



UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO

FACULTAD DE INGENIERÍA

---

---

**T E S I S**

“MANUAL PARA MANTENIMIENTO Y  
SEGUIMIENTO DE HISTORIAS CLÍNICAS PARA  
EL DEPARTAMENTO DE SALUD MENTAL DEL  
HOSPITAL GENERAL DE MÉXICO”

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

PRESENTA:

LEONARDO MARTÍNEZ GONZÁLEZ

DIRECTORA DE TESIS:  
DRA. ANA MARÍA VÁZQUEZ VARGAS

ASESOR EXTERNO:  
DR. ALFONSO F. MARTÍN DEL CAMPO LAURENTS





Universidad Nacional  
Autónoma de México



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

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

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

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

*A mi madre, por todo su apoyo, consejos y cariño en todos estos años.*

*A mi padre, por su motivación y gula para seguir adelante.*

*A mi hermana Gaby, por su ayuda y consejos y a mi sobrino que viene en camino.*

*A mi hermana Irene, por su ayuda, cariño y paciencia.*

*A mis Abuelitas Irene y Antonia que desde el cielo observan todo lo q hago.*

*A la Dra. Ana María Vázquez, por ser mi asesora de tesis, gracias por toda su ayuda, su paciencia y también por darme el honor de ser su ayudante.*

*Al Dr. Martín del Campo, por ser mi asesor externo, por su disponibilidad para ayudarme y por su paciencia, gracias Doc.*

*A los médicos y personal de apoyo del Departamento de Salud Mental del H. G. que de alguna manera contribuyeron al desarrollo de mi tesis.*

*A mis amigos, Javier, Abel, Abundio y Jaime por su paciencia y disponibilidad para responderme todas mis dudas.*

*A mi amiga Cinthia, siempre estuviste ahí para decirme que no me rindiera y siguiera adelante, te quiero mucho.*

*A mi amigas Rosa, Maricarmen, Rosangel, Yoali, Tania, Rubi, Viri, gracias niñas, mis amigos Enrique, Luis Ángel, Carlos, Beto, J. Luis, Gustavo, Roberto, Café.*

*A Fernanda, gracias por todo niña.*

*A mi amiga Sofi por todo su apoyo.*

*A la Facultad de Ingeniería y a la UNAM por darme el honor de ser su alumno y pertenecer a una de las mejores Universidades del Mundo*

*Gracias!!*

## INDICE

|   |          |
|---|----------|
| <b>Introducción.....</b>                                    | <b>1</b> |
| <b>I. Marco Teórico de las bases de datos.....</b>          | <b>2</b> |
| 1.1 Conceptos de Bases de Datos.....                        | 2        |
| 1.2 El Modelo de Datos Entidad-Relación (E/R).....          | 2        |
| 1.2.1 Entidades y Relaciones.....                           | 2        |
| 1.2.2 Representación gráfica de Entidades y Relaciones..... | 3        |
| 1.3 Dependencias Funcionales.....                           | 4        |
| 1.4 Integridad Referencial.....                             | 6        |
| 1.5 Formas Normales.....                                    | 6        |
| 1.5.1 Primera forma normal.....                             | 6        |
| 1.5.2 Segunda forma normal.....                             | 7        |
| 1.5.3 Tercera forma normal.....                             | 8        |
| 1.5.4 Forma Normal de Boyce-Codd.....                       | 10       |
| 1.5.5 Cuarta forma normal.....                              | 11       |
| 1.5.6 Quinta forma normal.....                              | 13       |
| 1.6 Herramientas Utilizadas.....                            | 14       |
| 1.6.1 Lenguaje HTML.....                                    | 14       |
| 1.6.2 Lenguaje De Programación PHP.....                     | 15       |
| 1.6.3 Servidor Apache.....                                  | 16       |
| 1.6.4 Lenguaje de Programación JavaScript.....              | 16       |
| 1.6.5 Manejador de Bases de Datos. POSTGRESQL.....          | 17       |

