documentación técnica y las pruebas finales del nuevo sistema.

Por último, en la fase de implantación se desarrolla el plan de capacitación y aceptación del sistema, además de la liberación del mismo.

Se incluye también la bibliografía correspondiente y tres apéndices: el primero de ellos contiene el por qué de la utilización de INFORMIX, en el segundo se incluye el diseño de algunas pantallas, reportes y programas, y finalmente en el tercero se integra un glosario de términos técnicos y médicos.

CAPITULO 1

# FASE DE ANÁLISIS DEL SAAH

Durante esta primera fase del proyecto se llevará a cabo la planificación, cuyas actividades principales son la definición del problema, desarrollo de la estrategia de la solución recomendada y la planificación del proceso de desarrollo.

Los productos de esta etapa son la definición del sistema y el plan del proyecto. En la definición del sistema se plantea el problema así como las diversas alternativas de solución, eligiendo y justificando la más viable.

En el plan del proyecto se incluye el modelo del ciclo de vida, la estructura de administración del proyecto, la planificación del desarrollo y de recursos, y por último, las herramientas y técnicas que se emplearán.

## Definición del Sistema

### Definición del problema

Como consecuencia del encarecimiento de los servicios que ha otorgado la medicina privada en los últimos años, un mayor número de derechohabientes

demanda día con día los servicios de seguridad social que presta el I.S.S.S.T.E.; situación que ha hecho crecer la población atendida, implicando el manejo de más información, que se refiere tanto al derechohabiente en su calidad de paciente, como la correspondiente a la operación y administración de sus Hospitales.

De continuar con el manejo de información sin un sistema de cómputo integral, que interrelacione las diversas áreas implicadas en la atención del paciente, se podría afectar la oportuna toma de decisiones y, como consecuencia, repercutir en la calidad y cantidad de la prestación de los servicios médicos que se proporcionan a la derechohabencia.

## Justificación del sistema

El Instituto, a través de las Subdirecciones Generales Médica y de Administración, con el fin de hacer frente eficientemente a las crecientes necesidades de atención médica y de manejo de información, tomaron la decisión de utilizar sistemas automatizados, que permitieran apoyar la administración de los centros hospitalarios en forma expedita y racional, aprovechando para ello los recursos existentes, es decir, utilizando adecuadamente la capacidad instalada.

Para ello y a efecto de apoyar en forma automatizada la operación hospitalaria, se revisaron varios de los sistemas hospitalarios existentes tanto en México como en otros países, encontrándose que en todos los casos hubiera sido necesario realizar ajustes sustanciales que representarían considerables erogaciones económicas, las cuales afectarían el patrimonio del Instituto, además de no garantizar el éxito de la implantación.

Por esto, se decidió que en lugar de adaptar la operación de los centros hospitalarios a la filosofía de sistemas para hospitales de características diferentes, resultaba indispensable desarrollar un sistema integral de información que permitiera satisfacer las necesidades específicas del I.S.S.S.T.E., el cual, al concluirse, proporcionaría diversas ventajas técnicas, entre las que destacan el contar con independencia tecnológica y la posibilidad de implantar copias del sistema en varios centros hospitalarios del mismo Instituto, sin necesidad de pagar los derechos que se generan como consecuencia del uso del software desarrollado por empresas privadas. El sistema a desarrollar deberá cumplir con los siguientes objetivos:

## Objetivos

### Del proyecto:

- Simplificar y agilizar la administración y operación de los servicios médicos del hospital.
- Permitir la obtención de indicadores de la atención médica a través de las estadísticas generadas por el sistema.
- Generar información en forma oportuna para que pueda ser analizada por el personal médico y de enfermería.
- Optimizar la administración y la prestación de los servicios médicos de los hospitales de tercer nivel<sup>1</sup> del I.S.S.T.E., mediante un sistema de cómputo eficiente.
- Definición de un sistema con los atributos de calidad requeridos: confiabilidad, portabilidad, eficiencia, solidez y corrección.

## Alcances

El sistema, desde su diseño, estará enfocado a facilitar la operación de la atención médica desde el ingreso del paciente al hospital hasta su egreso, considerando los servicios de Consulta Externa, Admisión Hospitalaria, Hospitalización, Atención al Derechohabiente y Farmacia. A continuación se describirá brevemente cada uno de los servicios mencionados:

### Consulta Externa

Este servicio tendrá por objetivo la planificación y control de citas en el área asistencial de Consulta Externa, validando días de consulta habituales de la semana, horarios y médicos disponibles. La confirmación de la cita se dará al

---

<sup>1</sup> Las unidades médicas de 3er. nivel del ISSSTE son los hospitales regionales, los cuales cuentan con la mayor infraestructura en cuanto a equipo, no. de camas y especialidades, como por ejemplo el Hospital "Adolfo López Mateos".

presentarse el paciente en el consultorio, en caso de no presentarse, se cancelará su cita.

### **Admisión Hospitalaria**

Dentro de este servicio se contará con un control de lista de espera de pacientes pendientes de ingresar, esto es, cuando se reciba una solicitud de ingreso se registrarán los datos personales del solicitante y del responsable del paciente, así como los específicos del ingreso. A partir de estos datos y del censo de la ocupación hospitalaria a diversos niveles, se emitirán estados de previsión de disponibilidad de camas por servicio y fecha.

### **Relaciones Públicas (Atención al Derechohabiente)**

Este servicio proporcionará información referente al paciente como es su estado de salud, cama asignada, médicos responsables, servicio de hospitalización, etc.

### **Hospitalización**

En esta sección se controlará el proceso de Hospitalización de los pacientes que se inicia con el ingreso al hospital y culmina al producirse la alta del paciente. Una vez realizado el ingreso, el sistema ofrecerá las opciones de asignación y cambio del personal médico (médicos y enfermeras asignados), estado de salud y traslados del paciente a unidades de Hospitalización especializada, así como el control del egreso que tuvo el paciente.

### **Farmacia**

En esta sección se controlará la atención de solicitudes de medicamentos generadas en los diversos servicios médicos (Hospitalización, Consulta Externa, Urgencias, Quirófanos, etc.); se generará el programa mensual de abastecimiento y se podrán realizar consultas al inventario, verificando y controlando caducidades y las existencias disponibles para surtir total o parcialmente las cantidades solicitadas.

Los cinco módulos descritos comprenden solamente la etapa I del total del proyecto. La segunda fase contempla el desarrollo de los módulos de: Urgencias y Quirófano, Archivo Clínico, Servicios Auxiliares del Diagnóstico (Laboratorio, Banco de Sangre, etc.), Contabilidad, Finanzas y Medicina Preventiva.

Por otra parte, en el diseño se considerará la incorporación de códigos a los diagnósticos médicos, mediante la disponibilidad de un módulo ágil de consulta para localizar sus claves de acuerdo al catálogo de enfermedades de la Organización Mundial de la Salud (O.M.S.). Esto garantizará la precisión de estadísticas de morbilidad y mortalidad hospitalaria, que son necesarias para la planificación institucional.

Los médicos y el personal con acceso al sistema podrán consultar en pantallas o en reportes la información de aquellos pacientes que se encuentran en el hospital, así como a los que se les ha brindado atención médica desde hace tiempo.

Finalmente, cabe señalar que a partir de la incorporación de este sistema, se empezará a generar con oportunidad información histórica relativa a la atención médica hospitalaria, la cual el Instituto la ha obtenido con retraso por muchos años.

## **Beneficios que se proporcionarán**

### **Del proyecto:**

- Control de la información generada en los módulos considerados en el diseño del sistema.
- Generación oportuna de estadísticas que apoyen la toma de decisiones.
- Contar con un nuevo sistema que no requiera recursos de cómputo adicionales para su desarrollo.
- Simplificación de los procesos administrativos y de atención médica .
- Manejo de información confiable y oportuna que fortalezca la toma de decisiones y apoye la operatividad interna de los hospitales.
- Contar con independencia tecnológica evitando así el pago de los derechos que se generan como consecuencia del uso del *software* desarrollado por empresas privadas.

- Desarrollo de un sistema de información que permita satisfacer las necesidades específicas de los hospitales del I.S.S.S.T.E.
- Posibilidad de la implantación del sistema en otros hospitales regionales del I.S.S.S.T.E.
- Posibilidad de adecuar el sistema a unidades medicas de 1o. y 2o.<sup>1</sup> nivel de atención.

## Riesgos

Con base en la experiencia obtenida en el desarrollo de sistemas, se ha observado que los riesgos que se pueden correr son:

### Del proyecto:

- Coordinación insuficiente de los usuarios para darle continuidad a la comunicación que se establecerá mediante el sistema.
- Inadecuada operación por resistencia al cambio.
- Falta de interés y colaboración por parte de los usuarios para su implementación.
- Prolongación en el tiempo estimado para su desarrollo debido a cambios imprevistos tanto internos como externos.

## Estrategia de solución

Dado que en el I.S.S.S.T.E. no existen antecedentes de desarrollo de sistemas integrales orientados a apoyar la operación hospitalaria, y de conformidad con el Programa Institucional de Desarrollo Informático, se instrumentará la realización de un programa para diseñar y desarrollar un sistema

<sup>1</sup> Las unidades médicas de ter. nivel del I.S.S.S.T.E. son las Clínicas de Adscripción, mientras que las de 2o. nivel son los Hospitales de Zona (como por ejemplo el Hospital 'Darío Fernández').

de cómputo integral que apoye la administración de los hospitales del Instituto. Para ello, se seleccionarán los hospitales con una mayor capacidad instalada y compleja infraestructura para hacer frente a la demanda de servicios médicos, lo cual, en su conjunto, presenta mayor dificultad en el manejo de información y, por lo mismo, se asegura que el sistema a desarrollar sea lo suficientemente consistente para garantizar su correcta operación. Como prueba piloto de la operación del sistema se considerará al Hospital Regional "Adolfo López Mateos", el cual será la principal fuente de información.

Para realizar dicha tarea se formará un círculo de calidad, integrado por el personal del Hospital, que determinará la información que requieren para mejorar la operación del mismo e incrementar la calidad del servicio. Este grupo, en forma colegiada y corresponsablemente, realizará el diagnóstico situacional, detectando necesidades específicas que requieran del apoyo prioritario de un sistema de información automatizado. Con base a este diagnóstico, se procederá a realizar el análisis de datos y de funciones que ofrecerá como resultado el diseño conceptual de un sistema que contemple una interrelación con todas las áreas del hospital involucradas en la prestación de los servicios médicos.

Es así como nace el Sistema Automatizado para la Administración Hospitalaria (S.A.A.H.) del I.S.S.S.T.E.

Los recursos computacionales de este sistema serán distribuidos en las áreas más importantes del hospital, considerando la ruta del paciente desde su ingreso hasta su egreso; de esta forma, el sistema irá registrando información en la admisión del paciente, consulta externa y los servicios que integran la hospitalización en piso.

La ruta del paciente se muestra en la página 12 (figura 1.1.). En esta figura se puede visualizar las posibles rutas que puede seguir un paciente a su ingreso, algunas de las cuales se mencionan a continuación:

- 1.- El paciente acude a Consulta Externa en cualquiera de sus especialidades; se le otorga la consulta y egresa a su domicilio, solicitando en caso de existir, la expedición de su licencia médica y/o el surtimiento de su receta.

2.- El paciente acude a Consulta Externa en cualquiera de sus especialidades y es programado a Hospitalización (ingreso programado), confirmando su ingreso a través de Admisión Hospitalaria, en donde se le asignará la cama y el servicio correspondiente. Una vez hospitalizado, el paciente puede ser canalizado a otros servicios (Terapia Intensiva, Quirófano, etc.). Dependiendo de la evolución del estado de salud se tramitará su egreso.

3.- El paciente acude a Urgencias en cualquiera de sus especialidades: pediatría, adultos o toco-quirúrgicas, donde dependiendo de su diagnóstico puede estar sólo en observación y egresar a su domicilio, o bien, ser canalizado a las áreas de: Hospitalización, Quirófano, o en caso de deceso, a Patología.

## RUTA DEL PACIENTE HOSPITAL REGIONAL "ADOLFO LOPEZ MATEOS"

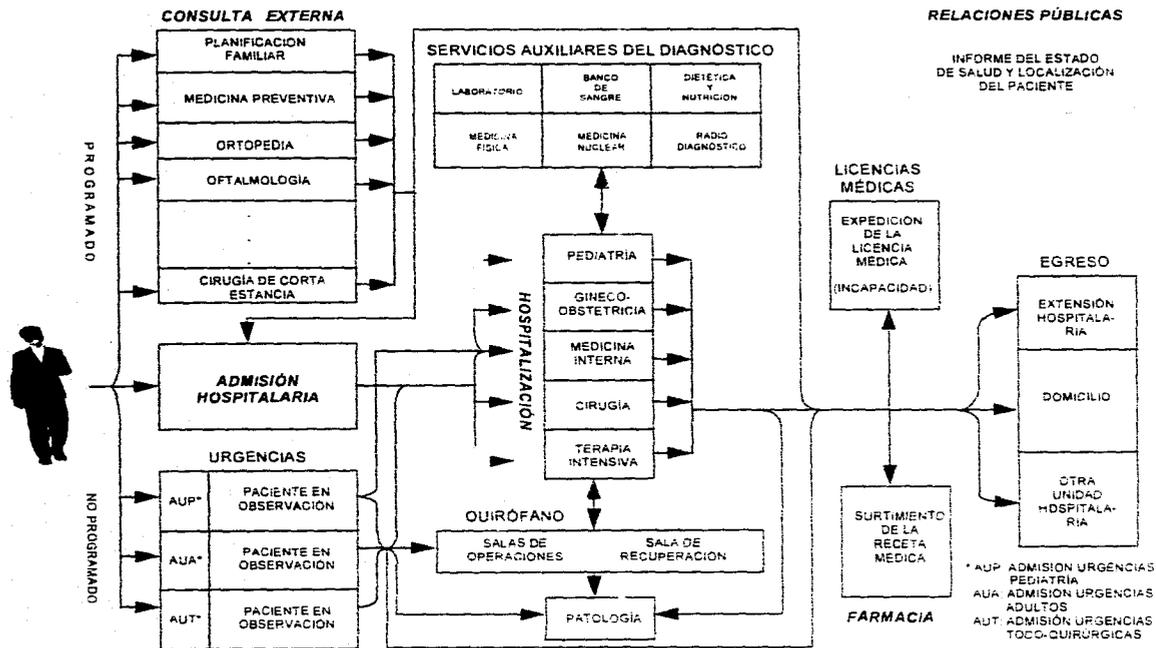


Fig. 1.1 Ruta del Paciente.

## Plan del Proyecto

El plan del proyecto contiene el modelo del ciclo de vida en el que se basará el desarrollo del sistema, la estructura de administración del proyecto, herramientas y técnicas que se emplearán y lenguajes de programación.

### Ciclo de vida

Para el desarrollo del proyecto se considerará el modelo del ciclo de vida. Por ciclo de vida de un sistema de información se entiende el conjunto de fases por las que pasa a lo largo del tiempo, desde la fase de estudio y concepción hasta la de realización, explotación y mantenimiento.

El modelo de fases del ciclo de vida del Sistema Automatizado para la Administración Hospitalaria se muestra en la figura 1.2.

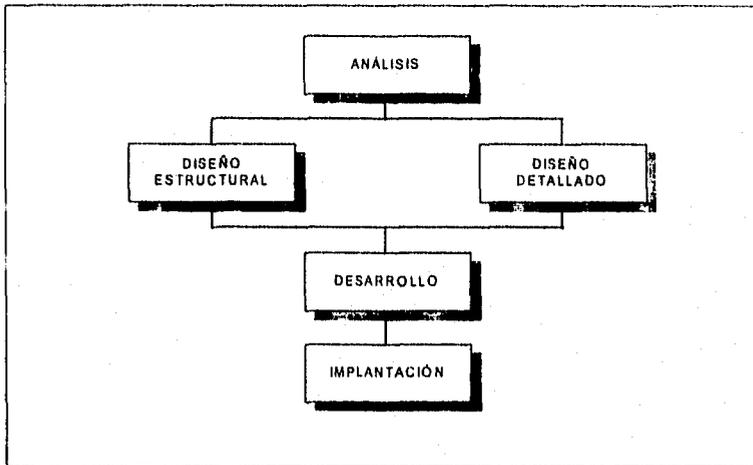


Fig 1.2. Modelo del ciclo de vida del S.A.A.H.

Durante la fase de análisis se identificarán los requerimientos y limitantes del sistema; se identificarán y programarán las actividades requeridas para llevar a cabo el desarrollo del proyecto.

En la fase de diseño se desarrollará la estructura modular del S.A.A.H. para definir las interfases entre los cinco módulos, submódulos y programas, así como la estructura y flujo de datos. Se incluirá además un diseño detallado en donde se definirá entre otras cosas, la estructura de la base de datos con las especificaciones de cada uno de los programas a desarrollar.

En la fase de desarrollo se llevará a cabo la programación y prueba de los programas que previamente se identificaron y especificaron. Se efectuarán tanto las pruebas individuales por programa como las pruebas de integración de los cinco módulos que formarán el nuevo sistema.

Por último, en la fase de implantación se llevará a cabo el plan de capacitación a los usuarios del nuevo sistema, para lograr su liberación dentro del hospital donde será la prueba piloto.

## **Estructura de administración del proyecto**

La estructura de administración del proyecto se muestra en el figura 1.3. (ver siguiente página). En ésta se representan los módulos a automatizar en la etapa I del proyecto, de igual manera se identifican los módulos a automatizar en la etapa II.

## **Herramientas y técnicas que se emplearán**

Para el desarrollo del proyecto nos auxiliaremos de diversas técnicas dentro de las cuales cabe destacar el modelo entidad-relación, el cual nos permitirá obtener un modelo conceptual de los datos mediante esquemas de entidades y relaciones del sistema, y la técnica de diseño Arriba-Hacia-Abajo (*Top-Down*).

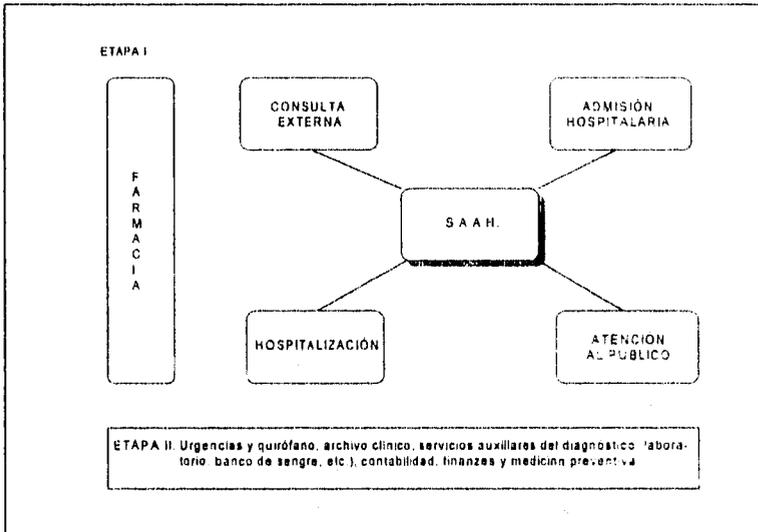


Fig. 1.3. Estructura de administración del proyecto.

### Metodología Entidad - Relación

La metodología entidad - relación se ha usado exitosamente en todo el mundo, como una herramienta de comunicación entre los analistas, diseñadores de sistemas y los usuarios finales, durante las fases de análisis de requerimientos y de diseño conceptual, debido a que es simple y fácil de entender.

Los enfoques convencionales para el diseño lógico de bases de datos normalmente tienen sólo una fase: convertir la información de los objetos del mundo real directamente al esquema del usuario. Esta metodología para el diseño lógico de bases de datos consiste de dos fases principales:

- 1.- Definir el esquema conceptual usando diagramas entidad - relación y

## 2.- Traducir el esquema conceptual al esquema del usuario.

Las ventajas de esta metodología son:

- La división de las funciones y el trabajo hace que el diseño de la base de datos sea más simple y más organizado.
- El esquema conceptual es fácil de diseñar ya que no está restringido por las capacidades del DBMS (*Data Base Management System*) y es independiente del esquema físico y de consideraciones de eficiencia .
- El esquema conceptual es más estable que el esquema del usuario. Si se desea cambiar de un DBMS a otro, el esquema conceptual será el mismo, ya que éste es independiente del DBMS que se use.
- El esquema conceptual expresado por un diagrama entidad - relación es más fácil de entender por gente sin conocimientos en computación .

### **Técnica Arriba-Hacia-Abajo (*Top-Down*)**

Esta técnica consiste en un conjunto de diagramas que gráficamente describen las funciones del sistema de un nivel general a un nivel detallado.

Al principio cada función principal se identifica y luego se subdivide en funciones menores. La suma de esas funciones menores es igual a la función principal.

Los diagramas *Top-Down* se pueden usar desde el inicio de un proyecto hasta su implantación y son útiles para el mantenimiento de programas.

El principal objetivo de la técnica *Top-Down* es:

- Proveer una estructura por la cual se puedan entender las funciones del sistema. Los diagramas están organizados en una estructura jerárquica como en un organigrama, donde cada diagrama en cualquier nivel es un subconjunto del nivel de arriba.

## Planificación del desarrollo mediante Gráficas de Gantt

Para llevar a cabo un buen proyecto de desarrollo de *software*, debemos comprender el ámbito de trabajo a realizar, las tareas a ejecutar y la agenda a seguir. Esta última define los tiempos en los cuales recaerá la duración del proyecto. El apoyo que proporcionan las técnicas gráficas para la planificación de cualquier desarrollo es indudablemente de gran utilidad. Actualmente existen diversas técnicas gráficas que permiten facilitar éstas tareas, como por ejemplo, las Gráficas de Gantt, Redes Pert, etc.

Con el fin de planificar y controlar adecuadamente las actividades que se llevarán a cabo durante el desarrollo de proyecto, nos apoyaremos en la técnica de Gráficas de Gantt.

La planificación del desarrollo del S.A.A.H. se muestra en la figura 1.4. (ver página siguiente). En ésta se describen las actividades que se deberán ejecutar, los responsables de las mismas y los tiempos estimados y reales para la realización de cada una de éstas.

## Lenguaje de Programación

Por restricciones del proyecto, el desarrollo del sistema se llevará a cabo con los recursos de cómputo propios del Instituto. Por lo que en este caso estará sustentado mediante el manejador de bases de datos relacional INFORMIX, bajo el ambiente UNIX, el cual cuenta con un lenguaje propio de programación de cuarta generación, lo cual permitirá su permanencia futura. Cabe destacar que gracias a la modularidad que ofrece INFORMIX la mayoría de los módulos podrán ser reutilizados en unidades médicas de menor nivel del I.S.S.S.T.E.

SISTEMA AUTOMATIZADO PARA LA ADMINISTRACION HOSPITALARIA (S.A.A.H.)

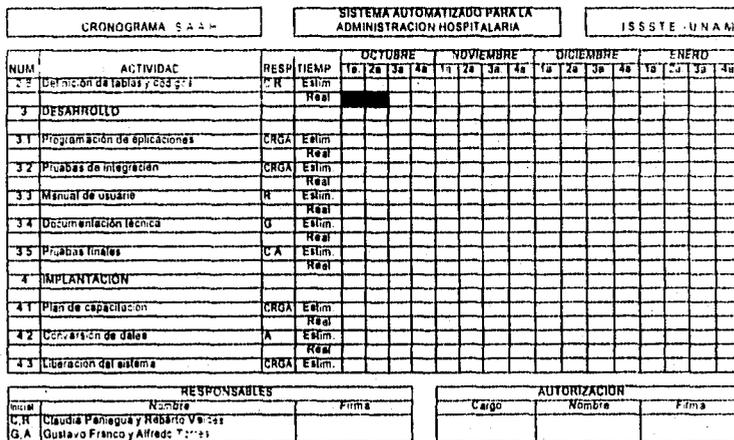
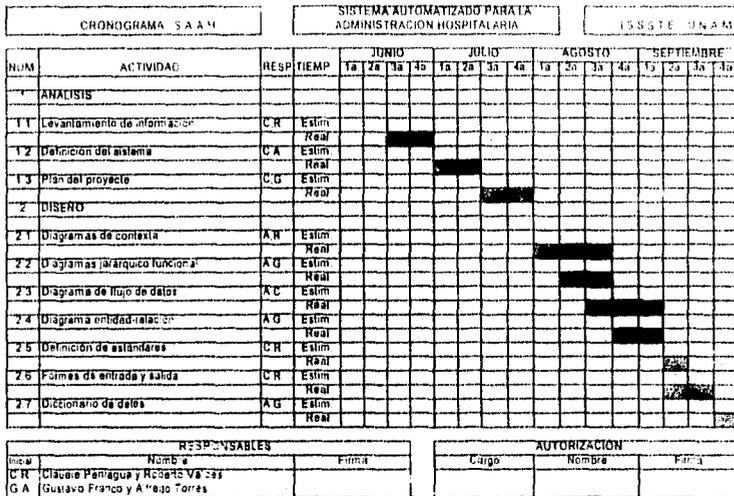


Fig. 1.4. Planificación del desarrollo del S.A.A.H.

**INFORMIX-4GL** es un lenguaje de cuarta generación, parecido a Pascal, Fortran, Cobol y otros lenguajes de alto nivel, o lenguajes de "Tercera Generación".

Usando **INFORMIX-4GL** se puede fácilmente:

- Crear menús de anillo (similares a los de LOTUS 1-2-3).
- Captar información por medio de pantallas de captura.
- Uso de **RDSQL** (*Rapid Development Structured Query Language*) para el manejo de la base de datos.
- Ayudas en varios niveles.
- Pantallas y manejo de ventanas para consulta rápida.
- Captura de información multi-columnas en una forma.

**INFORMIX-4GL** usa declaraciones no-procedimentales, las cuales permiten describir lo que se requiere, sin especificar pasos detallados. Por lo que existe un desarrollo acelerado de Bases de Datos con respecto a los lenguajes de tercera generación (COBOL, PASCAL, FORTRAN, etc.). Al utilizar menos líneas de código, reduce en menores errores y en una eficiente depuración y mantenimiento.