|   |            |
|---|------------|
| <b>II. Entorno del Problema y Requerimientos del Sistema.....</b>           | <b>20</b>  |
| 2.1 Entorno del Problema.....   | 20         |
| 2.2 Requerimientos del Sistema.....   | 21         |
| 2.2.1 Requerimientos de Software.....                                       | 21         |
| 2.2.2 Requerimientos de Hardware.....                                       | 23         |
| <br>  |            |
| <b>III. Desarrollo del Seguimiento del Paciente y la Interconsulta.....</b> | <b>24</b>  |
| 3.1 Migración de datos de MySQL a PostgreSQL. Análisis del sistema.....     | 24         |
| 3.2 Modelo Entidad Relación.....  | 26         |
| 3.3 Diccionario De Datos.....   | 28         |
| 3.4 Elaboración de la Interfaz Grafica.....                                 | 48         |
| 3.5 Seguimiento de pacientes.....   | 54         |
| 3.6 Interconsulta.....  | 61         |
| 3.7 Borrado de Expediente de un paciente.....                               | 63         |
| 3.8 Modulo de Médicos.....  | 64         |
| <br>  |            |
| <b>Conclusiones.....</b>  | <b>66</b>  |
| <b>Glosario.....</b>  | <b>68</b>  |
| <b>Bibliografía.....</b>  | <b>70</b>  |
| <b>Apéndice A Manual de Usuario.....</b>                                    | <b>71</b>  |
| <b>Apéndice B. Funciones en Lenguaje PHP Utilizadas en el Sistema.....</b>  | <b>108</b> |

## Introducción

El Departamento de Salud Mental es una Sección del Hospital General de México en el que diariamente se atienden pacientes de diversos padecimientos, todos los datos que se reciben ahí son almacenados desde hace mucho tiempo en carpetas, por lo que surgió la necesidad de tener un sistema de información para poder manejar de una manera rápida y sencilla todos los datos de los pacientes.

En dicha sección labora el Dr. Alfonso Martín del Campo quien me proporcionó todos los requerimientos del sistema y también iba respondiendo todas las dudas que surgían en la elaboración del mismo.

En principio se elaboró un sistema a cargo del compañero Raúl González<sup>1</sup> quien también trabajó con el Dr. Martín del campo, dicho sistema funcionó muy bien para las altas, bajas y estadísticas de los pacientes, sin embargo tuvo que ser modificado para que se pudiera realizar el seguimiento y las interconsultas de pacientes.

El presente trabajo de tesis se basó en el sistema de Raúl y a partir de él se realizaron los cambios necesarios para mejorar el sistema y así poder realizar el seguimiento de pacientes y la interconsulta.

Se presenta al final el Manual de Usuario, describiendo paso a paso como se utiliza el Sistema de Administración de Historias Clínicas.

Finalmente se ofrece también un conjunto de funciones en PHP y SQL que fueron utilizadas en el sistema, con el fin de dar material de apoyo a la docencia y a la elaboración de Bases de Datos.

---

<sup>1</sup> González A. Raúl. Tesis de Licenciatura de la Carrera Ingeniero en Computación defendida en marzo de 2007.

## Capítulo I. Marco Teórico de las Bases de Datos.

### 1.1 Conceptos de Bases de Datos

Una Bases de Datos es un conjunto de información almacenada en memoria auxiliar que permite acceso directo y un conjunto de programas que manipulan esos datos.

Una Base de Datos es un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo.

### 1.2 El Modelo de Datos Entidad-Relación (E/R)

Cuando se utiliza una base de datos para gestionar información, se está plasmando una parte del mundo real en una serie de tablas, registros y campos ubicados en un ordenador; creándose un modelo parcial de la realidad. Antes de crear físicamente estas tablas en la computadora se debe realizar un modelo de datos.

Se suele cometer el error de ir creando nuevas tablas a medida que se van necesitando, haciendo así el modelo de datos y la construcción física de las tablas simultáneamente. El resultado de esto acaba siendo un sistema de información confuso, con datos dispersos que terminan por no cumplir adecuadamente los requisitos necesarios.

#### 1.2.1 Entidades y Relaciones

El modelo de datos más extendido es el denominado ENTIDAD/RELACIÓN (E/R) En el modelo E/R se parte de una situación real a partir de la cual se definen **entidades** y **relaciones** entre dichas entidades:

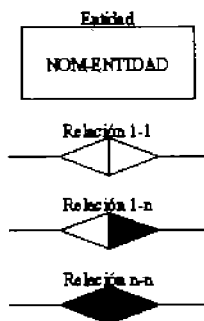
- **Entidad.**- Objeto del mundo real sobre el que queremos almacenar información (Ej: una persona). Las entidades están compuestas de *atributos* que son los datos que definen el objeto (para la entidad persona serían DNI, nombre, apellidos, dirección,...). De entre los atributos habrá uno o un conjunto de ellos que no se repite; a este atributo o conjunto de atributos se le llama **clave** de la entidad, (para la entidad persona una clave sería DNI). En toda entidad siempre hay al menos una clave que en el peor de los casos estará formada por todos los atributos de la tabla. Ya que puede

haber varias claves y necesitamos elegir una, lo haremos atendiendo a estas normas:

- **Que sea única.**
- **Que se tenga pleno conocimiento de ella.-** ¿Por qué en las empresas se asigna a cada cliente un número de cliente?
- **Que sea mínima,** ya que será muy utilizada por el gestor de base de datos.
- **Relación.-** Asociación entre entidades, sin existencia propia en el mundo real que estamos modelando, pero necesaria para reflejar las interacciones existentes entre entidades. Las relaciones pueden ser de tres tipos:
  - **Relaciones 1-1.-** Las entidades que intervienen en la relación se asocian una a una (Ej: la entidad HOMBRE, la entidad MUJER y entre ellos la relación MATRIMONIO).
  - **Relaciones 1-n.-** Una ocurrencia de una entidad está asociada con muchas (n) de otra (Ej: la entidad EMPERSA, la entidad TRABAJADOR y entre ellos la relación TRABAJAR-EN).
  - **Relaciones n-n.-** Cada ocurrencia, en cualquiera de las dos entidades de la relación, puede estar asociada con muchas (n) de la otra y viceversa (Ej: la entidad ALUMNO, la entidad EMPRESA y entre ellos la relación MATRÍCULA).

## 1.2.2 Representación gráfica de Entidades y Relaciones

Para asimilar fácilmente un diseño de datos cuando se emplea el modelo E/R se utilizan los siguientes elementos gráficos:





La utilización de estos elementos dará como resultado lo que se denomina el esquema entidad-relación de la base de datos. Los ejemplos que se incluyen en el apartado anterior, gráficamente quedarían como sigue:



Para pasar del diagrama E/R a la creación de tablas en cada entidad del esquema se creará una tabla con tantos campos como atributos tenga la entidad.

### 1.3 Dependencias Funcionales

Una dependencia funcional sobre una relación R es una proposición de la forma "Si dos tuplas de R concuerdan en los atributos  $A_1, A_2, \dots, A_n$  ( es decir las tuplas poseen los mismos valores en sus componentes respectivas para cada uno de esos atributos), también lo harán en otro atributo B." Esta dependencia se escribe formalmente como  $A_1, A_2, \dots, A_n \rightarrow B$  y se dice que  $A_1, A_2, \dots, A_n$  determinan funcionalmente a B".

| Title         | Year | Length | filmType | StudioName | starName       |
|---------------|------|--------|----------|------------|----------------|
| StarWars      | 1977 | 124    | Color    | Fox        | Carrie Fisher  |
| StarWars      | 1977 | 124    | Color    | Fox        | Mark Hamill    |
| StarWars      | 1977 | 124    | Color    | Fox        | Harrison Ford  |
| Mighty Ducks  | 1991 | 104    | Color    | Disney     | Emilio Estevez |
| Wayne's World | 1992 | 95     | Color    | Paramount  | Dana Carvey    |
| Wayne's World | 1992 | 95     | Color    | Paramount  | Mike Myers     |

Figura 1.La relación Movie<sup>2</sup>

Por ejemplo si se tiene la relación Movie de la Figura 1, existen varias dependencias funcionales:

title year  $\rightarrow$  length

<sup>2</sup> Ullman, Jeffrey D. Introducción a los Sistemas de bases de Datos. México, 1999. Prentice Hall.

title year → filmType

title year → studioName

Dado que las tres dependencias tienen el mismo lado izquierdo, title y year, se puede resumir en una línea mediante la expresión abreviada:

title year → length filmType studioName

De un modo informal este conjunto de dependencias funcionales indica que si dos tuplas poseen el mismo valor en sus componentes title y si también posee el mismo valor en sus componentes year, entonces tendrán los mismos valores en sus componentes length, en sus componentes filmtype y en sus componentes studioName. Esta es una afirmación lógica. Los atributos title y year forman una llave de los objetos película, por lo que con un determinado título y año la película tendrá una duración, un tipo de película y un estudio propietario únicos.