Provee de declaraciones de procedimientos (característico de los lenguajes de tercera generación) para usarlas cuando se requiere, lo que permite mayor flexibilidad en el diseño de la base de datos.

Los Sistemas Operativos en los que puede ejecutarse son : UNIX, XENIX, MS-DOS, OS/2 y VMS.

Se puede ejecutar desde una computadora personal hasta una *mainframe*.

Para mayor información acerca de las características de este manejador consúltese el apéndice A.

## CAPITULO 2

# FASE DE DISEÑO DEL SAAH

Esta fase del proyecto la dividiremos en dos subfases: el diseño estructural y el detallado. En el primero se identifican los componentes de la programación, su descomposición en módulos de procesamiento y estructuras de datos conceptuales, y la especificación de interconexión entre componentes. El segundo se refiere a detalles de como definir tablas, códigos y parámetros, especificaciones de programación, etc.

## **Diseño Estructural**

### **Marco Teórico**

En el diseño estructural, el sistema se divide en subsistemas, módulos y submódulos, mediante la identificación de todas sus funciones bajo una estructura jerarquizada. Esta es una filosofía básica aplicable a cualquier clase de problemas, dividir el problema total en subdivisiones que pueden ser manejados

en forma separada. Las ventajas del diseño estructural y modular son las siguientes:

- El trabajo tiende a ser clasificado de lo más complejo a lo menos complejo, en forma jerarquizada y controlada.
- La documentación es fácil de crear y mantener.
- El mantenimiento de los sistemas se facilita. Este es un factor importante para reducir los tiempos de mantenimiento y sus implicaciones en el resto de las fases del ciclo de vida.
- Las pruebas del sistema son fáciles y normalmente más comprensivas.
- Se facilita la integración con otros sistemas.
- Se identifican claramente los módulos de propósito general desarrollados y utilizados por dos o más sistemas diferentes.

Con la finalidad de conocer las funciones de las diferentes áreas contempladas en el diseño del sistema, así como el flujo y secuencia de procesos, nos apoyaremos en los siguientes diagramas:

#### **a) Diagrama de Contexto**

El objetivo de este diagrama es mostrar la relación existente entre las diferentes áreas que interactúan con el área a automatizar. En la parte central se muestra la entidad principal (módulo a automatizar), y a su alrededor se muestra la relación que existe con las demás entidades del hospital.

#### **b) Diagrama Jerárquico Funcional**

Este diagrama proporciona una representación del sistema y la relación jerárquica de sus elementos (subsistemas, submódulos, procesos, etc.). Su construcción se realiza con base en el principio de diseño de Arriba-Hacia-Abajo (*Top-Down*). Cada bloque en el diagrama jerárquico tiene un número para los diagramas de flujo de datos.

**c) Diagrama de Flujo de Datos**

El diagrama de flujo de datos es una técnica que tiene como finalidad interpretar gráficamente el flujo de los datos identificando las entradas, archivos y salidas de un sistema, pasando por todos sus procesos de transformación.

La representación simbólica utilizada en los diagramas de flujo de datos del sistema es la que se muestra en la figura 2.1.

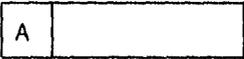
Símbolo	Representación
	Flujo de Datos
	Proceso de Transformación
	Entidades
	Archivo de Datos
	Catálogo de Datos

Fig. 2.1. Representación simbólica para los Diagramas de Flujo de Datos.

**d) Diagrama Entidad-Relación**

Antes de diseñar formalmente los archivos del nuevo sistema, es importante conocer la relación que tienen sus entidades entre sí, mediante la

definición de la Entidad-Relación de los archivos, para lo cual se requiere hacer lo siguiente:

- 1.- Relacionar cada uno de los archivos que previamente se identificaron en el diagrama de flujo de datos (entidades).
- 2.- Definir los elementos de datos principales que describen a los registros de cada archivo (campos).
- 3.- Identificar al elemento de datos que por su valor diferencia a un registro de otro (clave del registro).
- 4.- Graficar la relación de los campos de todas las entidades o archivos del nuevo sistema.

## Marco de Aplicación

A continuación se desarrollarán por módulo cada uno de los diagramas ya mencionados correspondientes al S.A.A.H.

### Diagrama Jerárquico Funcional General

La figura 2.2. muestra el diagrama jerárquico general del S.A.A.H. En éste se observan los cinco módulos contemplados en el diseño del sistema: Consulta Externa, Admisión Hospitalaria, Hospitalización, Relaciones Públicas y Farmacia.

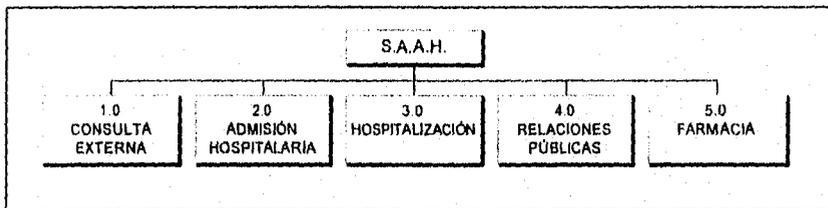


Fig. 2.2. Diagrama jerárquico funcional general.

## Consulta Externa

### Diagrama de contexto

En la figura 2.3. se muestra el diagrama de contexto de Consulta Externa y los módulos con los que se interrelaciona.

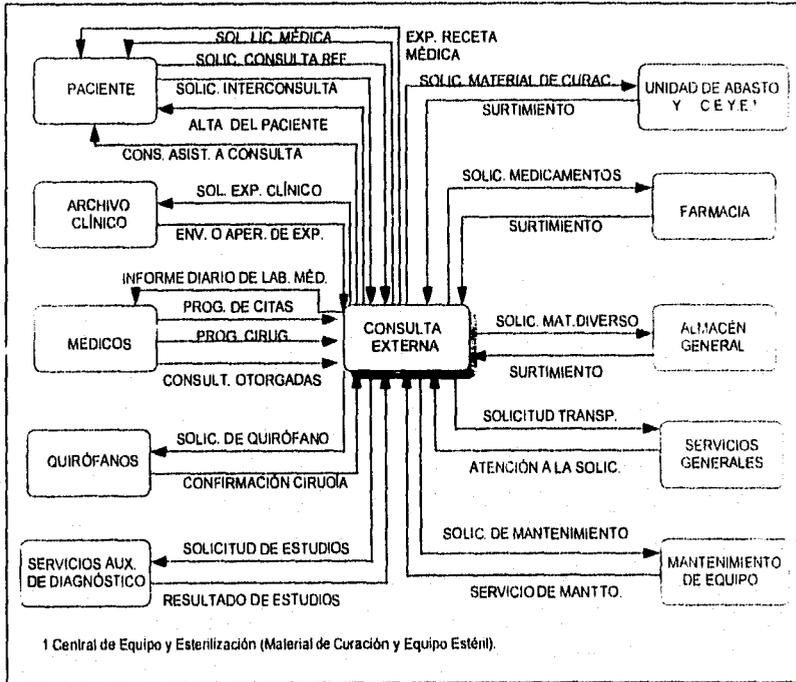


Fig. 2.3. Diagrama de contexto de Consulta Externa.

**Diagrama Jerárquico Funcional**

A continuación, en la figura 2.4. se muestra el diagrama jerárquico funcional de Consulta Externa con sus principales submódulos: Control de Citas, Requerimientos del Paciente, Requerimientos del Servicio, Roll Médico e Ingresos Programados.

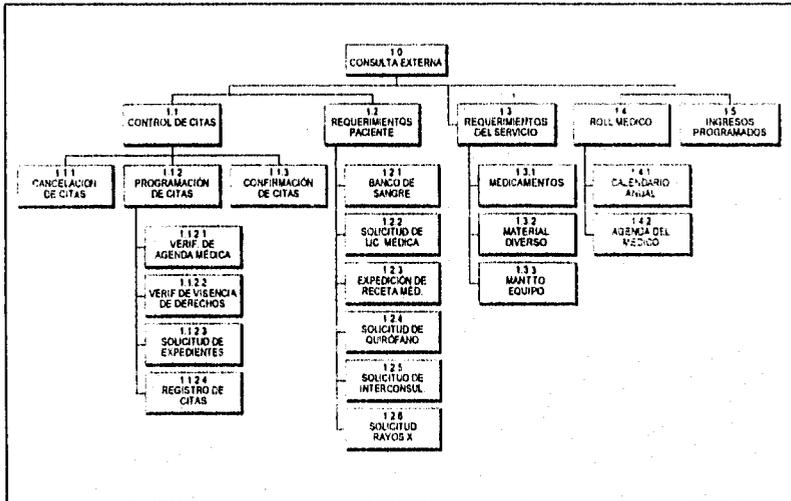


Fig. 2.4. Diagrama jerárquico funcional de Consulta Externa.

**Diagrama de Flujo de Datos**

En la figura 2.5. se muestra el diagrama de flujo de datos del módulo de Consulta Externa.

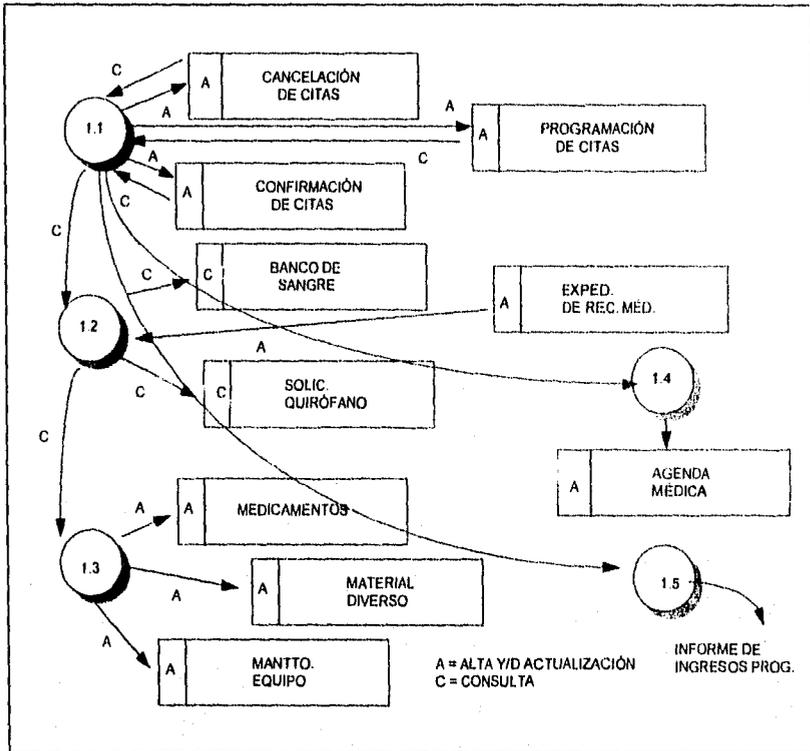


Fig. 2.5. Diagrama de flujo de datos de Consulta Externa.

**Diagrama Entidad - Relación**

En la figura 2.6. se muestra el diagrama Entidad - Relación de Consulta Externa. Cabe mencionar que sólo se especifican los campos de relación entre los archivos; estos campos (Especialidad, Médico, Expediente y Clave de Medicamento) representan las llaves primarias de enlace de las Entidades diagramadas.

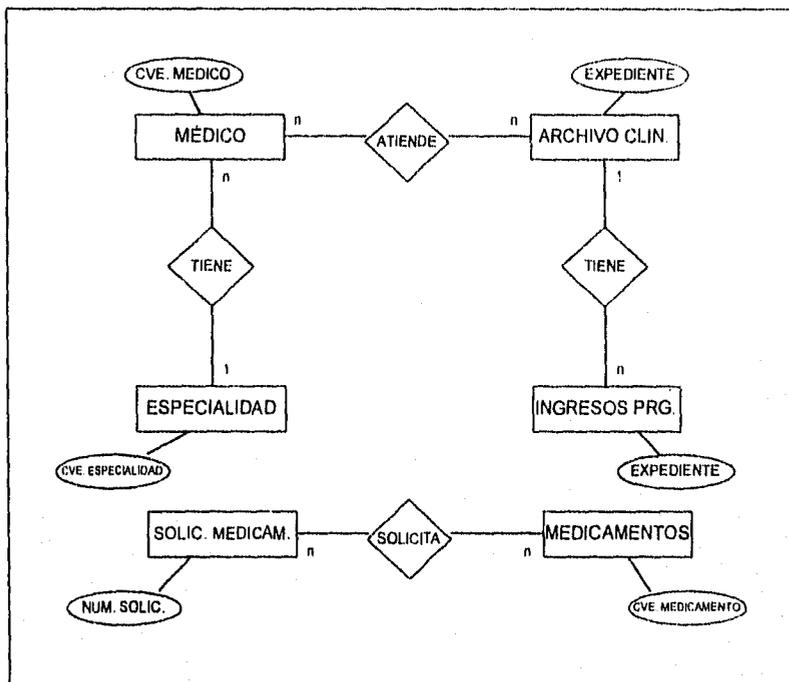


Fig. 2.6. Diagrama Entidad - Relación de Consulta Externa.

## Admisión Hospitalaria

### Diagrama de Contexto de Admisión Hospitalaria

A continuación se muestra en la figura 2.7., el diagrama de contexto de Admisión Hospitalaria y los módulos con los que se interrelaciona.

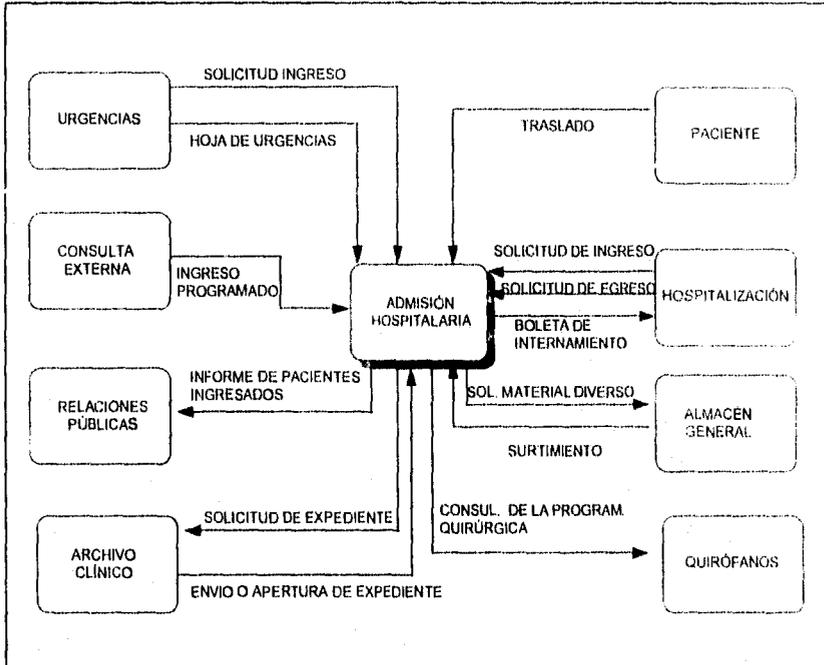


Fig. 2.7. Diagrama de contexto de Admisión Hospitalaria.

## Diagrama Jerárquico Funcional

En la figura 2.8., se muestra el diagrama jerárquico funcional de Admisión Hospitalaria el cual cuenta con los submódulos de Control de Ingresos, Control de Camas y Control de Egresos.

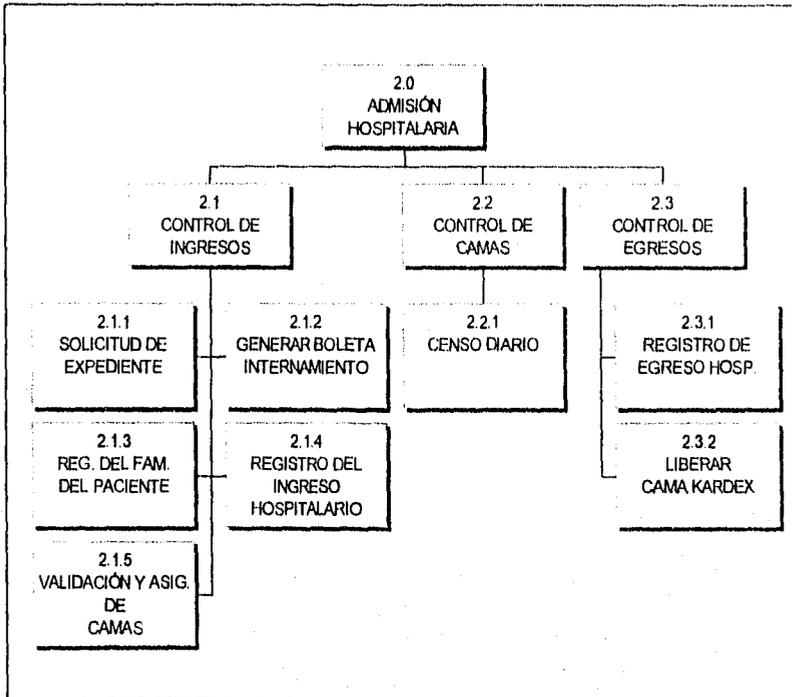


Fig. 2.8. Diagrama jerárquico funcional de Admisión Hospitalaria.

**Diagrama de Flujo de Datos**

En la figura 2.9. se muestra el diagrama de flujo de datos del módulo de Admisión Hospitalaria.

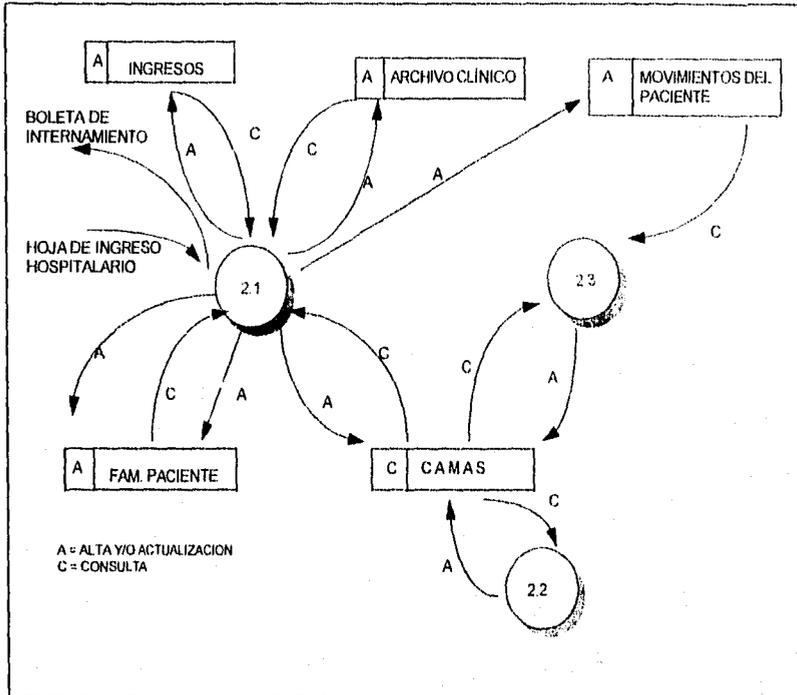


Fig. 2.9. Diagrama de flujo de datos de Admisión Hospitalaria.

**Diagrama Entidad - Relación**

En el diagrama Entidad - Relación de Admisión Hospitalaria representado en la figura 2.10., muestra los campos de acceso entre los archivos.

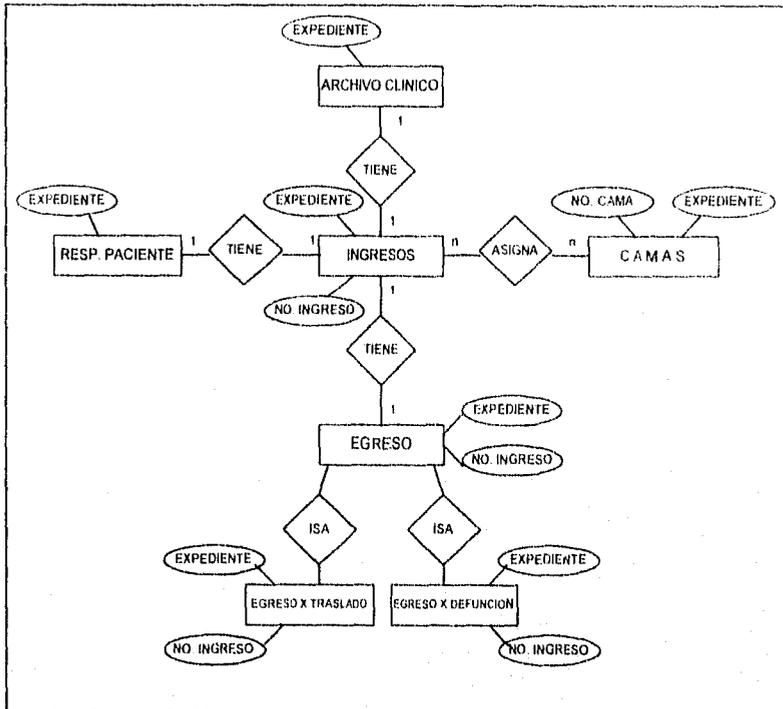


Fig. 2.10. Diagrama Entidad - Relación de Admisión Hospitalaria.

## Hospitalización

### Diagrama de Contexto de Hospitalización

A continuación, en la figura 2.11. se muestra el diagrama de contexto de Hospitalización.

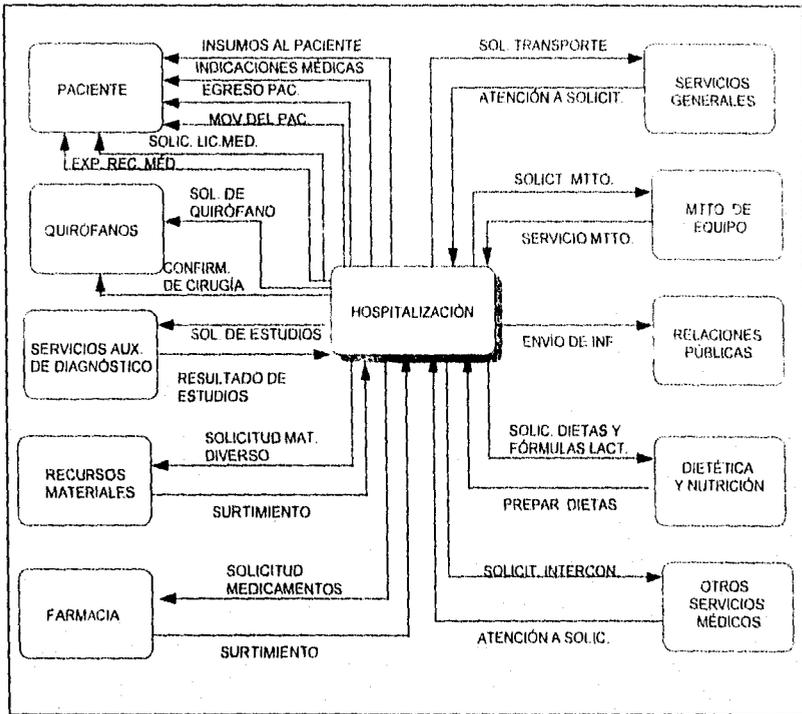


Fig. 2.11. Diagrama de contexto de Hospitalización.

## Diagrama Jerárquico Funcional

A continuación se muestra el diagrama jerárquico funcional (figura 2.12.) de Hospitalización.

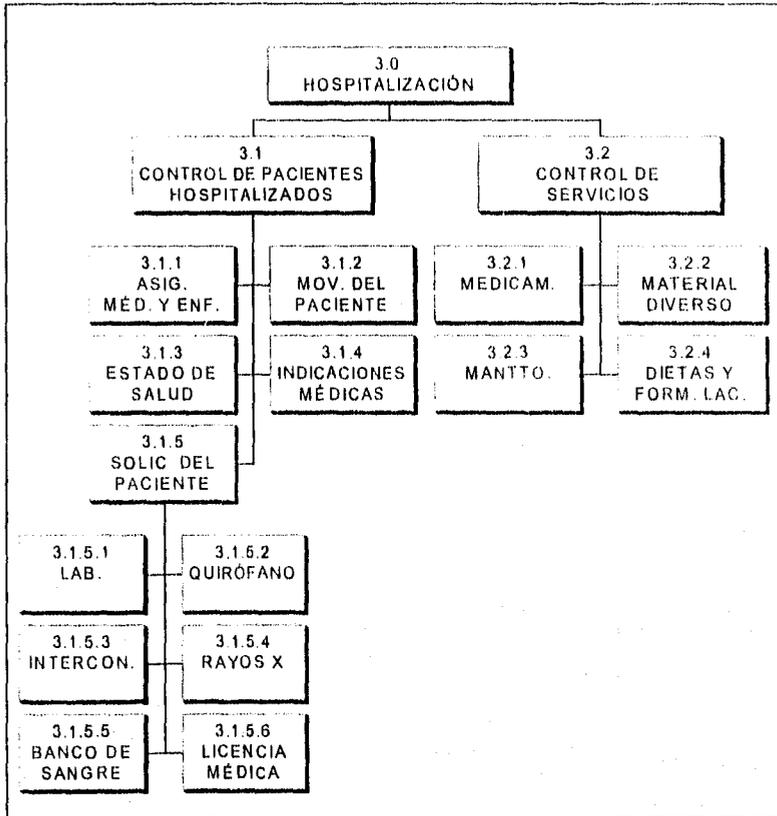


Fig. 2.12. Diagrama jerárquico funcional de Hospitalización.

### Diagrama de Flujo de Datos

En la figura 2.13. se muestra el diagrama de flujo de datos del módulo de Hospitalización.