Por otra parte, se tiene la proposición:

title year → starName

Es falsa; no es una dependencia funcional. Se podría pensar que la dependencia es correcta pues title y year forman una llave de películas, pero por la manera en que se definió la clase Movie, sólo es verdad que en cada película existe un conjunto de estrellas determinado de manera única.

Así aun cuando todas las tuplas coinciden en las otras propiedades de la clase Movie, esto no sucede en el nombre de la estrella.

### 1.4 Integridad Referencial

A diferencia de las restricciones del valor único que establecen que a lo sumo existe un valor en determinado papel, la restricción de integridad referencial afirma que hay exactamente un valor en ese papel.

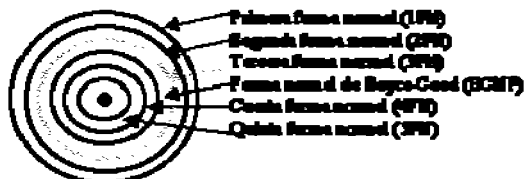
La restricción indica que debe existir un objeto al cual se le está haciendo referencia., hay varias formas para lograr esto:

1. Se puede prohibir que se suprima un objeto referenciado.
2. Se puede exigir que si se suprimiera un objeto referenciado, todos los objetos que hacen referencia a el también sean eliminados.

### 1.5 Formas Normales.

Son las técnicas para prevenir las anomalías en las tablas. Dependiendo de su estructura, una tabla puede estar en primera forma normal, segunda forma normal o en cualquier otra.

Relación entre las formas normales:



#### 1.5.1 Primera forma normal.

##### **Definición formal:**

Una relación R se encuentra en 1FN si y solo si por cada renglón columna contiene valores atómicos.

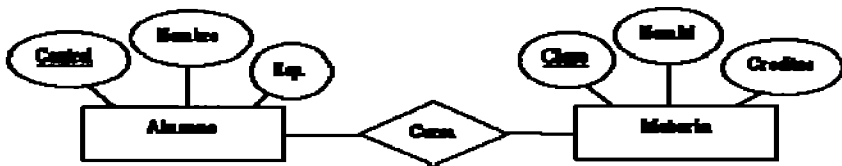
Abreviada como 1FN, se considera que una relación se encuentra en la primera forma normal cuando cumple lo siguiente:

1. Las celdas de las tablas poseen valores simples y no se permiten grupos ni arreglos repetidos como valores, es decir, contienen un solo valor por cada celda.
2. Todos los ingresos en cualquier columna (atributo) deben ser del mismo tipo.

3. Cada columna debe tener un nombre único, el orden de las columnas en la tabla no es importante.
4. Dos filas o renglones de una misma tabla no deben ser idénticas, aunque el orden de las filas no es importante.

Por lo general la mayoría de las relaciones cumplen con estas características, así que podemos decir que la mayoría de las relaciones se encuentran en la primera forma normal.

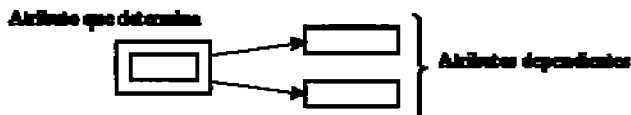
Para ejemplificar como se representan gráficamente las relaciones en primera forma normal consideremos la relación alumno cursa materia cuyo diagrama E-R es el siguiente:



Como esta relación maneja valores atómicos, es decir un solo valor por cada uno de los campos que conforman a los atributos de las entidades, ya se encuentra en primera forma normal, gráficamente así representamos a las relaciones en 1FN.

### 1.5.2 Segunda forma normal.

Para definir formalmente la segunda forma normal requerimos saber que es una **dependencia funcional**: Consiste en edificar que atributos dependen de otro(s) atributo(s).

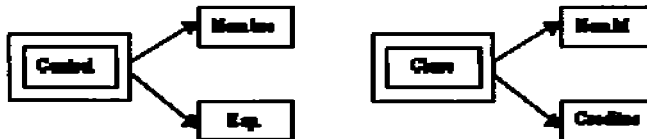


#### Definición formal:

Una relación R está en 2FN si y solo si está en 1FN y los atributos no primos dependen funcionalmente de la llave primaria.

Una relación se encuentra en segunda forma normal, cuando cumple con las reglas de la primera forma normal y todos sus atributos que no son claves (llaves) dependen por completo de la clave. De acuerdo con esta definición, cada tabla que tiene un atributo único como clave, esta en segunda forma normal.

La segunda forma normal se representa por dependencias funcionales como:



Nótese que las llaves primarias están representadas con doble cuadro, las flechas nos indican que de estos atributos se puede referenciar a los otros atributos que dependen funcionalmente de la llave primaria.

### 1.5.3 Tercera forma normal

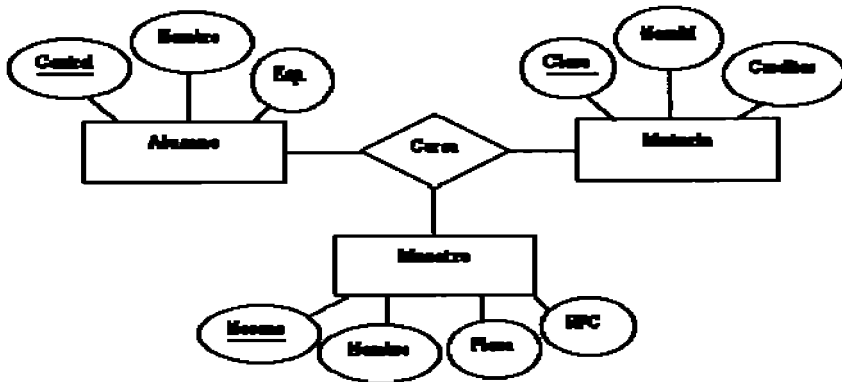
Para definir formalmente la 3FN necesitamos definir **dependencia transitiva**: En una afinidad (tabla bidimensional) que tiene por lo menos 3 atributos (A,B,C) en donde A determina a B, B determina a C pero no determina a A.

#### Definición formal:

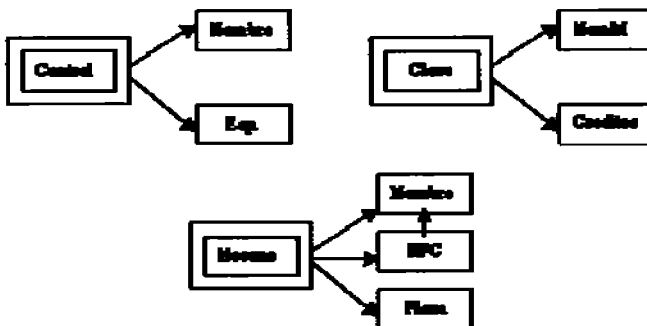
Una relación R está en 3FN si y solo si esta en 2FN y todos sus atributos no primos dependen no transitivamente de la llave primaria.

Consiste en eliminar la dependencia transitiva que queda en una segunda forma normal, en pocas palabras una relación esta en tercera forma normal si está en segunda forma normal y no existen dependencias transitivas entre los atributos, nos referimos a dependencias transitivas cuando existe más de una forma de llegar a referencias a un atributo de una relación.

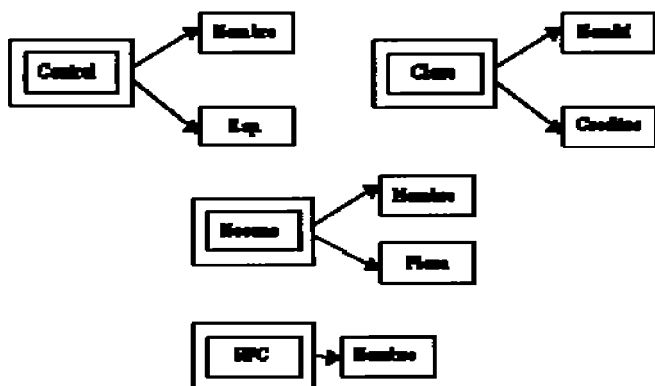
Por ejemplo, consideremos el siguiente caso:



Tenemos la relación alumno-curso-materia manejada anteriormente, pero ahora consideramos al elemento maestro, gráficamente lo podemos representar de la siguiente manera:



Podemos darnos cuenta que se encuentra graficado en segunda forma normal, es decir que todos los atributos llave están indicados en doble cuadro indicando los atributos que dependen de dichas llaves, sin embargo en la llave Necono tiene como dependientes a 3 atributos en el cual el nombre puede ser referenciado por dos atributos: Necono y RFC (Existe dependencia transitiva), podemos solucionar esto aplicando la tercera forma normal que consiste en eliminar estas dependencias separando los atributos, entonces tenemos:



#### 1.5.4 Forma Normal de Boyce-Codd.

**Determinante:** Uno o más atributos que, de manera funcional, determinan otro atributo o atributos. En la dependencia funcional  $(A,B) \rightarrow C$ ,  $(A,B)$  son los determinantes.

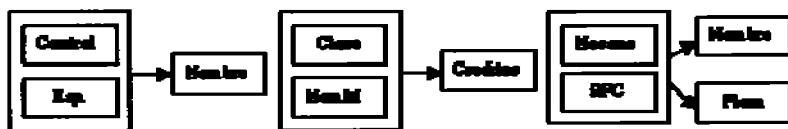
**Definición formal:**

Una relación R esta en FNBC si y solo si cada determinante es una llave candidato.

Denominada por sus siglas en ingles como BCNF; Una tabla se considera en esta forma si y sólo si cada determinante o atributo es una llave candidato.

Continuando con el ejemplo anterior, si consideramos que en la entidad alumno sus atributos control y nombre nos puede hacer referencia al atributo esp., entonces decimos que dichos atributos pueden ser llaves candidato.

Gráficamente podemos representar la forma normal de Boyce Codd de la siguiente forma:



Obsérvese que a diferencia de la tercera forma normal, agrupamos todas las llaves candidato para formar una global (representadas en el recuadro) las cuales hacen referencia a los atributos que no son llaves candidato.

### 1.5.5 Cuarta forma normal.

#### Definición formal:

Un esquema de relaciones R está en 4FN con respecto a un conjunto D de dependencias funcionales y de valores múltiples sí, para todas las dependencias de valores múltiples en D de la forma  $X \twoheadrightarrow Y$ , donde  $X \leq R$  y  $Y \leq R$ , se cumple por lo menos una de estas condiciones:

- \*  $X \twoheadrightarrow Y$  es una dependencia de valores múltiples trivial.
- \* X es una superllave del esquema R.

Para entender mejor aún esto consideremos una afinidad (tabla) llamada estudiante que contiene los siguientes atributos: Clave, Especialidad, Curso tal y como se demuestra en la siguiente figura:

| Clave | Especialidad | Curso    |
|-------|--------------|----------|
| S01   | Sistemas     | Natación |
| S01   | Bioquímica   | Danza    |
| S01   | Sistemas     | Natación |
| B01   | Bioquímica   | Guitarra |
| C03   | Civil        | Natación |

Suponemos que los estudiantes pueden inscribirse en varias especialidades y en diversos cursos. El estudiante con clave S01 tiene su especialidad en sistemas y Bioquímica y toma los cursos de Natación y danza, el estudiante B01 tiene la especialidad en Bioquímica y toma el curso de Guitarra, el estudiante con clave C03 tiene la especialidad de Civil y toma el curso de natación.

En esta tabla o relación no existe dependencia funcional porque los estudiantes pueden tener distintas especialidades, un valor único de clave puede poseer muchos valores de especialidades al igual que de valores de cursos. Por lo tanto existe **dependencia de valores múltiples**. Este tipo de dependencias produce redundancia de datos, como se puede apreciar en la tabla anterior, en donde la clave S01 tiene tres registros para mantener la serie de datos en forma independiente lo cual ocasiona que al realizarse una actualización se requiera de demasiadas operaciones para tal fin.

Existe una dependencia de valores múltiples cuando una afinidad tiene por lo menos tres atributos, dos de los cuales poseen valores múltiples y sus valores dependen solo del tercer atributo, en otras palabras en la afinidad R (A,B,C) existe una dependencia de valores múltiples si A determina valores múltiples de B, A determina valores múltiples de C, y B y C son independientes entre sí.