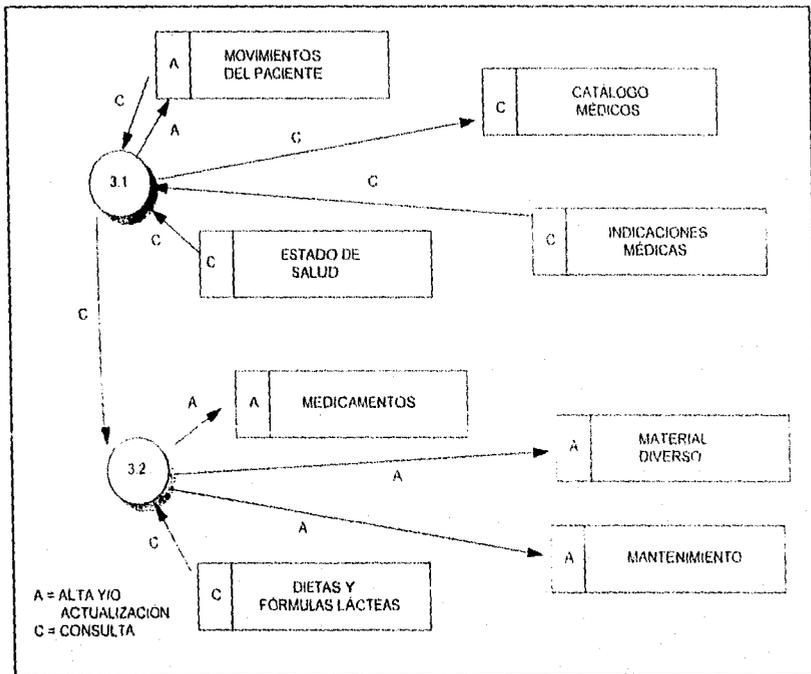


Fig. 2.13. Diagrama de flujo de datos de Hospitalización.

### Diagrama Entidad - Relación

Las relaciones entre los archivos de Hospitalización se muestran en el Diagrama Entidad - Relación de la figura 2.14.

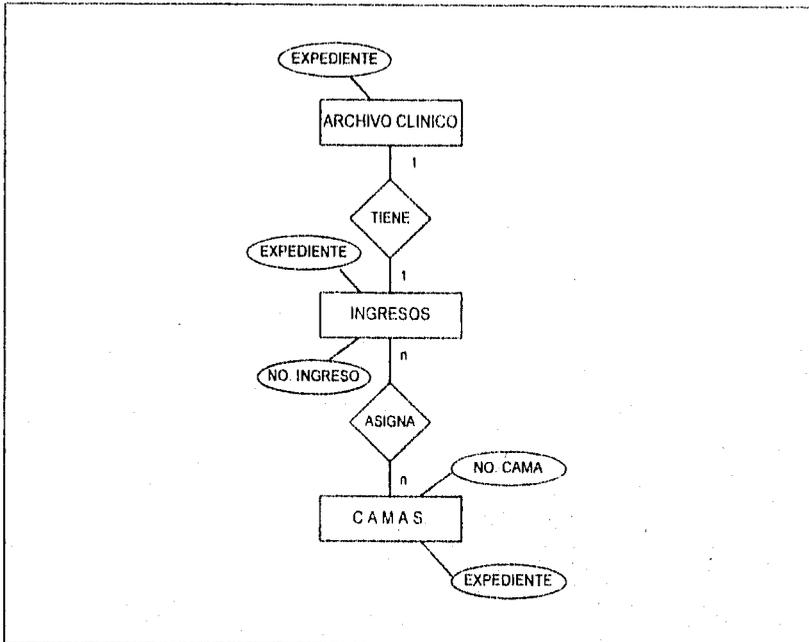


Fig. 2.14. Diagrama Entidad - Relación de Hospitalización.

## Relaciones Públicas

### Diagrama de Contexto

A continuación se muestra en la figura 2.15., el diagrama de contexto de Relaciones Públicas.

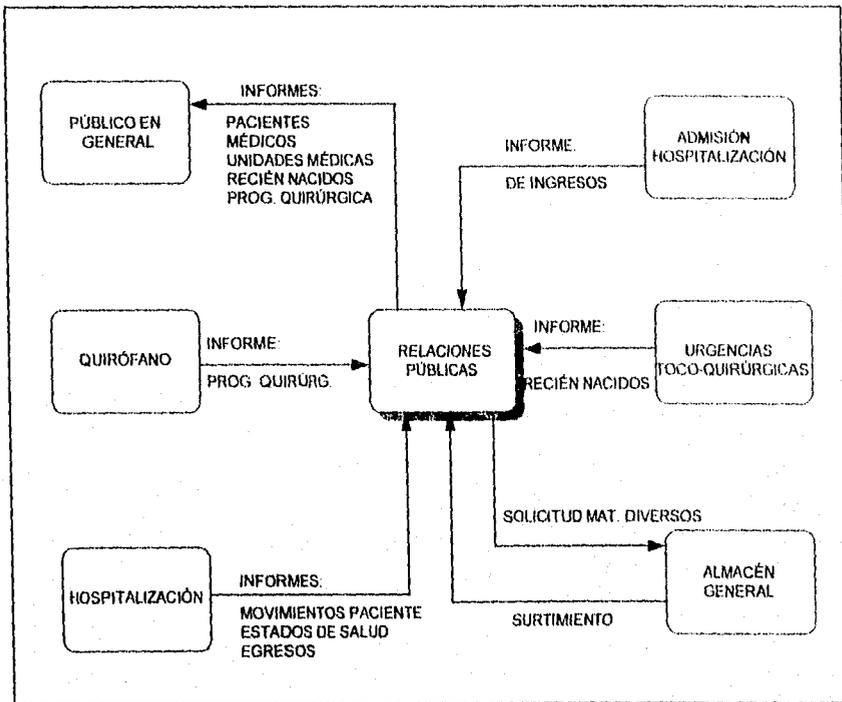


Fig. 2.15. Diagrama de contexto de Relaciones Públicas.

### Diagrama Jerárquico Funcional

A continuación se muestra el diagrama jerárquico funcional de Relaciones Públicas (figura 2.16.).

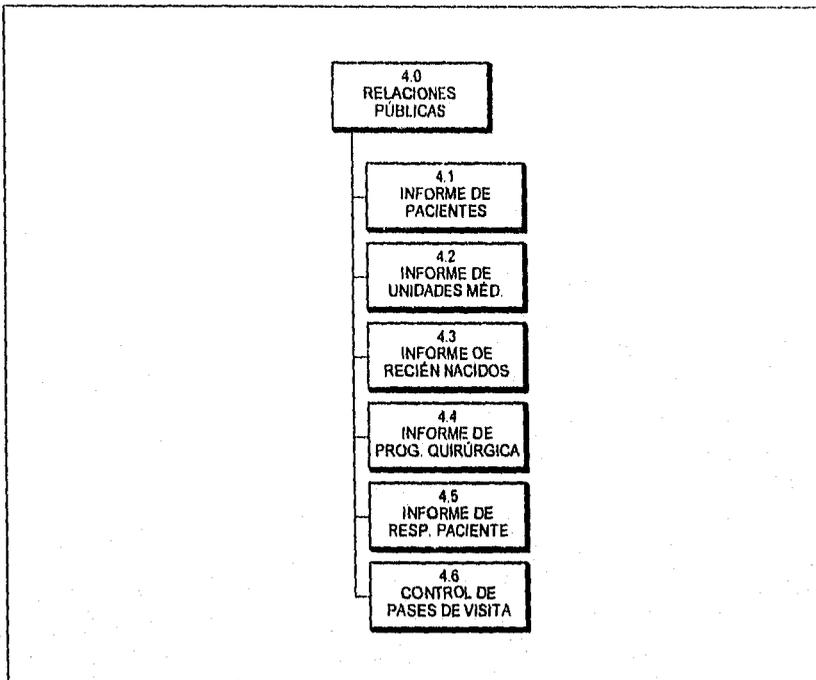


Fig. 2.16. Diagrama jerárquico funcional de Relaciones Públicas.

**Diagrama de Flujo de Datos**

En la figura 2.17. se muestra el diagrama de flujo de datos del módulo de Relaciones Públicas.

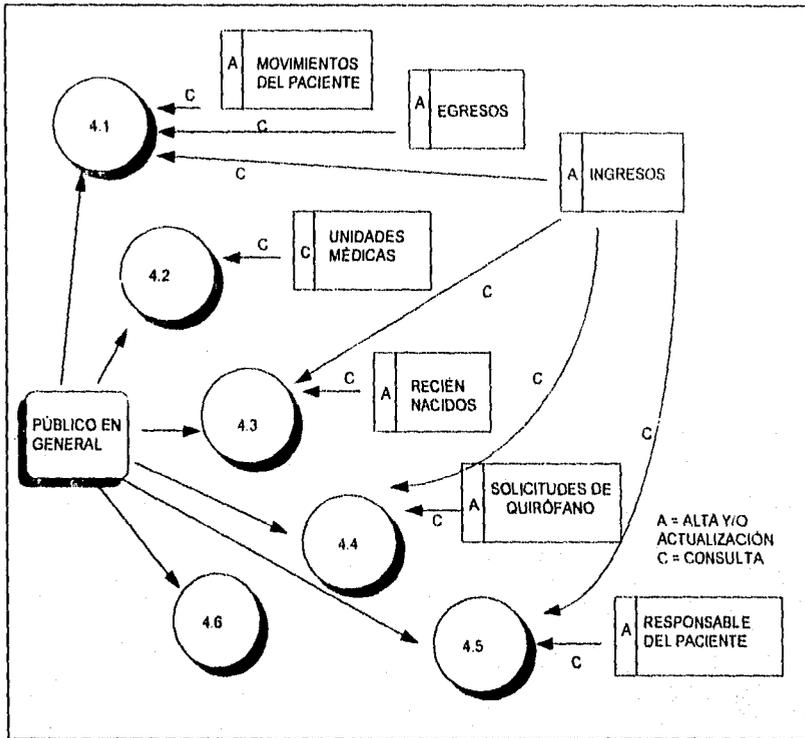


Fig. 2.17 Diagrama de flujo de datos de Relaciones Públicas.

**Diagrama Entidad - Relación**

La figura 2.18. muestra el diagrama Entidad - Relación de Relaciones Públicas.

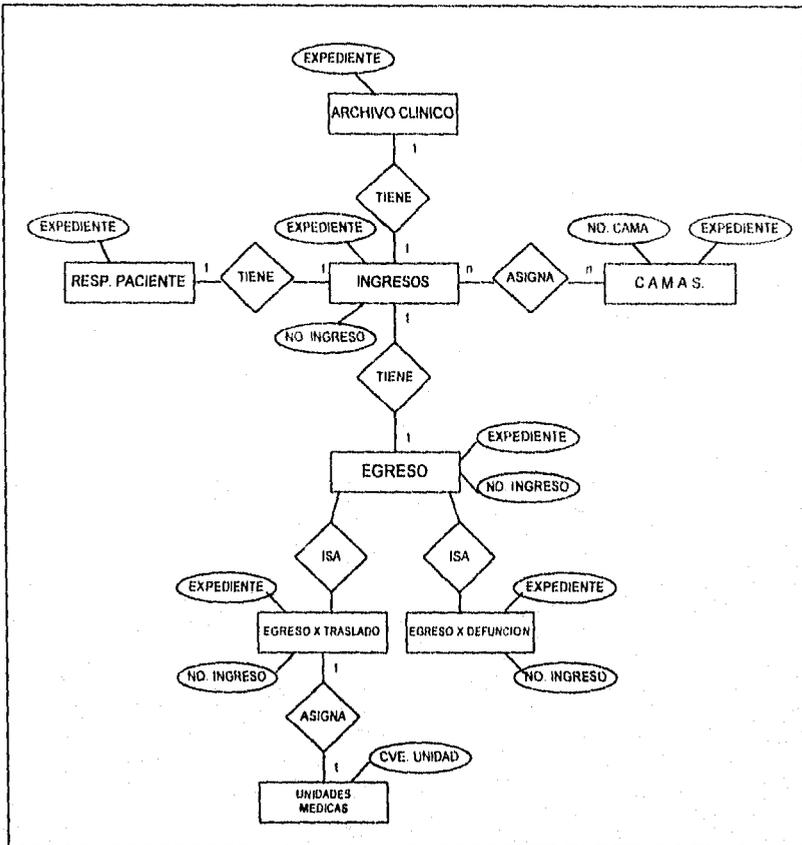


Fig. 2.18. Diagrama Entidad - Relación de Relaciones Públicas.

## Farmacia

### Diagrama de Contexto

A continuación se muestra el diagrama de contexto de Farmacia (figura 2.19.).

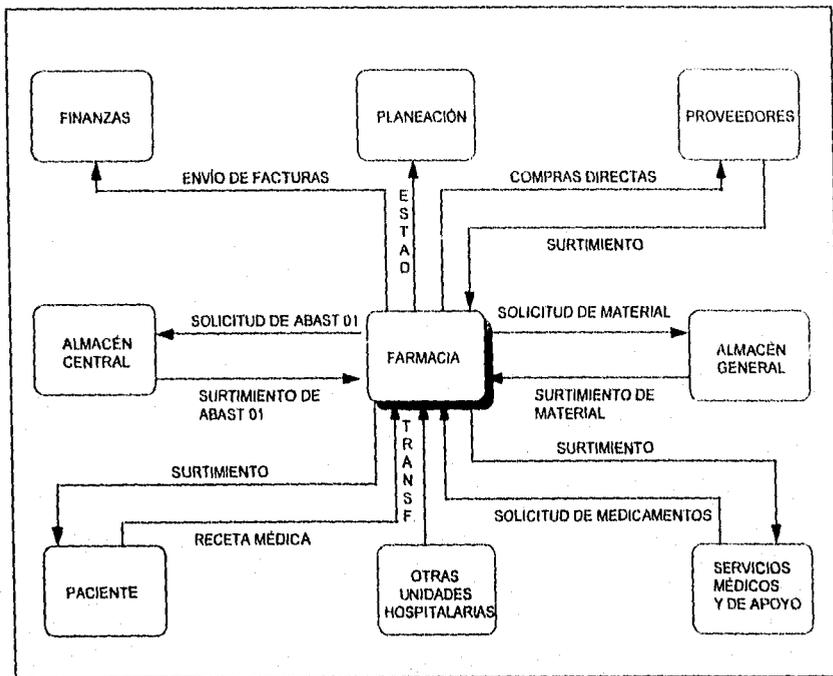


Fig. 2.19. Diagrama de contexto de Farmacia.

### Diagrama Jerárquico Funcional

A continuación se muestra el diagrama jerárquico funcional de Farmacia en la figura 2.20.

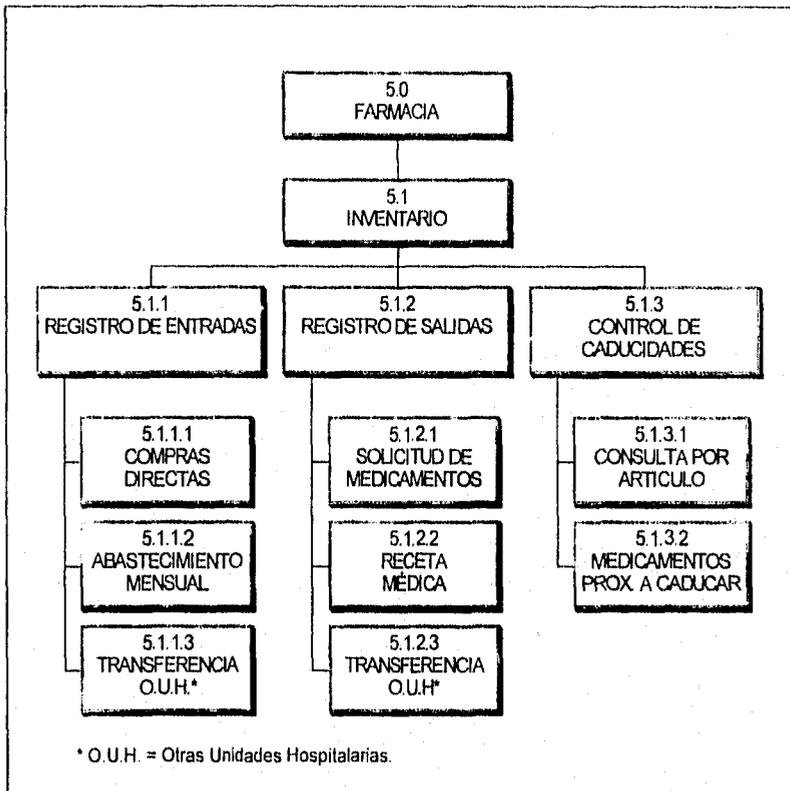


Fig. 2.20. Diagrama jerárquico funcional de Farmacia.

**Diagrama de Flujo de Datos**

En la figura 2.21. se muestra el diagrama de flujo de datos del módulo de Farmacia.

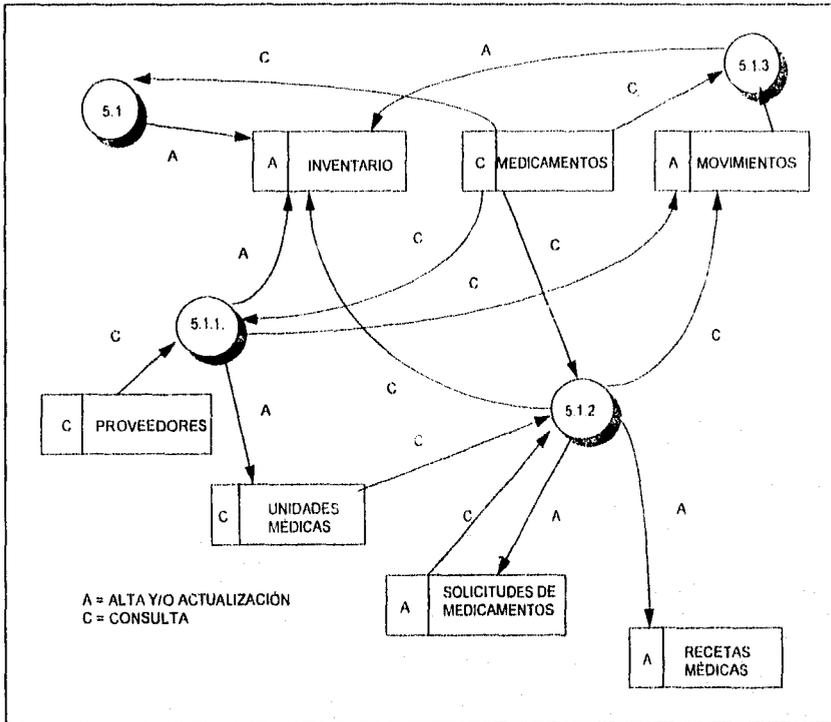


Fig. 2.21. Diagrama de flujo de datos de Farmacia.

**Diagrama Entidad - Relación**

El diagrama Entidad - Relación de Farmacia se muestra en la figura 2.22.

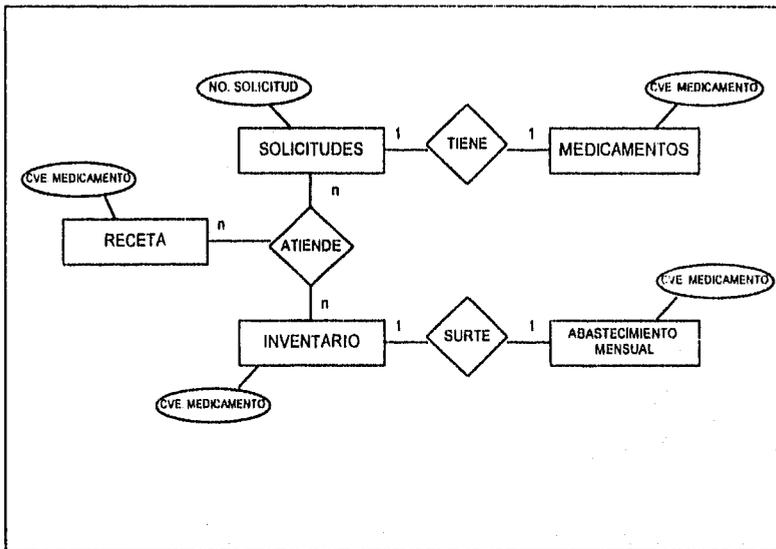


Fig. 2.22. Diagrama Entidad - Relación de Farmacia.





La siguiente lista muestra los diferentes tipos de iniciales que identifican a las secciones de los nombres de los módulos. Cabe señalar que solamente para las rutinas comunes se omitirá la posición del identificador del módulo y en los listados se omitirá la acción.

Tipo:	Módulo
F = Forma	A = Admisión Hospitalaria
L = Listado	C = Consulta Externa
P = Programa (4gl)	R = Relaciones Públicas
R = Rutina Común	H = Hospitalización
M = Módulo (4gl)	F = Farmacia

Acción	Descripción	Ejemplos
A =	Alta	fca_solmed = forma, consulta externa, alta, solicitud de medicamentos.
B =	Baja	
M =	Modificación	mfa_surmed = módulo, farmacia, alta, medicamentos surtidos.
C =	Consulta	

#### 4.- Estándares de Formas y Reportes

Para la estandarización de las formas y reportes se definen los siguientes lineamientos:

##### Formas

Para la definición de las formas, se establecen 8 áreas principales (ver figura 2.23.) las cuales se definen de la siguiente manera:

- 0.- Marco de la pantalla
- 1.- Siglas del sistema y del hospital
- 2.- Nombre del módulo
- 3.- Siglas "HREALM" (Hospital Regional "Adolfo López Mateos")

- 4.- Área de menú y submenús
- 5.- Área de trabajo (despliegue de información)
- 6.- Teclas de ayuda (antepenúltima línea de cada ventana)
- 7.- Mensajes (última línea de cada ventana)
- 8.- Ventana de ayuda (las mismas líneas de teclas de ayuda y mensajes)

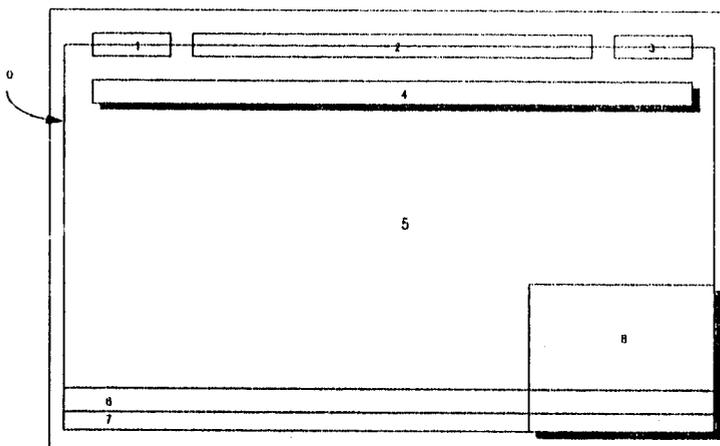


Fig. 2.23. Formato estándar para definición de Formas.

### Definición de Teclas de Ayuda

Para auxiliar al usuario en el uso de formas, se cuenta con las siguientes teclas de ayuda:

TECLA(S)	ACCIÓN
CTRL + N	Avance de página
CTRL + P	Retroceso de página
ESC	Aceptar y continuar
CTRL + B	Cancelar ó salir
CTRL + F	Consultar ayuda
RETURN	Siguiente campo

**Reportes:**

El formato para reportes se presenta en la figura 2.24. en donde se definen cuatro áreas principales, las cuales son:

- 1.- Siglas del sistema y del hospital
- 2.- Encabezado del reporte
- 3.- Fecha, hora y página
- 4.- Área de datos

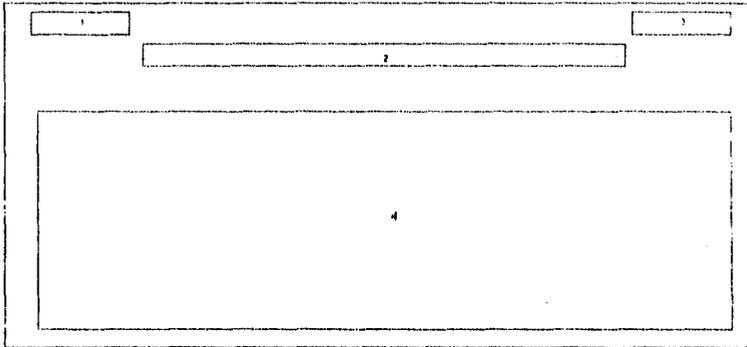


Fig. 2.24. Formato estándar para reportes del S.A.A.H.

El tamaño de los reportes en papel pueden tener las siguientes dimensiones: 8 ½" X 11" y 15" X 11", en 1, 2 y 3 tantos.

**5.- Estándares para la Documentación**

El estándar para la documentación de todos aquellos diseños que se realicen (formas, reportes, especificaciones de programas, etc.) deberán cumplir con el siguiente formato (fig. 2.25.).

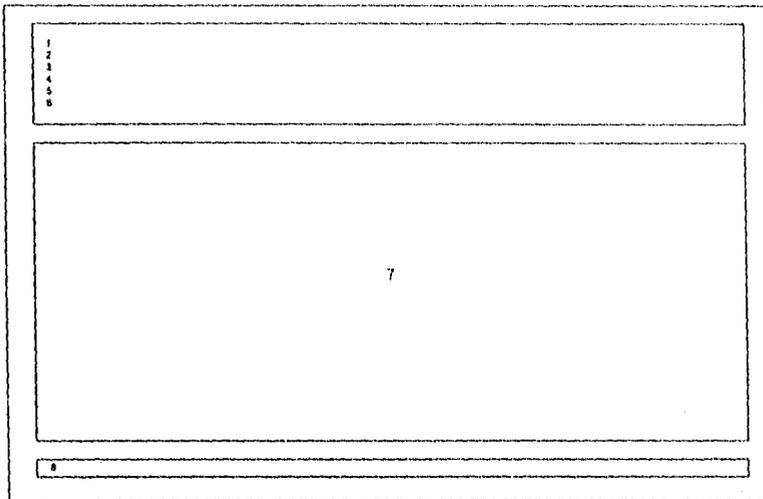


Fig. 2.25. Formato estándar para la documentación técnica del S.A.A.H.

en donde se especificará:

- |                |                   |
|----------------|-------------------|
| 1.- Sistema    | 5.- Objetivo      |
| 2.- Módulo     | 6.- Hoja          |
| 3.- Aplicación | 7.- Datos         |
| 4.- Nombre     | 8.- Observaciones |

### Diseño de Menús

Otra de las partes que deben quedar detalladas en esta etapa del diseño es la definición de menús que contendrá el sistema. Mediante los menús principales obtendremos una forma general de "navegar" dentro del sistema y a su vez, un esquema de dependencia entre formas. En las figuras siguientes (2.26., 2.27., 2.28., 2.29. y 2.30.) se describen los menús de cada módulo.

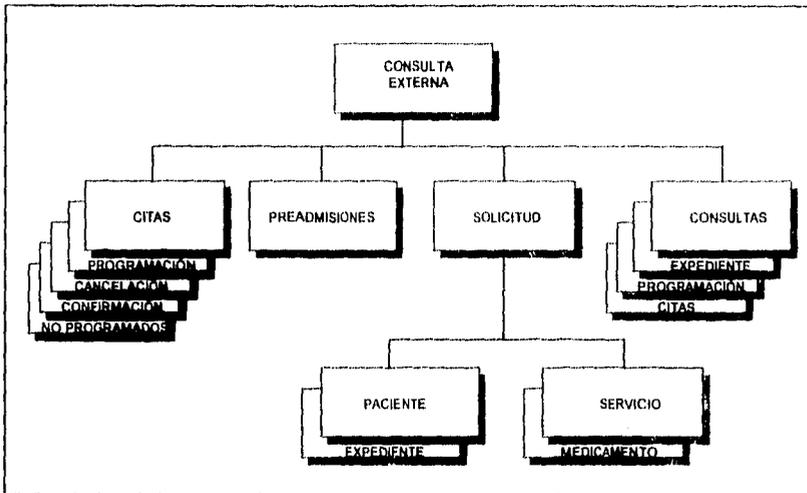


Fig. 2.26. Menú de Consulta Externa.

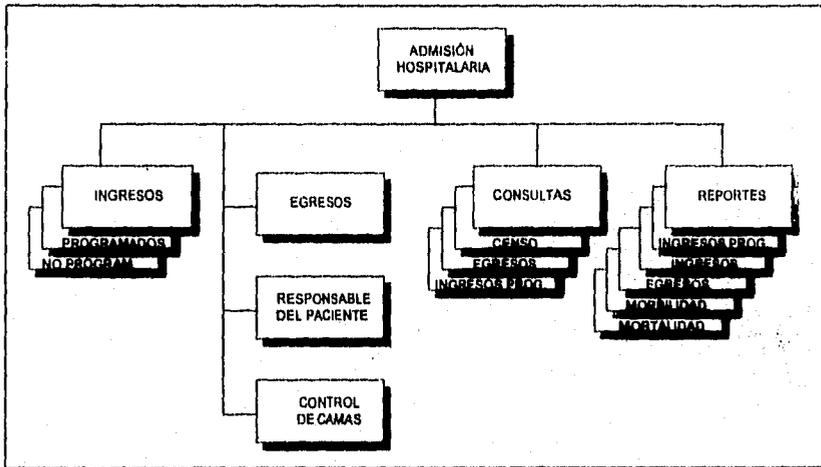


Fig. 2.27. Menú de Admisión Hospitalaria.

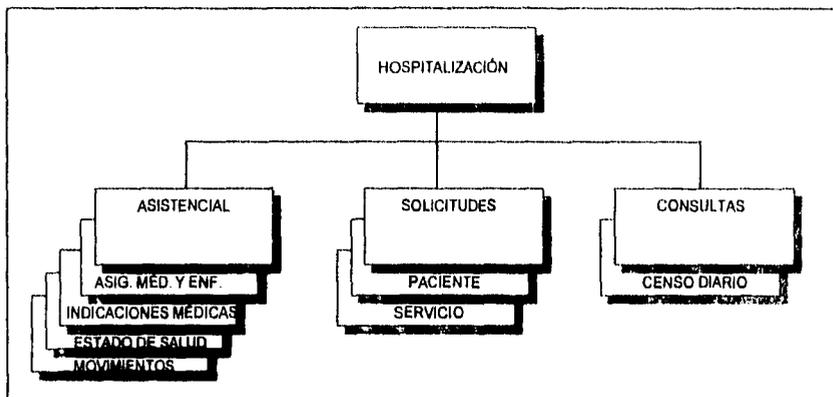


Fig. 2.28. Menú de Hospitalización.

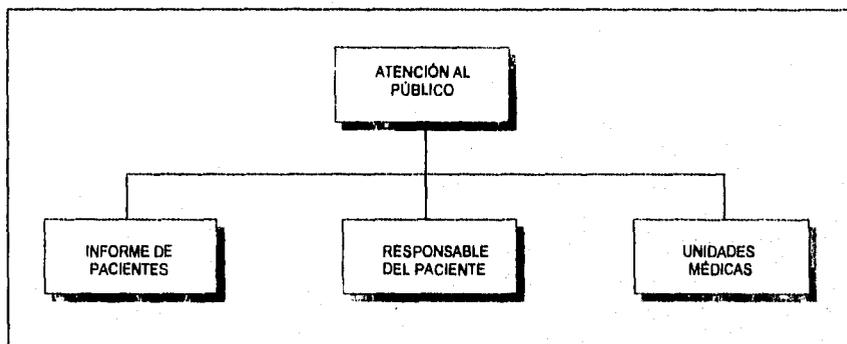


Fig. 2.29. Menú de Relaciones Públicas (Atención al Público).

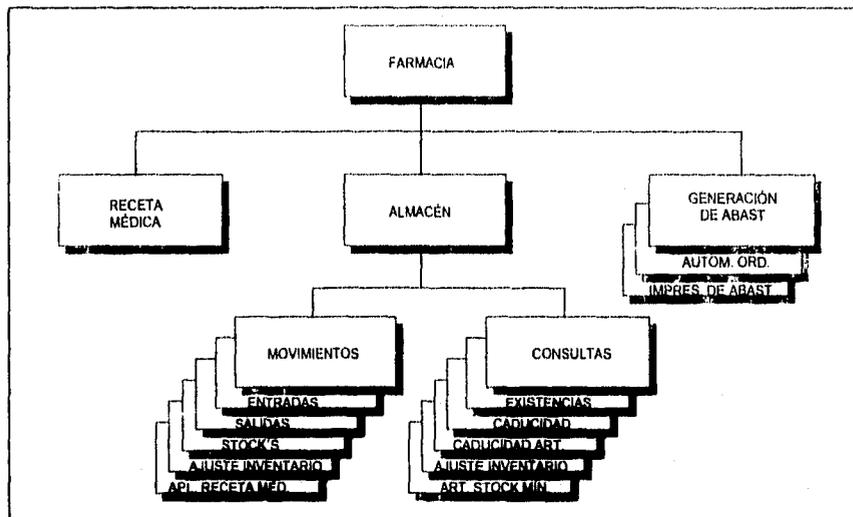


Fig. 2.30. Menú de Farmacia.

## Diseño de Formas y Reportes por Proceso

El diseño de formas y reportes es de vital importancia puesto que el programador no tendrá que distraerse en esta tarea, dedicándose completamente a su programación. A continuación se presenta el diseño de una forma (figura 2.31.) y un reporte (figura 2.32.) utilizados en el Sistema Automatizado para la Administración Hospitalaria. Parte de las demás formas y reportes se pueden consultar en el apéndice B.

Sistema: SAAH  
 Módulo: Consulta Externa  
 Aplicación: Forma (Pantalla)  
 Nombre: fca-ingprg (Ingresos Programados)  
 Objetivo: Registrar para el ingreso programado, la fecha y datos generales del paciente.

S.A.A.H. CONSULTA EXTERNA: INGRESO PROGRAMADO H.R.A.L.M.

EXPEDIENTE: [ ]	FECHA PROGRAMADA: [99/99/99]
ESPECIALIDAD: [ ]	
TIPO DE DERECHOHABIENTE: [ ]	
PATERO: [ ]	MATERO: [ ] NOMBRE: [ ]
SEXO: [ ]	EDO CIVIL: [ ] EDAD: [ ] CVE. EDAD: [ ]
CALLE/NUM.: [ ]	
COLONIA: [ ]	
ENT. FED.: [ ]	TELEFONO: [ ] C.P.: [ ]
OCUPACION: [ ]	
DEP. DE ADSCRIP.: [ ]	
CLINICA DE ADSCRIP.: [ ]	

OBSERVACIONES:

Fig. 2.31. Diseño de pantalla de registro para Ingresos Programados.

Sistema:	SAAH
Módulo:	Consulta Externa
Aplicación:	Reporte
Nombre:	la_ingprg (Ingresos Programados)
Objetivo:	Reportar los ingresos programados para la fecha especificada.

ISSSTE	SAAH
SUBDIRECCION GRAL. MEDICA	FEC.: 99/99/99
HOSPITAL. REG.	
LIC. ADOLFO LOPEZ MATEOS	

**CONSULTA EXTERNA**  
INGRESOS PROGRAMADOS PARA EL DIA 99/99/99

EXPEDIENTE	NOMBRE	EDAD	SEXO	ESPECIALIDAD
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]

**OBSERVACIONES:**

Fig. 2.32. Diseño de reporte de Ingresos Programados de Consulta Externa.

En la siguiente tabla se muestra una lista parcial de las formas generadas para el S.A.A.H.:

MÓDULO	NOMBRE	OBJETIVO
Admisión Hosp.	fam_coning.per	Confirmar el ingreso programado (Asignación de cama y no. de ingreso).
	fac_ingprg.per	Consulta de los Ingresos programados por fecha.
	faa_datpac.per	Registrar los datos generales del paciente.
	faa_datres.per	Registrar los datos generales del responsable del paciente.
	faa_egrpac.per	Registrar los datos referentes al egreso del paciente.
Consulta Externa	fca_solmed.per	Realizar la solicitud de medicamentos a Farmacia.
	fca_ingprg.per	Registrar el ingreso programado a piso desde Consulta Externa.
Farmacia	ffc_solpen.per	Consultar las solicitudes de medicamentos pendientes.
	ffa_sursol.per	Registrar los medicamentos surtido por cada solicitud.
Hospitalización	fha_asgmed.per	Registrar los datos de el médico y la enfermera del paciente.
	fha_edosal.per	Registrar y actualizar el estado de salud del paciente.
	fha_movpac.per	Registrar los movimientos del paciente en piso.
Relaciones Públicas	frc_conpac.per	Consultar la localización y estado de salud del paciente.

En la siguiente tabla se muestra una lista parcial de los reportes generados para el S.A.A.H.:

MÓDULO	NOMBRE	OBJETIVO
Admisión Hosp.	la_ingprg.4gl	Reportar los ingresos programados por periodo.
	la_hojaing.4gl	Generar la hoja de ingreso del paciente.
	la_bolinte.4gl	Generar la boleta de internamiento del paciente.
	la_ingtder.4gl	Genera la estadística de los ingresos por tipo de derechohabiente.
	la_ingedad.4gl	Genera la estadística de los ingresos por grupo de edad.
	la_egreso.4gl	Genera la hoja de egreso del paciente.
Farmacia	lf_solpen.4gl	Genera listado de solicitudes pendientes de surtir.
Hospitalización	lh_edosal.4gl	Generar el reporte del estado de salud del paciente.
	lh_solmed.4gl	Generar el reporte de la solicitud de medicamentos (También lo utiliza Consulta Externa).
	lh_morb.4gl	Genera la estadística de morbilidad.
	lh_mort.4gl	Genera la estadística de mortandad.

## Diseño Detallado

### Diccionario de Datos

Gran parte del valor de un diccionario de datos proviene del hecho de que se considera como una gramática casi formal para describir el contenido de los elementos de información que sirven de referencia para los analistas, diseñadores y programadores que están trabajando en un determinado proyecto.

El diccionario de datos define los elementos de información sin ambigüedades, además éste puede dar información sobre su función, la cual puede no ser inmediatamente obvia a partir de un examen de un diagrama de flujo de datos.

Para grandes sistemas basados en computadoras, el diccionario de datos crece rápidamente en tamaño y complejidad. De hecho es extremadamente difícil mantener un diccionario de datos manualmente. Por esta razón están disponibles varios sistemas de diccionario de datos automatizados. Además las nuevas estaciones de trabajo con herramientas CASE (*Computer Aid Software Engineering*) soportan la generación automática del diagrama de flujo de datos y el acoplamiento y gestión directa de los diccionarios de datos asociados.

Para la descripción del diccionario de datos del sistema S.A.A.H., se toma como referencia la siguiente notación:

Construcción de Datos	Notación	Significado
	**	Comentarios
	=	Está compuesto de ...
Secuencia	+	Unión de campos elementales
Selección	{   }	Uno u otro
Repetición	{ } <sup>n</sup>	n repetición de ...
	( )	Datos opcionales

Como el objeto es describir los datos manejados en el sistema, sus características, su uso y valores, a continuación sólo se describe una muestra de dichos datos.

### Definiciones Generales:

Las definiciones generales nos proporcionarán el apoyo necesario para la definición de construcciones más complejas. En la siguiente tabla se muestran estas definiciones:

Definición	Observaciones
caracter = [ a - z   0 - 9   ,   . ]	No se consideran las letras minúsculas puesto que el sistema las convertirá a mayúsculas.
numero = [ 0 - 9 ]	
fecha = día + mes + año	
día = { numero } <sup>2</sup>	Número = [ 1 - 31 ]
mes = { numero } <sup>2</sup>	Número = [ 1 - 12 ]
año = { numero } <sup>4</sup>	Número = [ 1990 - en adelante ]

### Definiciones Específicas:

En este apartado se describen las definiciones particulares para cada elemento que contendrá nuestra base de datos. La siguiente tabla muestra la definición de algunos de ellos.

Definición	Observaciones
<b>expediente =</b> rfc_paciente + homoclave rfc_paciente = { caracter } <sup>13</sup> rfc_paciente[ 1,4 ] = [ a - z ]	La Homoclave será obligatoria si ya existe un paciente con el mismo Registro Federal de Causantes (R.F.C.).
rfc_paciente[ 5,6 ] <= año (actual) rfc_paciente[ 7,8 ] = [ 1 - 12 ] rfc_paciente[ 9,10 ] = [ 1 - 31 ] homoclave = ( { caracter } <sup>3</sup> )	Año de nacimiento. Mes de nacimiento. Día de nacimiento.
<b>medico = { numero }</b>	Clave del Médico

<b>cama</b> = { numero }	
<b>estado_de_la_republica</b> = { numero } <sup>2</sup>	Número = { 1 - 32 }
<b>nombre_del_paciente</b> = paterno + materno + nombres paterno = { caracter } materno = { caracter } nombres = { caracter }	
<b>diagnostico_o_causa_del_ingreso</b> = {caracter }	Clave del diagnóstico de la Organización Mundial de la Salud
<b>edad_del_paciente</b> = edad + clave_de_la_edad edad = { numero } <sup>3</sup>	
<b>clave_de_la_edad</b> = [ A   M ]	A = Años , M = Meses
<b>sexo_del_paciente</b> = [ F   M ]	F = Femenino , M = Masculino
<b>fecha_de_ingreso</b> = fecha	Ver definición de Fecha en Definiciones Generales.
<b>telefono</b> = numero_telefonico + extension numero_telefonico = { numero } <sup>4</sup> extension = { numero }	
<b>parentesco_del_resp._del_paciente</b> = {caracter }	

## Definición de Tablas del S.A.A.H.

En la siguiente tabla se muestra una lista parcial de las tablas creadas para el S.A.A.H.:

NOMBRE	OBJETIVO
archivo_clinico	Tabla de archivo clínico.
ingresos	Tabla de ingresos.
movimiento_pac	Tabla de movimientos del paciente.
responsable_pac	Tabla del responsable del paciente.
egresos	Tabla del egreso hospitalario.
agenda_medica	Tabla de agenda de citas médicas.
camas	Catálogo de camas.
receta_medica	Tabla de recetas médicas.
solicitud_medic	Tabla de solicitud de medicamentos.
inventario_almac	Tabla del inventario de almacén general.
mov_almacen	Tabla de registro de movimientos del inventario.

A continuación se presenta la composición de cada una de las tablas del Sistema Automatizado para la Administración Hospitalaria.

SISTEMA : S.A.A.H.			
APLICACIÓN : Definición de Tablas			
NOMBRE : archivo_clinico			
CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Expediente	char(13)	No
paterno	Apellido Paterno	char(20)	
materno	Apellido Materno	char(20)	
nombre	Nombre del Paciente	char(20)	
sexo	Sexo	char(1)	No
edad	Edad	smallint	
estado_civil	Estado Civil del paciente	smallint	

ocupacion	Ocupación del Paciente	smallint	
cve_edad	Clave de la Edad	char(1)	
tipo_derech	Tipo de Derechohabiente	smallint	No
dependencia	Clave de la Dependencia	smallint	
adscrip	Clinica de Adscripción	smallint	
grp_sangre	Grupo Sanguineo	smallint	
cve_diag	Clave Diagnostico	char(3)	
fecha_ing	Fecha de Ingreso	date	No
cve_diag_ult	Clave diagnóstico Principal del último Egreso.	char(3)	
fecha_ult_egreso	Fecha del último Egreso	date	
cve_diag_egr_ce	Clave del diagnóstico Egreso de C.E.	char(3)	
cve_ult_serv	Clave del último Servicio- Hospitali.	smallint	
cve_ult_esp	Clave de la última Especialidad-Hosp.	smallint	
cve_ult_esp_ce	Clave de la última Especialidad con. Ext.	smallint	
fecha_ult_cita_ce	Fecha de última cita Consulta Ext	date	
sit_exped	Situación del Expediente	char(1)	No
fecha_aper_alta	Fecha de Apertura de Alta de Reg.	date	
serv_sol_aper	Servicio Solicitante de Apertura	smallint	
esp_sol_aper	Especialidad Solicitante de Apertura	smallint	
fecha_baja	Fecha de Baja	date	

SISTEMA : S.A.A.H.			
APLICACIÓN : Definición de Tablas			
NOMBRE : solicitud_medic			
CAMPO	DESCRIPCIÓN	TIPO	NULO
num_sol	Número de Solicitud	smallint	No
especialidad	Clave de la Especialidad del Solicitant	smallint	No
fecha_sol	Fecha de Solicitud	date	
cve_econom	Clave Económica	smallint	
cant_sol	Cantidad Solicitada	smallint	
cant_despach	Cantidad de Orden de Despacho	smallint	
fecha_sol_atend.	Fecha de Solicitud Atendida	date	

SISTEMA : S.A.A.H.			
APLICACIÓN : Definición de Tablas			
NOMBRE : ingresos			
CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Expediente del paciente	char(13)	No
num_ingreso	Número de Ingreso	smallint	No
ctrl_ingreso	Consecutivo de ingreso del Paciente	decimal (16,2)	
tipo_derech	Tipo de Derechohabiente	smallint	
edo_civil	Estado Civil	smallint	
edad	Edad	smallint	
cve_edad	Clave de la Edad (A/M)	char(1)	
ocupacion	Ocupación del Paciente	smallint	
estado	Estado de la República	smallint	
dependencia	Clave de la Dependencia	integer	
clinica	Clinica de Adscripción	integer	
procedencia	Especialidad de Procedencia	smallint	
situacion	Situación del Ingreso (P / H / E )	char(1)	No
fecha_ingreso	Fecha de ingreso	date	No

SISTEMA : S.A.A.H.			
APLICACIÓN : Definición de Tablas			
NOMBRE : movimiento_pac			
CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Expediente	char(13)	No
motiv_mov	Motivo del movimiento	smallint	
fec_mov	Fecha de movimiento	date	
hora_mov	Hora de movimiento	smallint	
min_mov	Minuto de movimiento	smallint	
num_cama	Número de cama que ocupa el pac.	smallint	
cve_medico	Clave médico responsable	smallint	
cve_espec	Clave de especialidad	smallint	

cve_diag	Clave diagnóstico	char(3)	
cve_diag_cau	Diagnóstico o causa del ingreso	char(3)	
edo_salud	Estado de salud	smallint	
cve_enfermera	Clave enfermera responsable	smallint	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : responsable\_pac

CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Expediente	char(13)	No
responsable	Nombre de responsable	char(25)	
calle_num	Calle y número	char(30)	
colonia	Colonia	char(30)	
cod_postal	Código postal	smallint	
telefono	Teléfono	char(10)	
parentesco	Parentesco	char(2)	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : egresos

CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Expediente del Paciente	char(13)	No
fecha_egreso	Fecha del Egreso	date	
hora_egreso	Hora del Egreso	smallint	
min_egreso	Minuto del Egreso	smallint	
ctrl_ingreso	Consecutivo de ingreso del Paciente	smallint	
motivo_egreso	Motivo del Egreso	smallint	
desti_egreso	Destino del Paciente	smallint	
cve_diag1	Clave del Diagnóstico Principal	char(2)	
cve_diag2	Clave del Diagnóstico Secundario	char(2)	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : agenda\_medica

CAMPO	DESCRIPCIÓN	TIPO	NULO
servicio	Servicio	smallint	No
espacialidad	Especialidad	smallint	No
cve_medico	Médico	smallint	No
dia	Día	smallint	
hora_inic	Hora de inicio	smallint	
min_inic	Minuto de inicio	smallint	
hora_sal	Hora de salida	smallint	
min_sal	Minuto de salida	smallint	
activ	Actividad	smallint	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : Camas

CAMPO	DESCRIPCIÓN	TIPO	NULO
expediente	Exped. Paciente que ocupa cama	char(13)	
num_cama	Número de cama	smallint	No
tipo_cama	Tipo de cama	smallint	
cve_serv	Clave de servicio	smallint	
cve_espec	Clave de la especialidad	smallint	
num_cuarto	Número de cuarto	smallint	
camasexo	Cama (tipo de sexo)	char(1)	
cama_censo	Cama censable (s/n)	char(1)	
cuartoaisla	Cuarto aislado (s/n)	char(1)	
sit_cama	Situación de la cama	smallint	
costo_cama	Costo día cama	smallint	
bano	Baño (s/n)	char(1)	
oxigeno	Oxígeno (s/n)	char(1)	
succion	Succión	char(1)	

unidad_transp	Unidad de trasplante (s/n)	char(1)	
unidad_sangre	Unidad de sangrado (s/n)	char(1)	
radium	Radium (s/n)	char(1)	
piso	Piso	smallint	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : receta\_medica

CAMPO	DESCRIPCIÓN	TIPO	NULO
fec_registro	Fecha de captura	date	No
grupo	Grupo	smallint	No
serie	Serie	char(8)	
folio_receta	Folio de receta medica	smallint	
num_medicam	Número de medicamento	smallint	
cve_medicam	Clave medicamento	smallint	
cantidad	Cantidad med.	smallint	
tipo_desp	Tipo de despacho med.	smallint	
sit_receta	Situación de la receta	smallint	
sit_acum	Situación del acumulado	smallint	

SISTEMA : S.A.A.H.  
 APLICACIÓN : Definición de Tablas  
 NOMBRE : inventario\_almac

CAMPO	DESCRIPCIÓN	TIPO	NULO
almacen	Clave del almacén	smallint	No
tipo_material	Tipo de material	smallint	
cve_econom	Clave económica	smallint	
exist_libro	Existencia en libros	smallint	
exist_fisica	Existencia física (último inventario)	smallint	
fec_inven	Fecha de inventario	date	
precio_unit	Precio unitario	smallint	
stock_max	Stock máximo	smallint	

stock_min	Stock mínimo	smallint	
per_resurt	Periodo de resurtido (solic. Ordinaria)	smallint	
rep_resurt_ex	Periodo de resurtido (solic. Extraordinaria)	smallint	
fec_ult_mov	Fecha del último movimiento	date	
ajustes	Ajustes	smallint	
exist_no_disp	Existencia no disponible	smallint	

SISTEMA : S.A.A.H.			
APLICACIÓN: Definición de Tablas			
NOMBRE : mov_almacen			
CAMPO	DESCRIPCIÓN	TIPO	NULO
cve_almacen	Clave del almacén	smallint	No
tipo_material	Tipo de material	smallint	
cve_econom	Clave económica	smallint	
lote_entrada	No. de lote de entrada	char(8)	
fec_caduc	Fecha de caducidad	date	
cve_proveedor	Clave del proveedor	smallint	
num_referen	No. de referencia o destino	smallint	
tipo_mov	Tipo de mov. (e/s)	smallint	No
fec_mov	Fecha del mov.	date	
precio_uni	Precio unitario	smallint	
mot_dev	Motivo de la devolución	smallint	
cve_unid_med	Clave de la unidad médica	smallint	

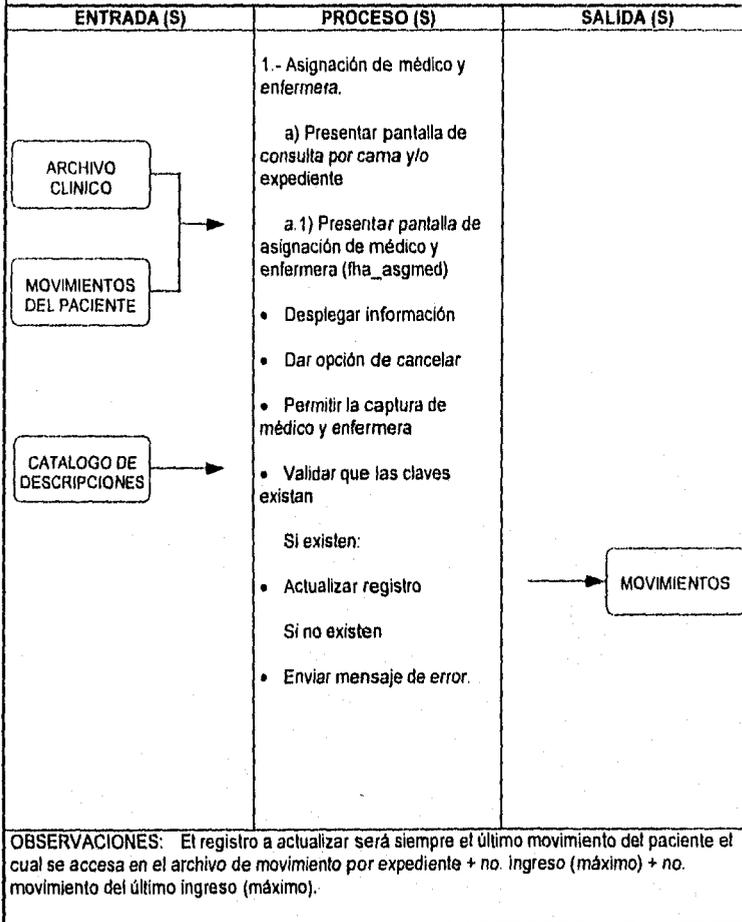
## Especificación de Programas

Con el objeto de explicar la secuencia de ejecución de instrucciones de los programas del Sistema Automatizado para la Administración Hospitalaria (S.A.A.H.), a continuación se presenta la especificación en seudocódigo estructural de algunos de ellos (Módulo de Consulta Externa, Hospitalización y Admisión Hospitalaria).