En la tabla anterior Clave determina valores múltiples de especialidad y clave determina valores múltiples de curso, pero especialidad y curso son independientes entre sí.

Las dependencias de valores múltiples se definen de la siguiente manera: Clave ->->Especialidad y Clave->->Curso; Esto se lee "Clave multidetermina a Especialidad, y clave multidetermina a Curso"

Para eliminar la redundancia de los datos, se deben eliminar las dependencias de valores múltiples. Esto se logra construyendo dos tablas, donde cada una almacena datos para solamente uno de los atributos de valores múltiples.

Para el ejemplo, las tablas correspondientes son:

Tabla Especialidad

| Clave | Especialidad |
|-------|--------------|
| S01   | Sistemas     |
| B01   | Bioquímica   |
| C03   | Civil        |

Tabla ECurso

| Clave | Curso    |
|-------|----------|
| S01   | Natación |
| S01   | Danza    |
| B01   | Guitarra |
| C03   | Natación |

### 1.5.6 Quinta forma normal.

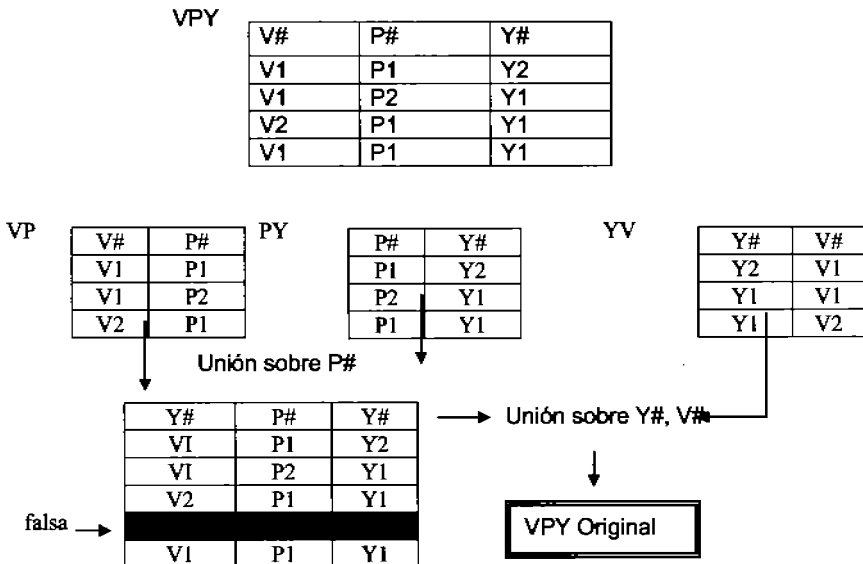


Figura. 2. La relación VPY es la unión de sus 3 proyecciones binarias, pero no de 2 de ellas cualquiera<sup>3</sup>

Se tiene la Relación VPY de la base de datos de proveedores, partes y proyectos. La parte superior de la figura 2 presenta un valor de ejemplo. La Relación VPY es toda llave y no comprende en absoluto DFs o DMVs no triviales (y está por tanto en 4FN). También la figura 2 muestra:

- Las tres proyecciones binarias VP, PY y YV correspondientes al valor de relación VPY que muestra la parte superior de la figura;
- El efecto de juntar las proyecciones VP y PY (sobre P#),
- El efecto de juntar ese resultado y la proyección YV (sobre Y# y V#).

El resultado de la primera junta es producir una copia de la relación VPY original más una tupla (falsa) adicional; y entonces el efecto de la segunda junta es eliminar ese tupla falsa, regresando así a la relación VPY original. En otras palabras, la relación VPY original es descomponible en 3. Nótese que el resultado final es el mismo – independientemente del par de proyecciones que elijamos para la primera junta – aunque el resultado intermedio es diferente en cada caso.

<sup>3</sup> Date, C.J. Introducción a los Sistemas de Bases de Datos. México, 2001. Prentice Hall.