SISTEMA : S.A.A.H. MODULO : Consulta Externa APLICACIÓN: Descripción de Programas NOMBRE : mca_proin OBJETIVO : Registrar los ingresos programados en Consulta Externa HOJA : ½		
ENTRADA (S)	PROCESO (S)	SALIDA (S)
	1.- Altas de ingresos programados  a) Presentar pantalla de ingresos (fac_ingprg)  Solicitar llave expediente  a.1) Verificar si existe tomando en cuenta el mayor no. de ingreso.  Si existe <ul style="list-style-type: none"> <li>• Desplegar datos</li> <li>• Dar opción de cancelar</li> <li>• Dar opción de cambios</li> <li>• Grabar registro</li> <li>• Actualizar registro</li> </ul> De lo contrario:	
OBSERVACIONES:		

<p>SISTEMA : S.A.A.H.                  MODULO : Consulta Externa                  APLICACIÓN: Descripción de Programas                  NOMBRE : mca_proin                  OBJETIVO : Registrar los ingresos programados en Consulta Externa                  HOJA : 2/2</p>		
ENTRADA (S)	PROCESO (S)	SALIDA (S)
	<ul style="list-style-type: none"> <li>• Permitir captura de datos de ingreso grabando en el campo de situación de ingreso una "p" de ingreso programado.</li> <li>• Dar opción de cancelar</li> <li>• Grabar registro</li> <li>• Grabar registro</li> </ul>	<div style="display: flex; flex-direction: column; align-items: center; justify-content: center;"> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;">INGRESOS</div> <div style="border: 1px solid black; padding: 5px;">ARCHIVO CLINICO</div> </div>
OBSERVACIONES:		

**MODULO** : Hospitalización  
**APLICACIÓN:** Descripción de Programas  
**NOMBRE** : mha\_asigme  
**OBJETIVO** : Permitir asignar médico y enfermera responsables del paciente  
**HOJA** : 1/1



SISTEMA : S.A.A.H.  
 MODULO : Admisión Hospitalaria  
 APLICACIÓN: Descripción de Programas  
 NOMBRE : maa\_confin  
 OBJETIVO : Permitir ingresar a hospitalización los ingresos programados  
 HOJA : 1/3

ENTRADA (S)	PROCESO (S)	SALIDA (S)
	<p>1.- Confirmación de ingresos programados</p> <p>a) Presentar pantalla de consulta por fecha.</p> <p>a.1) Verificar si existen ingresos programados del día solicitado y aquellos registros cuya situación de ingreso sea "p".</p> <p>Si existen:</p> <ul style="list-style-type: none"> <li>• Presentar pantalla de ingreso confirmados (fam_coning)</li> <li>• Desplegar datos</li> <li>• Si la fecha es la del día en curso, permitir seleccionar un registro para confirmar su ingreso, presentando la pantalla de confirmación de ingresos (fam_confir).</li> <li>• Dar opción de cancelar</li> <li>• Dar opción de cambios</li> <li>• Si se confirman los cambios, presentar pantalla de datos (fam_dating), cuyos valores por default son fecha y la hora del sistema.</li> <li>• Dar opción de cancelar</li> </ul> <p>Si se confirma el ingreso:</p>	
OBSERVACIONES:		

SISTEMA : S.A.A.H.  
 MODULO : Admisión Hospitalaria  
 APLICACIÓN: Descripción de Programas  
 NOMBRE : maa\_confin  
 OBJETIVO : Permitir ingresar a hospitalización los ingresos programados  
 HOJA : 2/3

ENTRADA (S)	PROCESO (S)	SALIDA (S)
	<ul style="list-style-type: none"> <li>• Actualizar registro                      fecha=fecha ingreso.                      Situación de ingreso="h"                      asignarle su siguiente no.                      ingreso. Si no existe por                      default es 1.</li> <li>• Actualizar registro</li> <li>• Actualizar registro</li> <li>• Grabar registro                      Nota: el valor por default por                      cada ingreso para el campo                      que indica el número                      consecutivo de movimiento                      por paciente = 1</li> <li>b) Presentar rutina para                      preguntar si se desea registrar                      los datos del responsable.</li> <li>En caso afirmativo:</li> <li>• Presentar pantalla del                      paciente (faa_datres)</li> </ul>	<div style="display: flex; flex-direction: column; align-items: center;"> <div style="margin-bottom: 10px;">→ <span style="border: 1px solid black; padding: 2px 10px;">INGRESOS</span></div> <div style="margin-bottom: 10px;">→ <span style="border: 1px solid black; padding: 2px 10px;">ARCHIVO CLINICO</span></div> <div style="margin-bottom: 10px;">→ <span style="border: 1px solid black; padding: 2px 10px;">CAMAS</span></div> <div>→ <span style="border: 1px solid black; padding: 2px 10px;">MOVIMIENTOS DEL PACIENTE</span></div> </div>
OBSERVACIONES:		

SISTEMA : S.A.A.H.  
 MODULO : Admisión Hospitalaria  
 APLICACIÓN: Descripción de Programas  
 NOMBRE : maa\_confin  
 OBJETIVO : Permitir ingresar a hospitalización los ingresos programados  
 HOJA : 3/3

ENTRADA (S)	PROCESO (S)	SALIDA (S)
	<ul style="list-style-type: none"> <li>• Permitir captura de datos                      Nota: después de capturar el nombre del responsable, presentar rutina para preguntar si se desea registrar el mismo domicilio del paciente.</li> <li>• En caso afirmativo desplegar el domicilio del paciente en los campos correspondientes.</li> <li>• Permitir finalizar la captura</li> <li>• Grabar registro</li> </ul> <p>c) Emitir hoja de ingreso</p> <p>d) Emitir boleta de internamiento</p> <p>e) Emitir solicitud de expediente</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px; text-align: center;">RESPONSABLE DEL PACIENTE</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px; text-align: center;">r_inghos INGRESO HOSPITALARIO</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px; text-align: center;">r_bolint BOLETA DE INTERNAMIENTO</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">r_solexp SOLICITUD DE EXPEDIENTE</div>
OBSERVACIONES:		

En la siguiente tabla se muestra una lista parcial de los programas realizados para el S.A.A.H.:

MÓDULO	NOMBRE	OBJETIVO
Admisión Hosp.	mam_admhosp.4gl	Menú principal de Admisión Hospitalaria.
	maa_ingprg.4gl	Alta del ingreso del paciente (confirmación).
	mam_ingprg.4gl	Modificación del ingreso del paciente.
	maa_datpac.4gl	Registro de datos del paciente.
	maa_datres.4gl	Registro de datos del responsable del paciente.
	mab_ingbaj.4gl	Eliminación de ingresos no confirmados.
Consulta Externa	mcm_conext.4gl	Menú principal de Consulta Externa.
	mca_agenda.4gl	Registro de la agenda médica.
Farmacia	mfm_farm.4gl	Menú principal de Farmacia.
	mfa_solmed.4gl	Registro de solicitud de medicamentos.
	mfm_solmed.4gl	Modificación a la solicitud de medicamentos.
	mfb_solmed.4gl	Eliminación de solicitudes de medicamentos.
	mfa_surmed.4gl	Registro de medicamentos surtidos.
Hospitalización	mhm_hosp.4gl	Menú principal de Hospitalización.
	mhc_movpac.4gl	Consulta de movimientos del paciente.
	mha_egreso.4gl	Registro de egresos.
	mha_medico.4gl	Registro de médicos y enfermeras.
	mha_edosal.4gl	Registro del estado de salud del paciente.
	mhc_movpac.4gl	Cambios en el registro de movimientos del paciente.
	mha_diaoms.4gl	Registro de diagnósticos de la O.M.S.
	mha_camars.4gl	Registro de camas.
	mha_espec.4gl	Registro de especialidades.
	mha_hosp.4gl	Registro de hospitales.

Relaciones Públicas	mmm_relpub.4gl	Menú principal de Relaciones Públicas.
	mrc_datpac.4gl	Consulta de pacientes.
Rutinas Comunes	rc_cribus.4gl	Rutina de criterios de búsqueda.
	rc_busnom.4gl	Rutina de búsqueda por nombre.

### CAPITULO 3

# **FASE DE DESARROLLO DEL SAAH**

La fase de desarrollo del proyecto S.A.A.H. incluye principalmente la traducción de las especificaciones del diseño en código fuente, así como su depuración, documentación y pruebas.

El objetivo de esta fase es obtener un sistema en operación totalmente documentado y probado. Para ello, las principales actividades que se desarrollaron son las que se mencionan a continuación:

- Entender todas las especificaciones de programas para realizar su codificación, programación y prueba, mediante la utilización del *software* instalado (INFORMIX).
- Creación de la base de datos.

- Desarrollo de la programación.
- Preparar la documentación del sistema y de los programas, así como el manual de usuario, utilizando los estándares previamente establecidos.
- Preparación y ejecución de las pruebas de integración y aceptación del sistema.
- Planificación de la implantación.

## Programación, pruebas y documentación

Esta parte del desarrollo del sistema se fundamenta en las especificaciones de los programas derivadas del diseño.

La programación es la producción de código fuente que sea fácil de leer y comprender. Este objetivo puede alcanzarse haciendo el código fuente tan claro y sencillo como sea posible. Sencillez, claridad y elegancia son los sellos de los buenos programas.

Los objetivos principales de las pruebas de integración y aceptación son evaluar y mejorar la calidad de los distintos módulos del sistema generados durante el desarrollo y modificación del *software*.

La documentación interna del código fuente comienza con el seguimiento de la estandarización predeterminada para los nombres de los programas y de sus variables, continúa con la localización y la composición de los comentarios y termina con la organización visual del programa, cabe señalar que la claridad del código fuente facilitará la depuración, prueba y mantenimiento del mismo. En el listado 3.1., se presenta un listado con las especificaciones mencionadas.

El listado 3.1. que corresponde al programa principal del módulo de Admisión Hospitalaria, presenta el menú principal con las siguientes aplicaciones: ingresos, egresos, responsable del paciente, control de camas, consultas y reportes.



```

EXIT MENU
END MENU
CALL cierra_window(1)
CALL cierra_pres()
END MAIN
FUNCTION submenu_ingresos()
  DEFINE
    seguimos SMALLINT
  OPEN WINDOW v_ingresos AT 3, 4 WITH 2 ROWS, 73 COLUMNS
  OPTIONS INPUT WRAP
  LET seguimos = TRUE
  WHILE seguimos
    MENU " INGRESOS "
    COMMAND "Programados" " ~~~~~ "
    CALL submenu_programados()
    EXIT MENU
    COMMAND "No Programados" " ~~~~~ "
    EXIT MENU
    COMMAND "Terminar"
    LET seguimos = FALSE
    EXIT MENU
  END MENU
END WHILE
CLOSE WINDOW v_ingresos
END FUNCTION
FUNCTION submenu_programados()
  DEFINE
    seguimos SMALLINT,
    rut_param SMALLINT,
    hay_inter SMALLINT,
    opcion CHAR(1)
  OPEN WINDOW v_programados AT 3, 4 WITH 2 ROWS, 73 COLUMNS
  OPTIONS INPUT WRAP
  LET seguimos = TRUE
  LET rut_param = FALSE
  WHILE seguimos
    MENU " PROGRAMADOS "

```

Listado 3.1. Programa Principal del módulo de Admisión Hospitalaria.(Continua).

```

LET opcion = "C"
LET rut_param = TRUE
EXIT MENU
COMMAND "Reprogramacion" " ~~~~~ "
EXIT MENU
COMMAND KEY ("N") "caNcelacion" " ~~~~~ "
EXIT MENU
COMMAND "Terminar"
LET seguimos = FALSE
EXIT MENU
END MENU
IF seguimos THEN
  IF (rut_param) THEN
    CALL param_consulta("F") RETURNING hay_inter
    CALL mensaje(6, TRUE, FALSE, "Seleccionando Informacion . . .")
  END IF
  IF NOT hay_inter THEN
    CALL mensaje(1, FALSE, FALSE, "")
    CASE (opcion)
  WHEN "C" CALL cons_ingresos_conf()
  END CASE
  END IF
  END IF
END WHILE
CLOSE WINDOW v_programados
CALL cierra_window(2)
END FUNCTION

```

### Listado 3.1. Programa Principal del módulo de Admisión Hospitalaria.

El listado 3.2., presenta en un arreglo de once elementos (registros), los ingresos confirmados del periodo seleccionado. Para ello, despliega la forma que se muestra en la figura 3.3., la cual contiene los datos generales del paciente como son : expediente, nombre del paciente, sexo, edad y la fecha programada de ingreso.



```

DECLARE cur_ingresos SCROLL CURSOR FOR
  SELECT   ingresos.expediente, @apel_pat, @apel_mat, @nombre,
           sexo, edad, clave_edad, f_ingreso
  FROM     ingresos i, archivo_clinico a
  WHERE    i.expediente = a.expediente AND
           i.f_ingreso BETWEEN vg_inicial AND vg_final AND
           i.st_ingreso = "P"

  ORDER BY ingresos.f_ingreso
LET ind = 1
LET pos = 0
OPEN cur_ingresos
FETCH FIRST cur_ingresos INTO a_pac[ind].expediente, apel_pat,
                              apel_mat, nombre, a_pac[ind].sexo, a_pac[ind].edad,
                              a_pac[ind].clave_edad, a_pac[ind].f_ingreso
IF (STATUS = NOTFOUND) THEN
  CALL mensaje(5, TRUE, TRUE, "Ingresos Programados NO Localizados ...")
  CLOSE cur_ingresos
ELSE
  WHILE TRUE
    LET a_pac[ind].paciente = apel_pat CLIPPED, " ", apel_mat CLIPPED, " ",
                              nombre CLIPPED

    LET ind = ind + 1
    FETCH NEXT cur_ingresos INTO a_pac[ind].expediente, apel_pat,
                                  apel_mat, nombre, a_pac[ind].sexo, a_pac[ind].edad,
                                  a_pac[ind].clave_edad, a_pac[ind].f_ingreso
  IF (STATUS = NOTFOUND) THEN
    EXIT WHILE
  END IF

  END WHILE
  CLOSE cur_ingresos
  CALL set_count(ind - 1)
  IF (ind >= 1) THEN
    OPEN WINDOW Vent_Conflg AT 7,4 WITH FORM "fac_ingcon"
    ATTRIBUTE(BORDER, COMMENT LINE 1, FORM LINE 1, MESSAGE
              LINE LAST)

```

Listado 3.2. Programa de Consulta de Ingresos Confirmados del módulo de Admisión Hospitalaria. (Continúa).

```

OPTIONS
  ACCEPT KEY CONTROL-E,
  NEXT KEY CONTROL-N,
  PREVIOUS KEY CONTROL-P
IF (ind>11) THEN
  DISPLAY "CTRL+N " AT 15,02 ATTRIBUTE(REVERSE)
  DISPLAY "Avanza " AT 15,10 ATTRIBUTE(RED)
  DISPLAY "CTRL+P " AT 15,18 ATTRIBUTE(REVERSE)
  DISPLAY "Retrocede " AT 15,26 ATTRIBUTE(RED)
END IF
DISPLAY "CTRL+E " AT 15,39 ATTRIBUTE(REVERSE, RED)
DISPLAY "Consulta " AT 15,47 ATTRIBUTE(RED)
DISPLAY "RETURN " AT 15,58 ATTRIBUTE(REVERSE)
DISPLAY "Terminar" AT 15,66 ATTRIBUTE(RED)
DISPLAY ARRAY a_pac TO sc_ingresos.*
  ON KEY(CONTROL-M)
    EXIT DISPLAY
  ON KEY(CONTROL-E)
    LET pos = ARR_CURR( )
    LET scr_pos = SCR_LINE( )
    DISPLAY a_pac[pos].* TO sc_ingresos[scr_pos].*
    ATTRIBUTE(REVERSE, RED)
    CALL mensaje(6, TRUE, TRUE,
      "Localizando Ingresos Programados . . . ")
    CALL confirmacion_ingresos(a_pac[pos].*)
    DISPLAY a_pac[pos].* TO sc_ingresos[scr_pos].*
    ATTRIBUTE(RED)
  END DISPLAY
CLOSE WINDOW Vent_Confing
OPTIONS ACCEPT KEY ESC
END IF
END IF
END FUNCTION

```

Listado 3.2. Programa de Consulta de Ingresos Confirmados del módulo de Admisión Hospitalaria.

```

DATABASE saah
SCREEN SIZE 22 by 80
{
EXPEDIENTE  PACIENTE                               S  EDAD  FECHA
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
[f001      ]  [f002                                ] [a] [f3 ] [b] [f004 ]
}
END
TABLES
  ingresos
  archivo_clinico
ATTRIBUTES
f001 = ingresos.expediente,      COLOR = 3;
f002 = formonly.paciente ,      COLOR = 3;
a   = archivo_clinico.sexo,     COLOR = 3;
f3  = ingresos.edad,           COLOR = 3;
b   = ingresos.clave_edad,     COLOR = 3;
f004 = ingresos.f_ingreso ,    COLOR = 3, FORMAT = "dd/mm/yyyy";
END
INSTRUCTIONS
  DELIMITERS " "
  SCREEN RECORD sc_ingresos[11] (ingresos.expediente,
                                formonly.paciente,
                                archivo_clinico.sexo, ingresos.edad,
                                ingresos.clave_edad, ingresos.f_ingreso)
END

```

Fig. 3.3. Forma de Consulta de Ingresos Confirmados de Admisión Hospitalaria.

En la pantalla de la fig. 3.4. se presenta una corrida ejemplo de los programas de los listados 3.1. y 3.2. descritos anteriormente. En esta se puede observar el submenú de ingresos programados con sus respectivas opciones: Confirmación, Reprogramación y Cancelación y Terminar. Una vez elegida la opción de Confirmación, se presenta una rutina común para solicitar el período en el cual se desea consultar los ingresos programados. Posteriormente, se despliegan los datos del paciente mostrados en la pantalla, permitiendo elegir alguno digitando simultáneamente las teclas de CTRL y E para confirmar su ingreso.

Si el número de pacientes a desplegar en la pantalla excede la cantidad de registros definidos en el arreglo de la forma, entonces se despliegan las teclas de CTRL+N y CTRL+P para avance y retroceso rápido, respectivamente.

PROGRAMADOS:		Confirmacion	Reprogramacion	Cancelacion	Terminar
PERIODO:		[ 29 / 11 / 95 ]	[ 30 / 11 / 95 ]		
EXPEDIENTE	PACIENTE	S	EDAD	FECHA	
TOVA680125-MJ1	TORRES VAZQUEZ ALFREDO	M	27 A	29 / 11 / 95	
PAGC700327-TS3	PANIAGUA GONZALEZ CLAUDIA	F	24 A	29 / 11 / 95	
GATR650428-XC4	GONZALEZ ALLENDE ROBERTO	M	31 A	30 / 11 / 95	
CAMG660226-AA2	CAMACHO MENDOZA GUSTAVO	M	30 A	30 / 11 / 95	
<div style="display: flex; justify-content: space-between; font-family: monospace;"> <span>CTRL+N   Avanza</span> <span>CTRL+P   Retrocede</span> <span>CTRL+E   Consulta</span> <span>RETURN   Terminar</span> </div>					

Fig 3.4. Pantalla de Consulta de Ingresos Confirmados de Admisión Hospitalaria.

Una vez elegido el paciente en la pantalla de consulta de ingresos confirmados (fig. 3.4.), el programa del listado 3.1. hace una llamada a la función de confirmación de ingresos (listado 3.2.), la cual permite confirmar el ingreso o bien cancelarlo, si así se desea. Si se confirma envía como parámetros los siguientes datos:

- Expediente
- Nombre del paciente
- Sexo
- Edad
- Fecha programada de ingreso

El programa que se muestra en el listado 3.2. abre la forma de confirmación de ingreso (fig. 3.3.), en la cual despliega los datos generales del paciente permitiendo registrar y/o modificar algún dato no registrado en la captura de ingresos programados en Consulta Externa.

Durante la confirmación de datos, el programa despliega en la línea de teclas de ayuda, la tecla de CTRL+F para aquellos datos en que se puede acceder a la ventana de ayuda, la cual despliega los valores que puede tomar ese campo (dominio); en caso de que no se conozca la clave del dato a capturar y/o modificar.

En caso de no haber cancelación de ingreso, el programa hace la llamada a la función que permite capturar los datos de hospitalización del paciente, como son: especialidad, cama, fecha y hora de ingreso, etc.

Una vez registrado los datos de hospitalización, y en caso de no haber cancelación de ingreso, el programa actualiza los archivos de: archivo clínico, ingresos, movimientos del paciente y camas. Además pregunta si se desea registrar los datos del responsable; en caso afirmativo, hace la llamada a la función que permite registrar los datos del responsable del paciente, agregando un registro en la tabla de responsable del paciente.

Las funciones de captura de datos de hospitalización y la del registro del responsable del paciente, entre otras, así como sus formas correspondientes se muestran en el apéndice B.



```

b_edociv      SMALLINT,
b_ocupac      SMALLINT,
b_depen       SMALLINT,
b_clinica     SMALLINT,
desc_espec    CHAR(50),
desc_derech   CHAR(50),
desc_edociv   CHAR(50),
desc_ocupac   CHAR(50),
desc_depen    CHAR(50),
desc_clinica  CHAR(50),
hay_inter     SMALLINT
    
```

```

OPEN WINDOW v_cons_ings AT 5,4 WITH FORM "faa_coning"
ATTRIBUTE(BORDER, MESSAGE LINE LAST, FORM LINE 1)
OPTIONS
    
```

```

INPUT        WRAP,
ACCEPT       KEY ESC,
INPUT        ATTRIBUTE (RED),
DISPLAY      ATTRIBUTE (RED)
    
```

# Selecciono los datos generales del archivo\_clinico.

```

SELECT calle, colonia, codigo_postal, telefono
INTO rArc.*
FROM archivo_clinico
WHERE archivo_clinico.expediente = IConf.expediente
    
```

# Selecciono los datos complementarios del ingreso.

```

SELECT espec_proc, tipo_derech, estado_civil, ocupacion, dependencia,
clinica_ads
INTO rIng.*
FROM ingresos
WHERE ingresos.expediente = IConf.expediente AND
ingresos.st_ingreso = "P"
    
```

# Llamo la Rutina para Conocer la Descripciones de la Datos a Desplegar

```

IF (rIng.espec_proc IS NOT NULL) THEN
CALL desc_equiv_cat("C", 01, rIng.espec_proc)
RETURNING rIng.espec_proc, desc_espec
END IF
    
```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continua).

```

IF (rIng.dependencia IS NOT NULL) THEN
    CALL desc_equiv_cat("C", 02, rIng.dependencia)
    RETURNING rIng.dependencia, desc_depen
END IF

IF (rIng.clinica_ads IS NOT NULL) THEN
    CALL desc_equiv_cat("C", 03, rIng.clinica_ads)
    RETURNING rIng.clinica_ads, desc_clinica
END IF

IF (rIng.tipo_derech IS NOT NULL) THEN
    CALL desc_equiv_cat("E", 01, rIng.tipo_derech)
    RETURNING rIng.tipo_derech, desc_derech
END IF

IF (rIng.estado_civil IS NOT NULL) THEN
    CALL desc_equiv_cat("E", 02, rIng.estado_civil)
    RETURNING rIng.estado_civil, desc_edociv
END IF

IF (rIng.ocupacion IS NOT NULL) THEN
    CALL desc_equiv_cat("E", 03, rIng.ocupacion)
    RETURNING rIng.ocupacion, desc_ocupac
END IF

DISPLAY "CTRL+B " AT 17,45 ATTRIBUTE(REVERSE)
DISPLAY "Cancela" AT 17,53
DISPLAY "ESC " AT 17,62 ATTRIBUTE(REVERSE)
DISPLAY "Procesa" AT 17,67
DISPLAY IConf.expediente, IConf.f_ingreso, desc_espec, IConf.paciente,
desc_edociv, desc_ocupac, desc_depen, desc_clinica
TO
expediente, f_ingreso, desc_espec, paciente, sexo, edad,
clave_edad, desc_derech, desc_edociv, desc_ocupac,
desc_depen, desc_clinica
# Bandera de interrupcion de opcion
LET hay_inter = FALSE

```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continúa)

```
INPUT      IConf.edad, IConf.clave_edad, rArc.calle, rArc.colonia,
           rArc.codigo_postal, rArc.telefono, rlng.tipo_derech,
           rlng.estado_civil, rlng.ocupacion, rlng.dependencia, rlng.clinica_ads
```

```
WITHOUT DEFAULTS FROM
```

```
        edad, clave_edad, calle, colonia, codigo_postal, telefono,
        tipo_derech, estado_civil, ocupacion, dependencia, clinica_ads
```

```
#Se define tecla para ayuda de campo
```

```
ON KEY(CONTROL-F)
```

```
    CASE
```

```
        WHEN INFIELD (tipo_derech) CALL cons_equiv_cat("E", "01")
            RETURNING rlng.tipo_derech, desc_derech
```

```
        LET b_derech = TRUE
```

```
        DISPLAY rlng.tipo_derech, desc_derech TO tipo_derech, desc_derech
```

```
        WHEN INFIELD (estado_civil) CALL cons_equiv_cat("E", "02")
            RETURNING rlng.estado_civil, desc_edociv
```

```
        LET b_edociv = TRUE
```

```
        DISPLAY rlng.estado_civil, desc_edociv TO estado_civil, desc_edociv
```

```
        WHEN INFIELD (ocupacion) CALL cons_equiv_cat("E", "03")
            RETURNING rlng.ocupacion, desc_ocupac
```

```
        LET b_ocupac = TRUE
```

```
        DISPLAY rlng.ocupacion, desc_ocupac TO ocupacion, desc_ocupac
```

```
        WHEN INFIELD(dependencia) CALL cons_equiv_cat("C", "02")
            RETURNING rlng.dependencia, desc_depen
```

```
        LET b_depen = TRUE
```

```
        DISPLAY rlng.dependencia, desc_depen TO dependencia, desc_depen
```

```
        WHEN INFIELD (clinica_ads) CALL cons_equiv_cat("C", "03")
            RETURNING rlng.clinica_ads, desc_clinica
```

```
        LET b_clinica = TRUE
```

```
        DISPLAY rlng.clinica_ads, desc_clinica TO clinica_ads, desc_clinica
```

```
    END CASE
```

```
AFTER    FIELD edad
```

```
    IF    (IConf.edad IS NULL OR IConf.edad > 115) THEN
           CALL mensaje (6, TRUE, TRUE, "Edad Invalida . . .")
           NEXT FIELD edad
```

```
    END IF
```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continúa).

```

AFTER  FIELD clave_edad
  IF    (IConf.clave_edad IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Edad Invalida . . .")
        NEXT FIELD clave_edad
  END IF

AFTER  FIELD calle
  IF    (rArc.calle IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique la Calle . . .")
        NEXT FIELD calle
  END IF

AFTER  FIELD colonia
  IF    (rArc.colonia IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique la Colonia . . .")
        NEXT FIELD colonia
  END IF

AFTER  FIELD codigo_postal
  IF    (rArc.codigo_postal IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique elCodigo Postal . . .")
        NEXT FIELD codigo_postal
  END IF

BEFORE FIELD tipo_derech
  LET  b_derech = FALSE
  CALL tecla_ayuda()

AFTER  FIELD tipo_derech
  IF    (rIng.tipo_derech IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique el Tipo de
        Derechohabiente . . .")
        NEXT FIELD tipo_derech
  END IF

```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continua).

```

IF NOT b_derech THEN
    CALL limpia_tecla()
    CALL desc_equiv_cat("E", 01, rlng.tipo_derech)
        RETURNING rlng.tipo_derech, desc_derech
    DISPLAY desc_derech TO desc_derech
END IF

BEFORE FIELD estado_civil
    LET b_edociv = FALSE
    CALL tecla_ayuda("D")

AFTER FIELD estado_civil
    IF (rlng.estado_civil IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique el Estado Civil ...")
        NEXT FIELD estado_civil
    END IF
    IF NOT b_edociv THEN
        CALL tecla_ayuda("Q")
        CALL desc_equiv_cat("E", 02, rlng.estado_civil)
        RETURNING rlng.estado_civil, desc_edociv
        DISPLAY desc_edociv TO desc_edociv
    END IF

BEFORE FIELD ocupacion
    LET b_ocupac = FALSE
    CALL tecla_ayuda("D")
AFTER FIELD ocupacion
    IF (rlng.ocupacion IS NULL) THEN
        CALL mensaje (6, TRUE, TRUE, "Verifique la Ocupacion ...")
        NEXT FIELD ocupacion
    END IF
    IF NOT b_ocupac THEN
        CALL tecla_ayuda("Q")
        CALL desc_equiv_cat("E", 03, rlng.ocupacion)
        RETURNING rlng.ocupacion, desc_ocupac
        DISPLAY desc_ocupac TO desc_ocupac
    END IF

```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continua).

```

BEFORE  FIELDdependencia
        LET  b_depen = FALSE
        CALL tecla_ayuda("D")

AFTER   FIELDdependencia
        IF   (rIng.dependencia IS NULL) THEN
            CALL mensaje (6, TRUE, TRUE, "Verifique la Dependencia . . .")
            NEXT FIELD dependencia
        END IF
        IF   NOT b_depen THEN
            CALL tecla_ayuda("Q")
            CALL desc_equiv_cat("C", 02, rIng.dependencia)
            RETURNING rIng.dependencia, desc_depen
            DISPLAY desc_depen TO desc_depen
        END IF

BEFORE  FIELDclinica_ads
        LET  b_clinica = FALSE
        CALL tecla_ayuda("D")

AFTER   FIELD clinica_ads
        IF   (rIng.clinica_ads IS NULL) THEN
            CALL mensaje (6, TRUE, TRUE, "Verifique la Clinica de
            Adscripcion . . .")
            NEXT FIELD clinica_ads
        END IF
        IF   NOT b_clinica THEN
            CALL tecla_ayuda("Q")
            CALL desc_equiv_cat("C", 03, rIng.clinica_ads)
            RETURNING rIng.clinica_ads, desc_clinica
            DISPLAY desc_clinica TO desc_clinica
        END IF

        ON  KEY(INTERRUPT)
        LET int_flag = FALSE
        LET hay_inter = TRUE
        EXIT INPUT
    
```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continua).

```

ON KEY(CONTROL-B)
LET hay_inter = TRUE
EXIT INPUT
END INPUT

IF NOT hay_inter THEN
CALL mensaje (5, TRUE, TRUE, "Capture los Datos de
Hospitalizacion.")
SELECT MAX(num_ingreso)
INTO sig_ingreso
FROM ingresos
WHERE ingresos.expediente = IConf.expediente
IF (sig_ingreso IS NULL) THEN
LET sig_ingreso = 1
ELSE
LET sig_ingreso = sig_ingreso + 1
END IF

BEGIN WORK
UPDATE archivo_clinico
SET (calle, colonia, codigo_postal, telefono) = (rArc.*)
WHERE archivo_clinico.expediente = IConf.expediente
UPDATE ingresos
SET (tipo_derech, estado_civil, ocupacion, dependencia,
clinica_ads, st_ingreso, num_ingreso)
= (rIng.tipo_derech, rIng.estado_civil, rIng.ocupacion,
rIng.dependencia, rIng.clinica_ads, "H", sig_ingreso)
WHERE ingresos.expediente = IConf.expediente AND
ingresos.st_ingreso = "P"

CALL datos_hospitalizacion(IConf.*)
# vg_confing: bancaria de confirmacion de ingreso
IF NOT vg_confing THEN
ROLLBACK WORK
ELSE
COMMIT WORK
END IF

```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria (Continua).

```

ELSE
    CALL mensaje (6, TRUE, TRUE, "Confirmacion de Ingreso Cancelada.")
END IF
CLOSE WINDOW v_cons_ings
END FUNCTION

FUNCTION tecla_ayuda(accion)
    DEFINE accion char (1)
    IF (opcion="D") THEN
        DISPLAY "CTRL+F " AT 17,27 ATTRIBUTE(REVERSE, RED)
        DISPLAY "Ayuda" AT 17,35
    ELSE
        DISPLAY " " AT 17,27
        DISPLAY " " AT 17,35
    END IF
END FUNCTION

```

Listado 3.5. Programa Principal del módulo de Admisión Hospitalaria.

```

DATABASE saah
SCREEN SIZE 22 by 80
{
EXPEDIENTE      :{f001      }[d1      ]
ESPECIALIDAD   :[d2      ]
FECHA PROGRAMADA :{f002      ] SEXO :{a      ] EDAD :{f03][b ]
CALLE           :{f004      ]
COLONIA         :{f005      ] CP :{f006      ]
TELEFONO       :{f007      ]
DERECHOHABIENTE :{f8      ] [d3      ]
ESTADO CIVIL    :{f9      ] [d4      ]
OCUPACION       :{f10     ] [d5      ]
DEPENDENCIA     :{f011    ] [d6      ]
CLINICA DE ADSCRIPCION :{f012 ] [d7      ]
}
END

```

Fig.3.6. Forma de Admisión Hospitalaria.

TABLES

ingresos,  
 archivo\_clinico

ATTRIBUTES

f001 = ingresos.expediente, NOENTRY;  
 d1 = FORMONLY.paciente, NOENTRY;  
 d2 = FORMONLY.desc\_espec, NOENTRY;  
 f002 = ingresos.f\_ingreso, NOENTRY;  
 a = archivo\_clinico.sexo, NOENTRY;  
 f03 = ingresos.edad;  
 b = ingresos.clave\_edad, UPSHIFT, INCLUDE = ("A", "M"),  
 COMMENTS = " A = A&OS M = MESES ";  
 f004 = archivo\_clinico.calle, UPSHIFT;  
 f005 = archivo\_clinico.colonia, UPSHIFT;  
 f006 = archivo\_clinico.codigo\_postal;  
 f007 = archivo\_clinico.telefono, UPSHIFT;  
 f8 = ingresos.tipo\_derech;  
 d3 = formonly.desc\_derech, NOENTRY;  
 f9 = ingresos.estado\_civil;  
 d4 = formonly.desc\_edociv, NOENTRY;  
 f10 = ingresos.ocupacion;  
 d5 = formonly.desc\_ocupac, NOENTRY;  
 f011 = ingresos.dependencia;  
 d6 = formonly.desc\_depen, NOENTRY;  
 f012 = ingresos.clinica\_ads;  
 d7 = formonly.desc\_clinica, NOENTRY;

END

Fig 3.7. Forma de Confirmación de Ingresos (Datos Generales del Paciente) del Módulo de Admisión Hospitalaria.

En el diseño del sistema se contempló el desarrollo de un submódulo de ayuda de campo para aquellos datos que así lo requerían. Esto, con el fin de conocer la descripción de los valores que pueden tener dichos datos. Como por ejemplo el dato de estado civil, se sabe que puede tomar varios valores (instancias), los cuales podrían ser: soltero, casado, divorciado, unión libre, etc.

La ventaja de este submódulo de ayudas es que el usuario puede modificar en cualquier momento dichos valores, agregando, modificando y/o eliminando los que desee, sin que esto repercuta en la programación.

En las figura 3.8. y 3.9., se muestra la forma de confirmación de ingresos (Datos Generales) con una ventana que presenta las ayudas de campo.

PROGRAMADOS:	Confirmacion	Reprogramacion	Cancelacion	Terminar	
EXPEDIENTE:	[TOVA680125-MJ1][TORRES VAZQUEZ ALFREDO ]				
ESPECIALIDAD	:[ORTOPEDIA ]				
FECHA PROGRAMADA	:[11 29 1995]	SEXO	:[M]	EDAD	:[ 27 ] [ A ]
CALLE	:[MORELOS 15 ]				
COLONIA	:[MIGUEL HIDALGO ]		CP	:[ 14410 ]	
TELEFONO	:[ 6660537 ]				
DERECHOHABIENTE	:[ 2 ]	{ NO DERECHOHABIENTE }			
ESTADO CIVIL	:[ 1 ]	{ SOLTERO }			
OCUPACION	:[ 1 ]	{ LIDER DE PROYECTOS }			
DEPENDENCIA	:[ 2 ]	{ GEO NEW YORK LIFE, S.A. }			
CLINICA DE ADSCRIPCION	:[ 2 ]	{ H.R.A.L.M. }			
CTRL+B    Cancela    ESC    Procesa					

Fig. 3.8. Pantalla del módulo de Confirmación de Ingreso de Admisión Hospitalaria.

SAAH		MODULO DE ADMISION HOSPITALARIA		HRAIM	
PROGRAMADOS:	Confirmacion	Reprogramacion	Cancelacion	Terminar	
EXPEDIENTE:	[TOVA680125-MJ1]	[TORRES VAZQUEZ ALFREDO	]		
ESPECIALIDAD	:[ORTOPEDIA		]		
FECHA PROGRAMADA	:[11 29 1995]	SEXO : [M]	EDAD : [ 27 ]	[ A ]	
CALLE	:[MORELOS 15		]		
COLONIA					
TELEFONO					
DERECHOHABIENTE					
ESTADO CIVIL					
DEPENDENCIA					
CLINICA DE ADSCRIPCION					

VALOR	DESCRIPCION
1	PERSONAL HRAIM
2	NO DERECHOHABIENTE

ESC Selecciona

Fig. 3.9. Pantalla de Confirmación de Ingresos de Admisión Hospitalaria, que presenta la ventana de ayuda de campo.

## Pruebas del sistema

Esta es una etapa crítica del sistema, en donde se verifica la calidad y desempeño del proyecto realizado mediante diferentes tipos de pruebas.

Estas pruebas al sistema implicaron dos clases de actividades: pruebas de integración y de aceptación.

## Pruebas de integración

Las pruebas de integración que se aplicaron al S.A.A.H. fueron de dos tipos: Ascendente y Descendente.

La primera es una estrategia tradicional empleada para integrar los componentes de cualquier *software* en un todo funcionando. Se divide por módulos, seguidos por pruebas de subsistemas y luego por pruebas del sistema completo.

La segunda empieza por la rutina principal y una o dos rutinas inmediatamente subordinadas en la estructura del sistema.

Ambas pruebas fueron necesarias al término del desarrollo de los programas. Estas se aplicaban desde los menús hasta módulos de cada subsistema.

Para efectuar las pruebas de integración del sistema se tomaron en cuenta las siguientes consideraciones.

- 1) Probar cada módulo del S.A.A.H. en forma independiente.
- 2) Utilizar datos reales, así como datos de prueba introduciendo también datos incongruentes e incorrectos para probar los procesos de validación de entradas.
- 3) Probar integralmente al sistema por medio del análisis de los resultados de cada uno de sus procesos.

Estas pruebas arrojaron la suficiente información para la corrección del código y la correcta integración del sistema. También, este procedimiento permitió la localización de los errores solamente en los nuevos módulos y no en los anteriores ya probados.

## Pruebas de aceptación

Implican la planificación y la ejecución de pruebas funcionales, de desempeño y de tensión para demostrar que el sistema implantado satisface los requisitos.

Además de las pruebas funcionales y de desempeño, se llevaron también al cabo las pruebas de tensión, éstas con el fin de establecer las limitaciones del sistema.

Las pruebas funcionales y de desempeño determinan la cantidad de tiempo de ejecución empleado, la eficiencia global del programa y el tiempo de respuesta. Las pruebas de tensión son aquellas diseñadas para romper en forma intencional la ejecución del programa.

Para este caso, las pruebas de aceptación incorporaron casos de prueba desarrollados durante las pruebas de integración.

Las pruebas se realizaron en conjunto con el usuario para determinar los ajustes necesarios, los cuales fueron básicamente de forma y no de fondo (pantallas principalmente). Las pruebas consistieron en:

- **Pruebas de captura.** Registro de información en varias terminales simultáneamente.
- **Pruebas de validación.** El usuario probó las diferentes pantallas con el fin de verificar la validación de datos.
- **Tiempos de respuesta.** Básicamente se realizaron sobre pantallas de actualización.
- **Impresión de Formatos.** Se mandó a imprimir diferentes formatos al mismo tiempo.
- **Pruebas de volumen.** Se generó información ficticia con el fin de cargar los archivos de la base de datos y probar su comportamiento, tanto del sistema como del equipo.

Los resultados obtenidos con estas pruebas permitieron realizar ajustes mínimos en procedimientos que agilizarían el funcionamiento del sistema, tratando de evitar los cuellos de botella en horas pico.

Cabe señalar que estas pruebas se hicieron en condiciones críticas como son:

- Ejecución de la misma aplicación en varias terminales a la vez.
- Enviando a impresión varios reportes a una sola impresora.
- Haciendo consultas en paralelo en cinco terminales (lo normal es una terminal).

Las condiciones normales de operación nunca podrán ser iguales a las de prueba, por lo que se asegura un óptimo funcionamiento del sistema.

CAPITULO 4

# FASE DE IMPLANTACIÓN DEL SAAH

La implantación del Sistema Automatizado para la Administración Hospitalaria deberá cubrir, desde la aceptación del diseño probado hasta la operación en modo satisfactorio del nuevo sistema, soportada por los correspondientes manuales de operación y usuario.

Para esta última fase del proyecto se llevó un control especial para la realización de las siguientes actividades:

- Plan de capacitación a usuarios.
- Control de los recursos.
- Operación del sistema por el usuario.
- Conversión al nuevo sistema.
- Liberación final del sistema.

## Plan de capacitación a Usuarios

En esta tarea se consideraron a los usuarios de las áreas afectadas por el nuevo sistema.

Se desarrolló un plan de capacitación para los usuarios indicando en él: los usuarios a ser formados, el o los responsables de su formación, sobre que temas, lugar y fecha donde se iba a efectuar la capacitación.

En cuanto a los usuarios elegidos para la formación, se formó un grupo reducido de cada área, con el propósito de que éstos a su vez enseñaran la operación de sus módulos al resto de sus compañeros.

El plan de capacitación se llevó a cabo apoyándonos en el manual de operación y el manual de usuarios, además de las ayudas *on-line* con las que dispone el sistema. Esto con el objeto de que el usuario dispusiera de ayudas en todo momento y de información sobre la(s) operación(es) a realizar, así como facilitar esta ardua labor.

## Control de Recursos

En esta parte fue importante planificar con todo detalle, el calendario de instalación de todos los medios necesarios para arrancar el sistema: *hardware*, instalaciones, materiales, adecuación de locales, etc.

Las actividades que se llevaron a cabo para este fin fueron las siguientes:

- Revisión del plano físico para conocer la localización física de cada terminal.
- Señalización en el plano del ducto para el cableado.
- Revisión del sistema de potencia para la instalación de las terminales.
- Banderización de cada terminal para conocer su ubicación.
- Instalación de terminales.

En total se instalaron 21 terminales en las distintas áreas contempladas en el diseño del sistema. Su distribución se muestra en la tabla 4.1.

Módulo	No. Terminales	Distribución
Consulta Externa	8	1 por piso (7) 1 Jefatura
Hospitalización	7	1 por piso (5) 1 Terapia Intensiva y 1 Unidad de Terapia Intensiva de Pedriatría
Farmacia	2	1 Mostrador 1 Almacén
Relaciones Públicas	1	1 Atención al Público
Coordinación de Enfermería	1	1 Oficina de Coordinación
Coordinación de Servicios al Derechohabiente	1	1 Oficina de Coordinación
Subdirección Médica	1	1 Oficina de la Subdirección

Fig. 4.1. Distribución de terminales del S.A.A.H. (fase I).

La instalación de *hardware* (cableado, terminales, etc.) se llevó a cabo por la compañía seleccionada en la licitación para la adquisición del equipo de cómputo.

## **Conversión de Datos**

Con el fin de conformar los catálogos del nuevo sistema, se solicitó a los usuarios del hospital que proporcionaran entre otros, los siguientes catálogos:

- Servicios
- Especialidades
- Médicos y enfermeras
- Diagnósticos
- Unidades médicas
- Medicamentos
- Proveedores de farmacia

También se conformaron los catálogos de equivalencias (dominio de los campos) del sistema.

## **Operación del Sistema por el Usuario**

Una vez conformados los catálogos y equivalencias del sistema, y la instalación de las terminales, se llevó a cabo una operación piloto por parte del usuario, utilizando los datos de un período previo. Esto con el fin de verificar los resultados obtenidos.

## **Conversión del Nuevo Sistema**

La conversión del nuevo sistema se llevó en paralelo, es decir, se continuó operando con el sistema manual, al mismo tiempo que se iba integrando gradualmente el nuevo sistema.

## **Liberación Final del Sistema**

Finalmente, la liberación del sistema se llevó a cabo con la entrega formal de éste a los usuarios y al personal del área de sistemas que se encarga de procesar los sistemas en producción (liberados).

Se entregó también, al grupo de mantenimiento de sistemas, toda la documentación técnica, que se conoce como memoria del desarrollo del sistema.

# RESULTADOS Y CONCLUSIONES

Evidentemente, los centros hospitalarios del I.S.S.S.T.E., constituyen hoy en día uno de los pilares del sector salud en México, por lo que resulta imprescindible contar en la actualidad, con sistemas más avanzados para lograr un buen funcionamiento de éstos, que repercuta en una mejor atención a la derechohabencia.

En adición, conforme sigan surgiendo nuevas aplicaciones en computación y en otras áreas, que permitan mejorar nuestro estilo de vida - equipo médico, reservaciones áreas, sistemas financieros, bancos de información médica, control de procesos, guías de navegación, etc.- continuaremos apoyándonos en el desarrollo de estas nuevas tecnologías.

Sin embargo, con la importancia que esto representa, el hablar de un desarrollo pleno y propio en cuanto a informática médica se refiere es aún muy prematuro. Por lo que se requiere realizar mayores esfuerzos para el establecimiento de sistemas computacionales integrales que permitan apoyar la administración y operación de los hospitales de nuestro país (principalmente los del sector público).

Indudablemente, para garantizar el establecimiento de estos nuevos sistemas, se requiere de una disciplina tecnológica y administrativa que permita desarrollar sistemáticamente éstos, de manera que sean los suficientemente económicos y eficaces.

La ingeniería de *software* nos proporciona ese enfoque sistemático para la producción y mantenimiento de nuevos productos de *software* que se deberán desarrollar y modificar a tiempo y dentro de la estimación de costos, con el fin de mejorar la calidad de estos nuevos productos, incrementando así, la satisfacción del cliente.

El enfoque establecido por la ingeniería de *software* consiste básicamente en la utilización de técnicas de resolución de problemas comunes a todas las ramas de la Ingeniería; las cuales sientan las bases para la planificación y desarrollo de nuevos proyectos. La única diferencia entre esta Ingeniería y las otras, es que no existen leyes físicas para la programación, la intangibilidad del producto y lo oculto de las interfaces entre los diversos módulos de programación.

Fundamentalmente, el Sistema Automatizado para la Administración Hospitalaria, fue desarrollado aplicando el enfoque sistemático en el que se basa la ingeniería de *software*. Para ello, se siguió el ciclo de vida tradicional para el desarrollo de un producto de *software* (análisis, diseño, desarrollo e implantación). Basándonos además en metodologías conocidas que nos permitieron conceptualizar el sistema. Dichas técnicas consistieron en dividir éste en módulos (unidades funcionales) independientes los cuales se comunican entre sí compartiendo la información de una base de datos dentro de un ambiente de seguridad y en tiempo real; generando de esta manera un banco de información médica confiable que podrá ser utilizado principalmente por el personal médico y de enfermería.

El nuevo sistema representa una invaluable herramienta durante la planificación, operación y control de los recursos del hospital, permitiendo una mejor administración y optimización de los mismos.

Mediante esta nueva herramienta se abatirán costos, gracias al control de las principales operaciones -asignación eficaz de personal médico, control de medicamentos, administración de camas, programación de citas, etc.-, a la reducción de procedimientos y al mejoramiento del flujo de información, con lo que se logrará un incremento sustancial en la productividad de los hospitales.

En esta etapa del proyecto no se determinaron los costos de la atención médica: costo por paciente, costo por servicio, costo por diagnóstico, etc., ya que se integrarán en la segunda etapa, con el fin de brindar en forma oportuna a la dirección del hospital información que servirá como base para una mejor planificación y una correcta toma de decisiones.

Asimismo en la segunda etapa del proyecto se pretende generar las estadísticas en forma tabular y gráfica, ya que son de vital importancia en un hospital. Esto se logrará integrando al sistema la interfase gráfica de Informix (Wingz).

Finalmente, cabe señalar que la segunda etapa se llevará a cabo bajo el mismo esquema de desarrollo de la primera, concluyendo así el proyecto final.

## BIBLIOGRAFÍA

Gane Chris y Sarion Trish  
ANÁLISIS ESTRUCTURADO DE SISTEMAS  
Librería " El Ateneo ", Editorial, Argentina 1992.

Márquez Vite Juan Manuel  
SISTEMAS DE INFORMACIÓN POR COMPUTADORA  
METODOLOGÍA DE DESARROLLO  
Editorial: Trillas, México 1987.

Pressman Roger S.  
INGENIERÍA DE SOFTWARE  
UN ENFOQUE PRÁCTICO  
Editorial: Mc.Graw-Hill, España 1990.

López-Fuensalida Antonio  
METODOLOGÍA DE DESARROLLO  
PRODUCCIÓN AUTOMÁTICA DE SOFTWARE CON HERRAMIENTAS CASE  
Editorial: Macrobit Editores, México 1991.

Fairley Richard  
INGENIERÍA DE SOFTWARE  
Editorial: Mc.Graw-Hill, México 1990.

Leffler Jonathan  
USING INFORMIX-SQL  
Editorial: Adison-Wesley y Publishing Company, Gran Bretaña 1992.

Pérez Peregrina Jesús Efrén  
LOS EXPERTOS EN SISTEMAS  
Editorial: Limusa, México 1986.

## APÉNDICE A

### ¿ PORQUÉ INFORMIX ?

El presente apéndice no es un manual de introducción a la programación en **INFORMIX**, supone conocimientos de los conceptos básicos de programación como son: variables, cláusulas de asignación, ciclos y funciones, por lo que sólo se pretende definir y mostrar las características básicas de **INFORMIX 4GL** e **INFORMIX SQL**.

#### ¿Qué es Informix ?

Es un lenguaje de cuarta generación, desarrollado por **RDS** (*Relational Database Systems, Inc.*), y diseñado específicamente para desarrollo de aplicaciones de Bases de Datos Relacionales. Está basado en **RDSQL** (*Rapid Development Structured Query Language*), una extensión de **SQL** (*Structured Query Language*), desarrollado por IBM.

**INFORMIX-4GL** puede trabajar con **INFORMIX-SQL**. Este último es creado bajo **RDSQL** y es compatible con los estándares establecidos por **ANSI** (*American National Standard Institute*), para las implementaciones de **SQL**.

#### Características Propias de Informix

- 1.- **Manejo de módulos de programación.** La construcción de sistemas de información se hace con base a una serie de módulos independientes, los cuales pueden ser ligados en cierto momento.
- 2.- **RDSQL.** Para interactuar con la base de datos, se cuenta con un conjunto poderoso de instrucciones muy simples para insertar, borrar, modificar y consultar la información. Este conjunto es proveído por el lenguaje de datos **RDSQL**.
- 3.- **Uso de formas de pantalla.** Con solo definir en un editor de textos convencional los formatos de las pantallas y después compilarlos, se cuenta con presentaciones de alta calidad para manejar la interacción con el usuario.

**4.- Creación de reportes.** Cuenta con unas estructuras muy simples de programación, pero de gran potencial para manejar cortes de control, sumalizaciones, control de formatos de página, encabezados y pie de páginas, etc., lo que permite generar reportes con una gran calidad.

**5.- Edición y ejecución de instrucciones en línea.** En **INFORMIX** se puede editar y ejecutar una serie de instrucciones sin pasar por compilación, lo que proporciona un poder de programación amplio en el momento de la ejecución de cualquier sistema.

Estas y otras características están disponibles para tener un sistema de información simple de programación y poderoso en utilidad, todo esto gracias a este lenguaje de cuarta generación.

### Arquitectura de Informix

La arquitectura de este producto se muestra en la figura A.1.

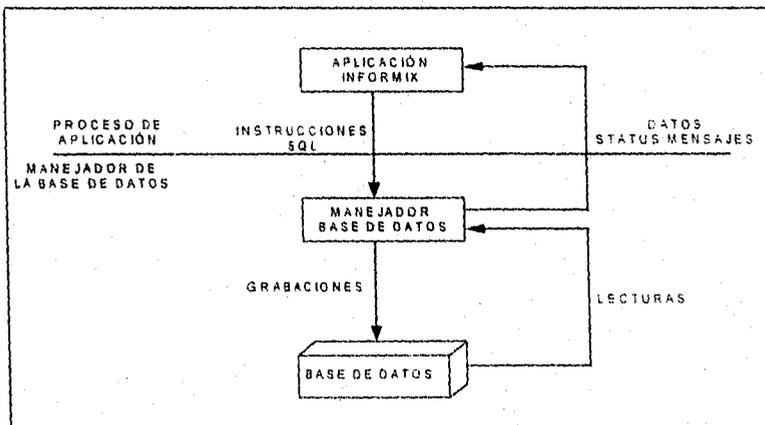


Fig. A.1. Arquitectura de Informix.

## Estructura del Producto

El manejador de base de datos de Informix cuenta con un lenguaje de definición de datos (DDL: *Data Definition Language*) y con un lenguaje de manipulación de datos (DML: *Data Manipulation Language*)

### Lenguaje de Definición de Datos

El lenguaje de definición de datos está integrado por las declaraciones que constituyen la base de datos y son un subconjunto de RDSQL. Las declaraciones básicas de este lenguaje se describen a continuación:

- 1.- Creación de la base de datos, con la siguiente declaración:

```
CREATE DATABASE nombre_db [WITH LOG IN "nombre-ruta"]
```

- 2.- Crea la tabla de la base de datos, con la siguiente sintaxis:

```
CREATE [TEMP] TABLE nombre_tabla
(nombre_columna tipo_dato [NOT NULL],...)
[IN "nombre_ruta"]
```

- 3.- Crea el índice, con la siguiente sintaxis:

```
CREATE [UNIQUE | DISTINCT] [CLUSTER] INDEX
nombre_indice ON nombre_tabla (nombre_columna [ASC | DESC].)
```

### 1.- Creación de una Base de Datos

En el prompt (\$) del sistema operativo, se llama al programa **INFORMIX-SQL** al escribir el siguiente comando:

```
isql
```

Si inicia desde el menú principal (superior), al oprimir la tecla <d> para seleccionar la opción de la base de datos (**Database**) y <c> para seleccionar la

opción de creación (**Create**); así la pantalla que solicita el nombre de la base de datos tiene la siguiente presentación:

CREATE DATABASE >>

Enter the name you want to assign to the new database

----- Press CTRL-W for Help -----

Después, proporcionar el nombre con el que se identificará a la base de datos y, si se crea con éxito, **INFORMIX-SQL** retorna al menú anterior.

Por último al oprimir <e> regresa al menú principal.

## Diccionario de la Base de Datos

Al crearse con éxito la base de datos, **INFORMIX-SQL** le da la extensión **.dbs** como un subdirectorio del directorio actual.

Cabe acotar que, aunque no se use como directorio **nombre.dbs** o cualquier contenido de éste, hay que cuidarse de no alterar o borrar ninguno de estos archivos, ni cambiar los permisos asignados por **UNIX** o los propietarios.

Borrar o alterar cualquiera de estos sistemas de archivos pueden dañar la integridad de la base de datos.

## 2.- Creación de una Tabla

Antes de crear una tabla es necesario que sea planeada documentalmente en papel, y considerando los siguientes aspectos:

- Nombres de tablas
- Nombres de columnas
- Tipos de datos
- Valores NULL
- Índices

## Nombres de las tablas

Para obtener las tablas se siguen las siguientes reglas:

- Por cada tabla de la base de datos se debe dar un nombre único.
- Sólo pueden usarse un máximo de 18 caracteres.
- Los nombres deben comenzar con un carácter, el resto del nombre puede consistir en letras, números, y guiones inferiores (\_).
- No hay distinción entre letras mayúsculas o minúsculas.
- Los nombres no pueden ser palabras reservadas de **INFORMIX-SQL**.

## Nombres de las Columnas

En el caso de los nombres de las columnas, se siguen los mismos criterios que en los nombres de las tablas, además de la siguiente consideración:

El nombre de las columnas debe ser único dentro de una tabla, sin embargo, el mismo nombre puede ser usado en más de una tabla.

## Tipos de Datos

<b>SMALLINT</b>	Números enteros desde -32,767 a +32,767
<b>INTEGER</b>	Números enteros desde -2,147,483,647 a, +2,147,483,647.
<b>SERIAL</b>	Almacenado como <b>INTEGER</b> , los números son asignados secuencialmente por <b>RDSQL</b> .
<b>SMALLFLOAT</b>	Números de punto-flotante con 8 dígitos significativos.
<b>FLOAT</b>	Número de punto-flotante de hasta 16 dígitos significativos.
<b>DECIMAL</b>	Números de punto-flotante con un máximo de 32 dígitos significativos, en este caso el formato específico de uso está compuesto por precisiones, que es el número total de dígitos, y escala, que es el número de dígitos después del punto decimal; para ambos casos el default es 16,2.

<b>MONEY</b>	Almacenado como un <b>DECIMAL</b> , despliega 2 dígitos después de punto decimal y un \$.
<b>DATE</b>	Almacenado como un <b>INTEGER</b> positivo representando el número de días desde Dic. 31, 1899. Almacenado como un <b>INTEGER</b> negativo antes de Dic, 31, 1899.
<b>CHAR (long)</b>	Para caracteres strings ( cadenas de caracteres). La longitud máxima debe ser especificada, en columnas de tipo carácter que pueden almacenar cualquier combinación de letras, dígitos, caracteres especiales o espacios.

### Valores NULL

Los valores **NULL** son usados como sustitutos de lugar en una columna, cuando no hay valor apropiado que insertar durante un registro de datos.

**INFORMIX-SQL** permite ingresar valores nulos (**NULL**) en columnas de la base de datos, como la alternativa del ingreso de un "0" en una columna numérica o un blanco en una columna carácter.

### Creación de una Tabla por RDSQL

El editor de esquema es una interface fácil de usar para crear tablas de la base de datos. El esquema puede pensarse como la estructura de una tabla de la base de datos, y puede llamar al editor de esquema como se muestra abajo.

Inicia en el menú principal de **INFORMIX-SQL**; oprimir <t> para tabla, seleccionar la base de datos a utilizar, y el siguiente menú será desplegado:

```
TABLE: Create Alter Info Drop Exit
Create a new table.
-----Press CONTROL-W for Help-----
```

<b>Create</b>	Despliega el editor de esquema para crear una nueva Tabla.
<b>Alter</b>	Permite usar el editor de esquema para para modificar Tablas existentes, aún si éstas contienen datos.
<b>Info</b>	Provee de información acerca de la estructura de la Tabla.
<b>Drop</b>	Borra una Tabla y todos sus datos, de la base de datos.
<b>Exit</b>	Regresa al menú principal de <b>INFORMIX-SQL</b> .

Desde el menú de tabla, oprimir <c> para crear (**Create**) una nueva tabla, y escribir "**customer**" como el nombre de la tabla y oprimir <RETURN>; ahora estará colocado en el editor de esquema,

## Índices

Un índice es una lista ordenada de valores de datos creada por **RDSQL** para acelerar la búsqueda y ordenación de los datos, y se pueden indexar columnas durante el proceso de creación de la tabla.

## Tipos de Índices

Un índice permite el ingreso de valores duplicados en una columna, si existe un índice distinto o único no se permiten valores duplicados.

## Lenguaje de Manipulación de Datos

En el lenguaje de manipulación de datos hay cuatro declaraciones con sus respectivas opciones; su uso es para ingresar, borrar y manipular los datos almacenados en la base de datos, las cuales se presentan a continuación:

- 1.- Inserción de filas, con la siguiente sintaxis:

```
INSERT INTO nombre_tabla [(lista - columnas)]
(VALUES (lista.valores)) | declaración SELECT
```

- 2.- Actualización de filas, con la siguiente sintaxis:

```
UPDATE nombre_tabla SET [(lista_columnas = expr [...]) |
(lista_columnas) | *] = (lista_expr)
[WHERE condición]
```

- 3.- Eliminación de filas con la siguiente sintaxis:

```
DELETE FROM nombre_tabla [WHERE condición]
```

- 4.- Localización de filas, con la siguiente sintaxis:

```
SELECT cláusula FROM cláusula

[WHERE cláusula]
[GROUP BY cláusula]
[HAVING cláusula]
[ORDER BY cláusula]
[INTO TEMP cláusula]
```

## Uso del Editor RDSQL

Una vez que el usuario ha creado una base de datos, puede insertar, borrar, actualizar y seleccionar datos (**INSERT, DELETE, UPDATE, SELECT**) al usar **RDSQL**.

Iniciar desde el menú principal - **INFORMIX-SQL**, como se muestra abajo:

```
INFORMIX-SQL:Form report Query-language User-menu
Database Table Exit
Run, Modify, Create, or Drop a form.
----- Press CTRL-W for Help -----
```

Seleccionar la opción **Query-language**, presionar <q>, así como la base de datos a utilizar y aparecerá la siguiente pantalla:

```
RDSQL: New Run Modify Use-editor Output Choose Save Info
Drop exit
Enter new RDSQL statements using RDSQL editor.
----- Press CTRL-W for Help -----
```

Las diez opciones disponibles desde el menú **RDSQL** son:

<b>New</b>	Lo coloca en el editor <b>RDSQL</b> de tal manera que pueda ingresar a una o más declaraciones <b>RDSQL</b> y oprima la tecla <b>ESC</b> para salir del editor.
<b>Run</b>	Ejecuta las declaraciones.
<b>Modify</b>	Lo coloca en el editor <b>RDSQL</b> , de tal manera que pueda cambiar las declaraciones actuales.
<b>Use-Editor</b>	Permite seleccionar un editor como <b>vi</b> .
<b>Output</b>	Envía los resultados de una declaración <b>RDSQL</b> a impresora, archivo o tubo ( <b>PIPE</b> ).
<b>Save</b>	Permite salvar un comando <b>RDSQL</b> en un archivo con el nombre que guste; la extensión <b>.sql</b> es agregada automáticamente y los archivos que contienen declaraciones <b>RDSQL</b> , son llamados archivos de comandos.

## Programas en Informix

Un programa en **INFORMIX-4GL** puede residir en un solo archivo del sistema operativo o en múltiples archivos del mismo, a estos archivos se les llaman módulos.

Los programas **INFORMIX-4GL** consisten en las siguientes rutinas: **MAIN**, **FUNCTION** y **REPORT**; una o en varias rutinas pueden residir en un módulo (archivo de sistema operativo), además estos módulos pueden ser ligados para crear un programa ejecutable,

Todo programa **INFORMIX-4GL** debe tener una sola rutina **MAIN**, **MAIN** que dirige el flujo del proceso, el cual debe ser terminado con las palabras **END MAIN**.

En su aplicación podrá contener un archivo con las variables **GLOBALS** del sistema, definición de **FORM** y mensajes de ayuda para el usuario **HELP**, o puede ser ligado a códigos fuentes **INFORMIX-RSQL/C** (si tiene **INFORMIX-ESQL/C**), código **C**, o módulos escritos en lenguaje ensamblador.

### Secciones de un Programa en Informix-4gl

<b>GLOBALS</b>	Pueden existir varias de ellas en un conjunto de programas a ligarse.
<b>MAIN</b>	Debe existir solamente una declaración.
<b>FUNCTION</b>	Pueden existir varias declaraciones.
<b>REPORT</b>	Pueden existir varias declaraciones.

### Módulos Informix-4gl

Los programas en **INFORMIX-4GL** pueden consistir en uno o más archivos, estos archivos son llamados módulos; en caso de tener un solo archivo **INFORMIX-4GL** lo denominará como módulo sencillo.

Cada módulo **INFORMIX-4GL** debe tener la extensión **.4gl**.

Un programa **INFORMIX-4GL** no está restringido respecto al formato, pues con la estructuración hace a los programas más fáciles de entender, escribir y depurar.

Se pueden incluir comentarios en los programa **INFORMIX-4GL**, al preceder la línea con el signo (#) o al encerrar los comentarios entre llaves ({}).

Programa **INFORMIX-4GL** en un sólo módulo.

```

GLOBALS
....
END GLOBALS
MAIN
....
END MAIN
FUNCTION add_cust()
....
END FUNCTION
    
```

Programa **INFORMIX-4GL** en multi-módulos

archivo1.4gl	archivo2.4gl
GLOBALS	GLOBALS
...	archivo1.4gl...
END GLOBALS	FUNCTION cust_menu
MAIN	...
...	END FUNCTION
END MAIN	REPORT labels()
FUNCTION init()	...
...	END REPORT
END FUNCTION	

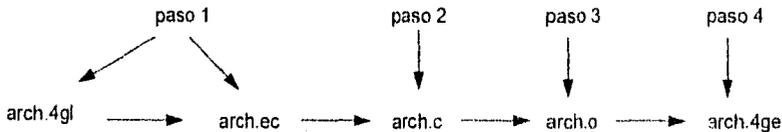
### Como Trabaja la Compilación de Informix-4gl

Las siguientes operaciones son llevadas a cabo en el curso de la preparación y ejecución de un programa **INFORMIX-4GL**.

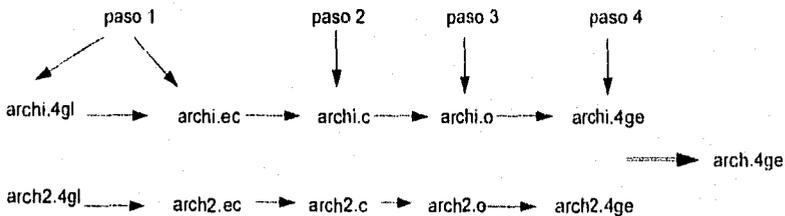
1. **INFORMIX-4GL** preprocesa el código 4gl para producir el código **INFORMIX-ESQL/C** el cual es quien finaliza con la extensión .ec.

2. El código **INFORMIX-ESQL/C** se preprocesa para producir el código de lenguaje **C**, el archivo de código **C** termina con la extensión **.c**.
3. El código de lenguaje **C** se compila para dejar un archivo objeto que termina con la extensión **.o**.
4. Así, el archivo objeto es ligado con las librerías de **INFORMIX-ESQL/C**, además de con sus propias librerías para generar un programa ejecutable.

### Compilación y Ligado de un Módulo Simple



### Compilación y Ligado de Varios Módulos



### Compilación de Módulos Informix-4gl

Los programas **INFORMIX-4GL** pueden ser compilados desde la línea de comando o a través del menú, con manejo del medio ambiente de programación.

**Desde la línea de comando:**

Para preprocesar, compilar y ligar un programa **4gl**, digitar el comando:

`c4gl fuente.4gl -o archivo_salida.4ge`

Donde **fuentes.4gl** es el archivo fuente y **archivo\_salida.4ge** es el nombre que le desee designar al programa ejecutable.

- Usar la opción **-e** para sólo preprocesar ( **archivo.c** ) y prevenir la compilación y ligado.
- Si no hay instrucción **MAIN** en el código **fuentes.4gl**, el código será compilado a **fuentes.o** pero no ligado.
- Listar cualquier otro archivo fuente **ESQL/C**, archivos objeto, archivos de código **C** y sus propias librerías especiales en la línea de comandos.

Sintaxis:

`c4gl fuente,4gl [-e] [otros_argumetos] -o archivo salida  
[otros.ec...] [otros,c...] [otros.o...] [-l tu librería...]`

- otros argumentos, incluye argumentos de compilación -c

Es importante mencionar que si no especifica el archivo de salida, entonces el archivo default es **a.out**. La conversión es nombrar los programas ejecutables **INFORMIX-4GL** con la extensión **.4ge**.

## Desde el Medio Ambiente de Programadores

Teclar **i4gl** en la línea de comando para obtener el menú principal del sistema.

```
INFORMIX-4GL : Module Form Program Query-Language Exit
Run, Modify, Create, or Drop a module.
----- Press CTRL-W for Help -----
```

- El menú permite compilar módulos individuales para crear código objeto, también permite compilar y ligar módulos simples.

- El módulo **FORMS** permite crear, modificar y compilar formas.
- El menú **PROGRAM** permite definir y ligar programas en módulos múltiples juntos.
- La opción **QUERY LANGUAGE** puede ser ejecutada sólo cuando tiene **INFORMIX-SQL** instalado en su sistema, y permite usar Instrucciones **RDSQL** sin salir del ambiente de **INFORMIX-4GL**.

## Los Menús **MODULE**

MODULE: Modify New Compile Program-Compile Run Exit  
 Modify a module.  
 ----- Press CTRL-W for Help -----

<b>MODIFY</b>	Lo coloca en un editor para modificar un archivo <b>.4gl</b> existente.
<b>NEW</b>	Lo coloca en un editor para crear un archivo <b>.4gl</b> nuevo.
<b>COMPILE</b>	Lo lleva a un menú que le permite compilar módulos.
<b>PROGRAM-COMPILE</b>	Le permite compilar programas( módulos múltiples ) de aplicaciones <b>INFORMIX-4GL</b> que ya han sido creadas.
<b>RUN</b>	Le permite la ejecución de programas compilados.

MODIFY MODULE: Compile Save-and-exit Discard-and-exit  
 Compile a module.  
 ----- Press CTRL-W for Help -----

COMPILE MODULE: Object Runnable Exit  
 Compile a module to object code.  
 ----- Press CTRL-W for Help -----

MODIFY MODULE: Correc Exit  
 Compile a module.  
 ----- Press CTRL-W for Help -----

<b>OBJECT</b>	Le crea sólo un archivo objeto; no liga un archivo y escribe el archivo objeto en el archivo con la extensión <b>.o</b> .
<b>RUNABLE</b>	Le crea un programa ejecutable, compila y liga; use esta opción para compilar un programa de módulo simple y le escribe el archivo objeto en el archivo con la extensión <b>.o</b> y el archivo ejecutable, si hay alguno, lo manda al archivo con la extensión <b>.Age</b> .

## APÉNDICE B

### DISEÑO DE FORMAS, REPORTES Y PROGRAMACIÓN

A continuación se presentan algunas formas diseñadas para el Sistema Automatizado para la Administración Hospitalaria.

Sistema:	SAAH
Módulo:	Rutina General
Aplicación:	Forma (Pantalla)
Nombre:	rc_cribus
Objetivo:	Solicitar información del paciente a través de su expediente.

SAAH	CRITERIOS DE BUSQUEDA	HRALM
------	-----------------------	-------

EXPEDIENTE: [            ]
----------------------------

OBSERVACIONES:

Fig. B.1. Forma de diseño de pantalla para Criterios de Búsqueda por Expediente.

Sistema: SAAH  
 Módulo: Rutina General  
 Aplicación: Forma (Pantalla)  
 Nombre: rc\_criper  
 Objetivo: Solicitar información mediante la selección de rangos de fechas.

SAAH	CRITERIOS DE BUSQUEDA	HRALM
<div style="border: 1px solid black; padding: 5px; display: inline-block;">                     PERIODO: [                      ] AL [                      ]                 </div>		

OBSERVACIONES:

Fig. B.2. Forma de diseño de pantalla para Criterios de Búsqueda por selección de rangos de Fechas.

**Sistema:** SAAH  
**Módulo:** Rutina General  
**Aplicación:** Forma (Pantalla)  
**Nombre:** rc\_criesp  
**Objetivo:** Solicitar información del paciente a través de la especialidad.

SAAH	CRITERIOS DE BUSQUEDA	HRAIM
<div style="border: 1px solid black; padding: 5px; display: inline-block;">                     ESPECIALIDAD: [   ] [   ]                 </div>		
OBSERVACIONES:		

Fig. B.3. Forma de diseño de pantalla para Criterios de Búsqueda por Especialidad.

Sistema: SAAH  
 Módulo: Rutina General  
 Aplicación: Forma (Pantalla)  
 Nombre: rc\_busnom  
 Objetivo: Solicitar información del paciente a través de su nombre.

SAAH	CRITERIOS DE BUSQUEDA			HRALM																		
<table border="1"> <tr> <td>NOMBRE: [</td> <td>]</td> <td>[</td> <td>]</td> <td>[</td> <td>]</td> </tr> <tr> <td></td> <td>PATERNO</td> <td>MATERNO</td> <td></td> <td>NOMBRE</td> <td></td> </tr> <tr> <td>CAMA: [</td> <td>]</td> <td>EXPEDIENTE: [</td> <td></td> <td>]</td> <td></td> </tr> </table>					NOMBRE: [	]	[	]	[	]		PATERNO	MATERNO		NOMBRE		CAMA: [	]	EXPEDIENTE: [		]	
NOMBRE: [	]	[	]	[	]																	
	PATERNO	MATERNO		NOMBRE																		
CAMA: [	]	EXPEDIENTE: [		]																		
OBSERVACIONES:																						

Fig. B.4. Forma de diseño de pantalla para Criterios de Búsqueda por Nombre.



Sistema: SAAH  
Módulo: Hospitalización  
Aplicación: Forma (Pantalla)  
Nombre: faa\_egrpc  
Objetivo: Pantalla de registro del egreso del paciente.

SAAH	EGRESO DEL PACIENTE	HRALM
CAMA: [ ]	EXPEDIENTE: [ ]	
ESPECIALIDAD: [ ] [ ]		
MEDICO: [ ] [ ]		
PACIENTE:	EDAD	SEXO
[ ] [ ] [ ] [ ]	[ ] [ ] [ ] [ ]	[ ] [ ] [ ] [ ]
DATOS DEL EGRESO:		
DX. PRINCIPAL: [ ] [ ]		
DX. SECUNDARIO: [ ] [ ]		
FECHA EGRESO: [ ] [ ]	HORA: [ ]	MIN.: [ ]
MOTIVO: [ ] [ ]		
DESTINO: [ ] [ ]		
ESC ACEPTAR	RETURN	TERMINAR

OBSERVACIONES:

Fig. B.6. Forma de diseño de pantalla para el registro del egreso del paciente.





Sistema: SAAH  
 Módulo: Relaciones Públicas  
 Aplicación: Forma (Pantalla)  
 Nombre: frc\_conpac  
 Objetivo: Pantalla para de selección para la localización del paciente .

---

SAAH LOCALIZACION Y ESTADO DEL PACIENTE HRALM

PACIENTE: [            ] [            ] [            ]

EXPEDIENTE	NOMBRE	S	EDAD	CAMA	FEC. INGRESO
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]
[            ]	[            ]	[            ]	[            ]	[            ]	[            ]

CTRL + P RETROCEDE    ESC ACEPTAR    RETURN TERMINAR

OBSERVACIONES:

Fig. B.8. Forma de diseño de pantalla para la localización del paciente.

Sistema: SAAH  
 Módulo: Relaciones Públicas  
 Aplicación: Forma (Pantalla)  
 Nombre: fha\_edosal  
 Objetivo: Consulta del estado del paciente y ubicación.

SAAH	ESTADO DE SALUD DEL PACIENTE	HRAIM
PACIENTE: [            ] [            ] [            ]		
EDAD: [ ] [ ] [ ] SEXO: [ ] [ ]		
SITUACION: [ ] [ ] [ ]		
MEDICO: [ ] [ ] [ ]		
ESPECIALIDAD: [ ] [ ] [ ]		
CAMA: [ ] [ ]		
MEDICO: [ ] [ ] [ ]		
EDO. DE SALUD: [ ] [ ] [ ]		
FECHA EGRESO: [ ] [ ] [ ]		
CONTROL CONSULTA      ESC ACEPTAR      RETURN TERMINAR		

OBSERVACIONES:

Fig. B.9. Forma de diseño de pantalla para la Consulta del Estado de salud del paciente y su localización.

Sistema: SAAH  
Módulo: Relaciones Públicas  
Aplicación: Forma (Pantalla)  
Nombre: rc\_catequiv1  
Objetivo: Proporcionar ayuda auxiliar para el campo de "Situación".

SAAH	ESTADO DE SALUD DEL PACIENTE	HRALM
PACIENTE: [            ] [            ] [            ]		
EDAD: [   ] [   ]	SEXO: [   ]	
SITUACION: [   ] [   ]		]
MEDICO: [   ] [   ]		]
ESPECIALIDAD: [   ] [   ]		]
CAMA: [   ]		
MEDICO: [   ] [   ]		
EDO. DE SALUD: [   ] [   ]		
FECHA EGRESO: [   ] [   ]		
		<div style="border: 1px solid black; padding: 5px; text-align: center;"> SITUACION  PISO  QUIROFANO  EGRESO </div>
ESC ACEPTAR	RETURN TERMINAR	

OBSERVACIONES:

Fig. B.9b. Forma de diseño de pantalla auxiliar para el campo de "Situación".

Sistema: SAAH  
 Módulo: Admisión Hospitalaria.  
 Aplicación: Listado (Reporte)  
 Nombre: la\_hojaing  
 Objetivo: Generar la hoja de ingreso hospitalario.

ISSSTE  
 SUBDIRECCION GRAL MEDICA  
 HOSPITAL REG  
 LIC ADOLFO LOPEZ MATEOS

SAAH  
 FEC 99/59/99

ADMISION HOSPITALARIA  
 HOJA DE INGRESO HOSPITALARIO

DATOS GENERALES DEL PACIENTE

EXPEDIENTE [ ] T. DE DERECHOS [ ]  
 NOMBRE [ ] [ ] [ ] [ ] EDAD [ ] [ ] SEXO [ ]  
 EDO CIV. [ ] DOMICILIO [ ]  
 COLONIA [ ] CIUDAD [ ] C.P. [ ]  
 ESTADO [ ] TEL. [ ] OCUP. [ ]  
 DEPENDENCIA [ ] CLINICA [ ]

DATOS DE HOSPITALIZACION

FEC INGRESO [ ] HORA [ ] [ ] SERVICIO [ ]  
 CUARTO [ ] CAMA [ ] ESPECIALIDAD [ ]  
 DIAGNOSTICO [ ]  
 [ ]

VIGENCIA DE DERECHOS: SI [ ] NO [ ] FECHA [ ]

DATOS DEL RESPONSABLE DEL PACIENTE

NOMBRE [ ] [ ] [ ] [ ]  
 DOMICILIO [ ] COLONIA [ ] C.P. [ ]  
 CIUDAD [ ] ESTADO [ ]  
 TELEFONO [ ]

NO DEJA ROPA NO OBJETOS DE VALOR [ ]

FIRMA

PROCEDENCIA

SERVICIO [ ]  
 ESPECIALIDAD [ ]  
 MEDICO QUE AUTORIZA [ ]  
 UNIDAD MEDICA [ ]  
 OBSERVACIONES [ ]  
 [ ]

OBSERVACIONES:

Fig. B.10. Reporte de la Hoja de Ingreso Hospitalario.

Sistema: SAAH  
 Módulo: Admisión Hospitalaria.  
 Aplicación: Listado (Reporte)  
 Nombre: lh\_censod  
 Objetivo: Generar la cédula para el censo diario.

ISSSTE  
 SUBDIRECCION GRAL. MEDICA  
 HOSPITAL REG.  
 LIC. ADOLFO LOPEZ MATEOS

SAAH  
 FEC.: 99/99/99

ADMISION HOSPITALARIA  
 CEDULA PARA EL CENSO DIARIO

TURNO RECORRIDO: [ ]  
 SERVICIO: [ ] PISO: [ ] OBSERV.: [ ]  
 RESPONSABLE: [ ] [ ] [ ] [ ]

EXPEDIENTE	NOMBRE	CAMA	SEX	OBSERVACIONES
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]
[ ]	[ ]	[ ]	[ ]	[ ]

OBSERVACIONES:

Fig. B.11. Reporte de la Cédula para el Censo Diario.

**Sistema:** SAAH  
**Módulo:** Admisión Hospitalaria.  
**Aplicación:** Listado (Reporte)  
**Nombre:** lh\_cedcen  
**Objetivo:** Generar el listado para la relación de expedientes del Censo Diario

ISSSTE  
 SUBDIRECCION GRAL. MEDICA  
 HOSPITAL. REG.  
 LIC. ADOLFO LOPEZ MATEOS

SAAH  
 FEC.: 99/99/99

ADMISION HOSPITALARIA  
 CEDULA PARA EL CENSO DIARIO

---

**ATENCION**  
 ARCHIVO CLINICO: ANEXO AL PRESENTE EXPEDIENTES DE (LOS) PACIENTE(S)  
 CUYO(S) DATOS(S) SE RELACIONAN:

EXPEDIENTE	NOMBRE	CAMA	NO. SOL.	FECHA SALIDA
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

**OBSERVACIONES:**

Fig. B.12. Reporte de la relación de expedientes del Censo Diario.

Sistema:	SAAH
Módulo:	Admisión Hospitalaria.
Aplicación:	Listado (Reporte)
Nombre:	la_bolinte
Objetivo:	Generar la Boleta de Internamiento.

ISSSTE	SAAH
SUBDIRECCION GRAL. MEDICA	FEC.: 99/99/99
HOSPITAL. REG.	
LIC. ADOLFO LOPEZ MATEOS	

ADMISION HOSPITALARIA  
BOLETA DE INTERNAMIENTO

---

EXPEDIENTE: [                    ] TIPO DE DERECHOABIENTE: [                    ]

PATERNO                    MATERNO                    NOMBRE(S)

[                    ] [                    ] [                    ]

PASE A LA SECCIÓN DE :

SERVICIO: [                    ]

ESPECIALIDAD: [                    ] PISO: [   ] CUARTO : [   ]

CAMA: [   ]

FECHA DE INGRESO: [                    ] HORA : [   ] : [   ]

MEDICO: [                    ]

RECIBIO: [                    ]

HORA DE RECEPCION: [   ] : [   ]

OBSERVACIONES:

Fig. B.13. Reporte de la Boleta de Internamiento.



FUNCTION cons\_equiv\_cat(opcion, identificador)

```

DEFINE
    opcion          CHAR(1),
    identificador  SMALLINT,
    ind            SMALLINT,
    pos           SMALLINT,
    scr_pos      SMALLINT,
    i            SMALLINT,
    orden_selecc CHAR(200),
    busca_desc  CHAR(200),
    a_valor     ARRAY[2500] OF RECORD
    clave       SMALLINT,
    descripcion CHAR(50)
END RECORD
CALL abre_window(3, 13, 29, 9, 49)
    OPTIONS INPUT    WRAP,
                   FORM    LINE 3,
                   ACCEPT  KEY ESC,
                   MESSAGE LINE LAST
IF (opcion = "E") THEN
    DISPLAY "VALOR"      AT 01,02 ATTRIBUTE(REVERSE, RED)
ELSE
    DISPLAY "CLAVE"     AT 01,02 ATTRIBUTE(REVERSE, RED)
END IF
DISPLAY "DESCRIPCION " AT 01,08 ATTRIBUTE(REVERSE, RED)
IF (opcion = "E") THEN
    LET orden_selecc = "SELECT valor, descripcion FROM dominio_campos",
    " WHERE dominio_campos.clave_equiv = ", identificador, " ORDER BY 1"
ELSE
    CALL mensaje(5, TRUE, TRUE, " Digite Referencia de Búsqueda ...")
    OPEN FORM f_buscat FROM "f_buscat"
    DISPLAY FORM          f_buscat
    CONSTRUCT BY NAME busca_desc ON desc_catalog.clave,
    desc_catalog.descripcion
    CALL mensaje(6, TRUE, FALSE, "Seleccionando Informacion ...")

```

Listado B.14. Rutina Común para el despliegue de Ventanas de ayuda  
(Continua).

```

LET orden_selecc = " SELECT clave, descripcion FROM
desc_catalog",
" WHERE ", busca_desc CLIPPED,
" ORDER BY 1"

END IF
PREPARE cur_auxiliar FROM orden_selecc
DECLARE cur_val SCROLL CURSOR FOR cur_auxiliar
LET ind = 1
LET pos = 0
OPEN cur_val
FETCH FIRST cur_val INTO a_valor[ind].clave, a_valor[ind].descripcion
IF (STATUS = NOTFOUND) THEN
CLOSE cur_val
IF (opcion = "E") THEN
CALL mensaje(6, TRUE, TRUE, "Valores NO Localizados . . .")
ELSE
CALL mensaje(6, TRUE, TRUE, "Informacion NO Localizada . . .")
END IF
CALL cierra_window(3)
RETURN
ELSE
WHILE TRUE
LET ind = ind + 1
FETCH NEXT cur_val INTO a_valor[ind].clave, a_valor[ind].descripcion
IF (STATUS = NOTFOUND) THEN
EXIT WHILE
END IF
END WHILE
CLOSE cur_val
CALL set_count(ind - 1)
IF (ind >= 1) THEN
OPEN FORM f_catequiv FROM "f_catequiv"

```

Listado B.14. Rutina Común para el despliegue de Ventanas de ayuda  
(Continua).

```

DISPLAY FORM      f_catequiv
OPTIONS NEXT KEY CONTROL-N,
ACCEPT KEY       CONTROL-E,
PREVIOUS KEY     CONTROL-P
CALL mensaje(1, FALSE, FALSE, "")

IF (ind > 5) THEN
    DISPLAY "CTRL+N "          AT 09,02 ATTRIBUTE(REVERSE)
    DISPLAY "Avanza"          AT 09,10
    DISPLAY "CTRL+P "          AT 09,17 ATTRIBUTE(REVERSE)
    DISPLAY "Retrocede"       AT 09,25
END IF

DISPLAY "ESC "              AT 09,35 ATTRIBUTE(REVERSE, RED)
DISPLAY "Selecciona"        AT 09,40
DISPLAY ARRAY a_valor TO sc_valores.*
ON KEY(ESC)
    LET pos = ARR_CURR()
    LET scr_pos = SCR_LINE()
    DISPLAY a_valor[pos].* TO sc_valores[scr_pos].*
    ATTRIBUTE(REVERSE, RED)
    EXIT DISPLAY
END DISPLAY

CALL cierra_window(3)
OPTIONS ACCEPT KEY ESC
RETURN a_valor[pos].*

END IF

END IF

END FUNCTION

```

Listado B.14. Rutina Común para el despliegue de Ventanas de ayuda.



DATABASE saah

FUNCTION datos\_hospitalizacion(IConf, sig\_ingreso)

DEFINE

```

IConf      RECORD
           expediente      LIKE ingresos.expediente,
           paciente        CHAR(60),
           sexo             LIKE archivo_clinico.sexo,
           edad             LIKE ingresos.edad,
           clave_edad       LIKE ingresos.clave_edad,
           f_ingreso        LIKE ingresos.f_ingreso
           END RECORD,
rDatHosp   RECORD      LIKE movimientos_pac.*,
           expediente_cama  LIKE movimientos_pac.expediente,
           desc_especialidad CHAR(50),
           hora_inter        CHAR(8),
           b_dx              SMALLINT,
           hay_inter         SMALLINT,
           sig_ingreso       SMALLINT
    
```

```

OPEN WINDOW v_dat_hosp AT 11, 5 WITH 11 ROWS, 72 COLUMNS
ATTRIBUTE(BORDER, MESSAGE LINE LAST, FORM LINE 1)
OPEN FORM faa_dathos FROM "faa_dathos"
DISPLAY FORM faa_dathos
    
```

OPTIONS

```

INPUT WRAP,
INPUT ATTRIBUTE (RED),
DISPLAY ATTRIBUTE (RED)
    
```

Listado B. 16. Módulo para el registro de los datos de hospitalización del paciente  
(Continua).

```

DISPLAY "CTRL+B " AT 10,44 ATTRIBUTE(REVERSE)
DISPLAY "Cancela" AT 10,52
DISPLAY "ESC " AT 10,61 ATTRIBUTE(REVERSE)
DISPLAY "Procesa" AT 10,66

```

```

LET hay_inter      = FALSE
LET hora_inter     = TIME
LET rDatHosp.cama  = NULL
LET rDatHosp.clave_dx = NULL
LET rDatHosp.hora_mov = hora_inter[1,2]
LET hora_inter[4,5] = hora_inter[4,5] USING "&&"
LET rDatHosp.minuto_mov = hora_inter[4,5]

```

```

INPUT rDatHosp.hora_mov, rDatHosp.minuto_mov, rDatHosp.cama,
      rDatHosp.clave_dx, rDatHosp.diagnostico
WITHOUT DEFAULTS FROM hora_mov, minuto_mov, cama, clave_dx,
diagnostico

```

```

ON KEY(CONTROL-F)
  IF INFIELD(clave_dx) THEN
    LET b_dx = TRUE
    CALL cons_equiv_cat("C", "04")
    RETURNING rDatHosp.clave_dx, rDatHosp.diagnostico
    DISPLAY rDatHosp.clave_dx, rDatHosp.diagnostico
    TO clave_dx, diagnostico
  END IF

```

```

AFTER FIELD hora_mov
  IF (rDatHosp.hora_mov IS NULL OR rDatHosp.hora_mov <= 0 OR
      rDatHosp.hora_mov > 24) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique la Hora del Ingreso ...")
  NEXT FIELD hora_mov
END IF

```

Listado B.16. Módulo para el registro de los datos de hospitalización del paciente  
(Continúa).

```

AFTER FIELD minuto_mov
  IF (rDatHosp.minuto_mov IS NULL OR rDatHosp.minuto_mov <= 0 OR
      rDatHosp.minuto_mov > 59) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique el Minuto del Ingreso . . .")
  NEXT FIELD minuto_mov
END IF

```

```

AFTER FIELD cama
  IF (rDatHosp.cama IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique el Numero de Cama . . .")
  NEXT FIELD cama
END IF
SELECT expediente, especialidad
INTO expediente_cama, rDatHosp.especialidad
FROM camas
WHERE camas.cama = rDatHosp.cama

```

```

IF (STATUS = NOTFOUND) THEN
  CALL mensaje (6, TRUE, TRUE, "Numero de Cama NO Existe . . .")
NEXT FIELD cama
ELSE
  IF (expediente_cama IS NOT NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Cama Ocupada . . .")
  NEXT FIELD cama
END IF
END IF

```

```

CALL desc_equiv_cat("C", 01, rDatHosp.especialidad)
RETURNING rDatHosp.especialidad, desc_especialidad
DISPLAY desc_especialidad TO desc_especialidad

```

```

BEFORE FIELD clave_dx
  LET b_dx = FALSE
  CALL tecla_ayuda("D", 10)

```

Listado B.16. Módulo para el registro de los datos de hospitalización del paciente  
(Continua).

```
AFTER FIELD clave_dx
  CALL tecla_ayuda("Q", 10)
  LET b_dx = FALSE
  IF NOT b_dx THEN
    CALL desc_equiv_cat("C", 04, rDatHosp.clave_dx)
    RETURNING rDatHosp.clave_dx, rDatHosp.diagnostico
    IF (rDatHosp.diagnostico IS NULL) THEN
      CALL mensaje (6, TRUE, TRUE,
        "Verifique la Clave del Diagnostico . . .")
    NEXT FIELD clave_dx
  END IF
END IF
DISPLAY rDatHosp.diagnostico TO diagnostico
IF (rDatHosp.clave_dx IS NOT NULL) THEN
  NEXT FIELD hora_mov
END IF

AFTER FIELD diagnostico
IF (rDatHosp.clave_dx IS NULL AND rDatHosp.diagnostico IS NULL) THEN
  CALL mensaje (6, TRUE, TRUE, "Verifique el Diagnostico Por Favor . . .")
  NEXT FIELD clave_dx
END IF

ON KEY(INTERRUPT)
  LET int_flag = FALSE
  LET hay_inter = TRUE
  EXIT INPUT

ON KEY(CONTROL-B)
  LET hay_inter = TRUE
  EXIT INPUT

END INPUT

IF NOT hay_inter THEN
```

Listado B.16. Módulo para el registro de los datos de hospitalización del paciente  
(Continúa).

```

IF sorn (" SE CONFIRMA EL INGRESO DEL PACIENTE ?", "", 16, 35, 1, 1, 1)
THEN
    LET rDatHosp.expediente = IConf.expediente
    LET rDatHosp.fecha_mov = IConf.f_ingreso
    LET rDatHosp.num_ingreso = sig_ingreso
    LET rDatHosp.cons_movpac = 1

    INSERT INTO movimientos_pac VALUES (rDatHosp.*)

    UPDATE camas SET expediente = rDatHosp.expediente
    WHERE camas.cama = rDatHosp.cama

    CALL mensaje (5, TRUE, TRUE, "Ingreso Confirmado . . .")
    IF sorn (" DESEA REGISTRAR LOS DATOS DEL RESPONSABLE ?",
            "", 16, 29, 1, 1, 1) THEN
        CALL responsable_paciente(IConf.expediente)
    END IF
END IF
ELSE
    CALL mensaje (6, TRUE, TRUE, "Confirmacion de Ingreso Cancelada . . .")
END IF
CLOSE WINDOW v_dat_hosp
RETURN hay_inter
END FUNCTION
    
```

Listado B.16. Módulo para el registro de los datos de hospitalización del paciente

A continuación se muestra en el listado B.17. la forma para el registro de los datos de hospitalización al ingreso del paciente.

```

DATABASE saah
SCREEN SIZE 22 by 80
(
HORA DE INGRESO : [f1];[f2]

CAMA           : [f003]
ESPECIALIDAD  : [f004]           ]

DIAGNOSTICO   : [f005 ]
               [f006           ]

)
END

TABLES
  movimientos_pac

ATTRIBUTES

f1   = movimientos_pac.hora_mov;
f2   = movimientos_pac.minuto_mov;
f003 = movimientos_pac.cama;
f004 = formonly.desc_especialidad TYPE CHAR, NOENTRY;
f005 = movimientos_pac.clave_dx;
f006 = movimientos_pac.diagnostico, UPSHIFT;
END

```

Listado B.17. Listado de la Forma para el registro de los datos de hospitalización al ingreso del paciente.

En el siguiente listado (B.18.), se presenta el programa del módulo para el registro del responsable del paciente. Al final del programa, se presenta el listado (B.19.) de la forma de registro del responsable del paciente.



```

LET hay_inter = FALSE
INPUT BY NAME rResp.responsable THRU rResp.parentesco

AFTER FIELD responsable
  IF (rResp.responsable IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique el Responsable . . .")
  NEXT FIELD responsable
END IF

BEFORE FIELD calle

IF sorn (" MISMA DIRECCION DEL PACIENTE ?", "", 16, 42, 1. 1. 1) THEN

  # Selecciono la direccion del paciente
  SELECT calle, colonia, codigo_postal, telefono
  INTO rResp.calle, rResp.colonia, rResp.codigo_postal, rResp.telefono
  FROM archivo_clinico
  WHERE archivo_clinico.expediente = Num_Expediente

  DISPLAY rResp.calle, rResp.colonia, rResp.codigo_postal, rResp.telefono
  TO calle, colonia, codigo_postal, telefono
  NEXT FIELD parentesco
END IF

AFTER FIELD calle
  IF (rResp.calle IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique la Calle . . .")
  NEXT FIELD calle
END IF

AFTER FIELD colonia
  IF (rResp.colonia IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique la Colonia . . .")
  NEXT FIELD colonia
END IF

```

Listado B.18. Módulo para el registro de los datos del paciente (Continua).

```
AFTER FIELD codigo_postal
  IF (rResp.codigo_postal IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique elCodigo Postal . . .")
  NEXT FIELD codigo_postal
END IF

AFTER FIELD parentesco
  IF (rResp.parentesco IS NULL) THEN
    CALL mensaje (6, TRUE, TRUE, "Verifique elParentesco . . .")
  NEXT FIELD parentesco
END IF

ON KEY(INTERRUPT)
  LET int_flag = FALSE
  LET hay_inter = TRUE
  EXIT INPUT

ON KEY(CONTROL-B)
  LET hay_inter = TRUE
  EXIT INPUT

END INPUT

IF NOT hay_inter THEN
  LET rResp.expediente = Num_Expediente
  INSERT INTO responsable_pac VALUES (rResp.*)
  CALL mensaje (5, TRUE, TRUE, "Registro de Responsable Confirmado . . .")
ELSE
  CALL mensaje (6, TRUE, TRUE, "Registro de Responsable Cancelado . . .")
END IF
CLOSE WINDOW v_resp_pac
END FUNCTION
```

Listado B.18. Módulo para el registro de los datos del paciente.

```
DATABASE saah
```

```
SCREEN SIZE 22 by 80
```

```
{
```

```
RESPONSABLE  {{f001          } }
```

```
CALLE        {{f002          } }
```

```
COLONIA      {{f003          } }
```

```
CODIGO POSTAL {{f004 } }
```

```
TELEFONO     {{f005    } }
```

```
PARENTESCO   {{f006          } }
```

```
}
```

```
END
```

```
TABLES
```

```
  responsable_pac
```

```
ATTRIBUTES
```

```
f001 = responsable_pac.responsable, UPSHIFT;
```

```
f002 = responsable_pac.calle, UPSHIFT;
```

```
f003 = responsable_pac.colonia, UPSHIFT;
```

```
f004 = responsable_pac.codigo_postal;
```

```
f005 = responsable_pac.telefono, UPSHIFT;
```

```
f006 = responsable_pac.parentesco, UPSHIFT;
```

```
END
```

Listado B.19. Listado de la Forma para el registro de los datos del paciente.

## APÉNDICE C

### GLOSARIO TÉCNICO

**Agregado de datos.** Una determinada colección de ítem de datos (elementos de datos) dentro de un registro. *Ver también* grupo.

**Almacenamiento de datos.** Cualquier lugar de un sistema donde se almacenan los datos entre transacciones o entre ejecuciones del sistema ( incluye archivos manuales y legibles por máquina, bases de datos y tablas).

**Argumento.** Un valor que se emplea como entrada a algún proceso, a menudo trasferido a través de una interfaz módulo - módulo.

**Argumento de búsqueda.** El valor del atributo empleado(s) para recuperar algún dato de un almacenamiento de datos, ya sea a través de un Índice o mediante una búsqueda. *Ver también* argumento.

**Atributo.** Un elemento de datos que contiene información sobre una entidad.

**Base de datos.** Una colección de datos interrelacionados almacenados juntos con redundancia controlada para servir a una o más aplicaciones; los datos están almacenados de manera que sean independientes de los programas que los usan, se emplea un procedimiento común y controlado para agregar nuevos datos y para modificar y recuperar los existentes dentro de una base de datos.

**Bases de Datos Relacional.** Una base de datos construida únicamente con relaciones normalizadas. El modelo relacional de una Base de Datos se puede definir como una vista de la información del usuario final, presentada a través de tablas o relaciones normalizadas , es decir, la información se organiza en archivos de cómputo en una forma muy semejante a como se trabaja en forma manual ( expedientes en un archivo).

Cada tabla refleja o equivale a un archivo secuencial tradicional, en donde las columnas corresponden a datos de un registro y las filas a los registros propiamente dichos (tuplas).

**Clave.** Un elemento de datos (o grupo de elementos de datos) empleado para encontrar o identificar un registro ("tupla").

**Clave candidata.** Un atributo o grupo de atributos cuyos valores identifican unívocamente cada tupla de una relación y para la cual no puede quitarse ninguno de los atributos sin destruir la identificación unívoca.

**Clave primaria.** Clave que identifica unívocamente un registro ("tupla").

**Desarrollo de arriba hacia abajo.** Una estrategia de desarrollo mediante la cual los módulos ejecutivos de control de un sistema se codifican y prueban primero para integrar una versión "esqueleto" del sistema, y después que se ha probado el funcionamiento de las interfaces del sistema, se codifican y prueban los módulos de los niveles inferiores.

**Diagrama de acceso inmediato de datos (DIAD).** Una representación de los caminos de acceso inmediato a un almacenamiento de datos que muestra lo que los usuarios quieren recuperar del almacenamiento de datos sin llevar a cabo la búsqueda o la clasificación.

**Diagrama de flujo de datos (DFD).** Una representación de los flujos de datos a través de un sistema de cualquier tipo, que muestra las entidades externas que son fuente o destino de los datos, los procesos que transforman los datos, y los lugares donde los datos son almacenados.

**Diccionario de datos.** Un almacenamiento de datos que describe la naturaleza de cada parte de los datos empleados en el sistema, que a menudo incluye las descripciones de procesos, entradas del glosario y otros artículos.

**Diseño.** El proceso (iterativo) de tomar un modelo lógico de un sistema con un conjunto de objetivos fuertemente establecidos para este sistema, y producir la especificación de un sistema físico que satisfaga dichos objetivos.

**Dominio.** El conjunto de todos los valores de un elemento de datos que forma parte de una relación. Es un equivalente efectivo de un campo o de un elemento de datos.

**Elemento de datos (ítem de datos, campo).** La unidad de datos más pequeña que tiene significado para el propósito que se trata.

**Elemento de datos continuo.** Aquel que puede tomar tantos valores dentro de su rango que no resulta práctico enumerarlos, por ejemplo, una suma de dinero.

**Elemento de datos discreto.** Aquel que toma sólo un número limitado de valores, cada uno de los cuales tiene generalmente un significado. Ver también elemento de datos continuo.

**En línea.** Conectado directamente a la computadora de manera que la entrada, la salida, los accesos de datos y los cálculos pueden tener lugar sin ninguna intervención humana.

**Entidad.** 1. Entidad externa: fuente o destino de datos en un diagrama de flujo de datos. 2. Algo sobre lo cual se almacena información en un almacenamiento de datos, por ejemplo, clientes, empleados, etc.

**Estructura de datos.** Uno o más elementos de datos con una forma de relación particular, que generalmente se utilizan para describir alguna entidad.

**Grado (de relación normalizada).** El número de dominios que integran la relación (Si existieran siete dominios, la relación es de grado 7).

**Índice.** Un almacenamiento de datos, que como parte del proceso de recuperación toma información sobre los valores de algún atributo(s) y retorna con información que permite que el / los registro(s) con esos atributos sea(n) recuperado(s) rápidamente.

**Índice secundario.** Un índice de un almacenamiento de datos basado en algún atributo distinto de la clave primaria.

**Proceso (transformación).** Un conjunto de operaciones de transformación de datos, lógicas o físicamente, de acuerdo con alguna lógica de proceso.

**Programación estructurada (codificación).** La construcción de programas empleando un pequeño número de construcciones lógicas, cada una con una entrada, una salida, en una jerarquía enlazada.

**Pseudocódigo.** Una herramienta para la especificación de un programa lógico en un formato legible, en forma similar al lenguaje corriente, pero sin observar las reglas sintácticas de ningún lenguaje de programación en particular.

**Relación.** Un archivo representado en formato normalizado como una tabla bidimensional de elementos de datos.

## GLOSARIO MÉDICO

**Control de Camas.** Comprende el conocimiento de camas vacías y camas ocupadas, asignación de camas a los pacientes, transferencias y permuta de los mismos; así como estadísticas del porcentaje de ocupación.

**Informes de Estado de Salud.** Es el resultado de la opinión médica en relación a la evolución de la salud de los pacientes que se atienden en el hospital. Dichos informes son generados en los tres turnos de servicio (matutino, vespertino y nocturno) y se clasifican como pacientes en mejoría, delicados, graves y defunciones.

**Morbilidad:** Es la generación de información que nos permite conocer los padecimientos de los pacientes egresados vivos y que fueron atendidos en nuestros hospitales. Los datos más importantes que se requieren son: códigos de diagnósticos médicos, sexo, edad y tipo de derechohabiente.

**Mortalidad:** Es la generación de información que nos permite conocer las causas de defunción, por las cuales los pacientes son egresados de nuestros hospitales. Los datos más importantes que se requieren son: código de la causa de defunción, sexo, edad y tipo de derechohabiente.

**Ingresos y Egresos Hospitalarios.** Es el registro de la información que identifica a una persona como paciente ingresado a un hospital, así como los datos clínicos (especialidad, diagnósticos médicos, etc.), generados en el periodo de internamiento, además de almacenar el resultado de los diagnósticos principales, con los cuales se egresa a los pacientes.