El ejemplo de la figura 2 está expresado en términos de relaciones no de tuplas. Sin embargo, si la tupla satisface cierta restricción de integridad independiente del tiempo, la posibilidad de descomponer VPY en 3 podría ser una propiedad más fundamental independiente del tiempo; es decir, una propiedad satisfecha por todos los valores válidos de la tupla. Para entender cuál debe ser esa restricción, el enunciado de VPY es igual a la junta de sus tres proyecciones, VP, PY y YV<sup>4</sup> equivale precisamente a la siguiente declaración:

Si el par (vl, pl) aparece en VP  
y el par (pl, yl) aparece en PY  
y el par (yl, vl) aparece en YV

Entonces la terna (vl, pl, yl) aparece en VPY

Ya que la terna (vl, pl, yl) aparece en la junta de VP, PY y YV. (la declaración contraria a ésta - si (vl,pl,yl) aparece en VPY entonces (vl,pl) aparece en la proyección VP, etc. - es claramente cierta para cualquier relación VPY de grado 3). Puesto que (vl, pl) aparece en VP si y solamente si (vl,pl,y2) aparece en VPY para cierto y2, y en forma similar para (pl,yl) y (yl,vl), podemos reescribir la declaración anterior como una restricción sobre VPY:

Si (vl, pl, y2), (v2,pl,yl), (vl,p2,y1) aparece en VPY  
Entonces (vl,pl,yl) aparece también en VPY

### Definición Formal

Una tupla R está en 5FN - también llamada forma normal de proyección-junta (FN/PJ) - si y solamente si cada dependencia de junta no trivial<sup>4</sup> válida para R está implicada por las claves candidatas de R.

## 1.6 Herramientas Utilizadas

### 1.6.1 Lenguaje HTML

**(HyperText Markup Language)** es el lenguaje con el que se escriben las páginas Web. Las páginas Web pueden ser vistas por el usuario mediante un tipo de aplicación llamada navegador, por lo tanto que el HTML es el lenguaje usado por los navegadores para mostrar las páginas Webs al usuario, siendo hoy en día la interfaz más extendida en la red.

Este lenguaje nos permite aglutinar textos, sonidos e imágenes y combinarlos a nuestro gusto. Además, y es aquí donde reside su ventaja con respecto a libros o

---

<sup>4</sup> Es Trivial si una de las proyecciones A, B, ..., Z es la proyección identidad de R (es decir, la proyección sobre todos los atributos de R).

revistas, el HTML nos permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto. El HTML se creó en un principio con objetivos divulgativos. No se pensó que la Web llegara a ser un área de ocio con carácter multimedia, de modo que, el HTML se creó sin dar respuesta a todos los posibles usos que se le iba a dar y a todos los colectivos de gente que lo utilizarían en un futuro. Sin embargo, pese a esta deficiente planificación, se han ido incorporando modificaciones con el tiempo, estos son los estándares del HTML. Numerosos estándares se han presentado ya. El HTML 4.01 es el último estándar a septiembre de 2001.

### 1.6.2 Lenguaje De Programación PHP (Hypertext Preprocessor)

El lenguaje PHP – creado en 1994 - es un lenguaje de programación de estilo clásico, es decir un lenguaje de programación con variables, sentencias condicionales, bucles, funciones. No es un lenguaje de marcas como podría ser HTML, XML o WML. Está más cercano a JavaScript o a C.

A diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML<sup>5</sup>.

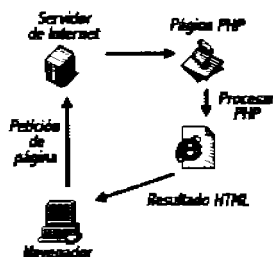


Figura1. Descripción de la ejecución de código PHP

La Figura 1 muestra el funcionamiento de las páginas en PHP alojadas en un servidor y consiste en lo siguiente:

- El navegador del cliente solicita el documento PHP.
- Llega la solicitud del servidor y el servidor localiza el documento, se inicia el intérprete de PHP y ejecuta todo su código.

<sup>5</sup> El Wireless Markup Language es un lenguaje cuyo origen es el XML (eXtensible Markup Language). Este lenguaje se utiliza para construir las páginas que aparecen en las pantallas de los teléfonos móviles y los asistentes personales digitales (PDA) dotados de tecnología WAP.

-Una vez ejecutado el código se genera el resultado en HTML y lo devuelve al servidor para que lo transfiera al cliente.

-El servidor transfiere el resultado en HTML y es mostrado en el navegador del cliente

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

PHP permite la generación dinámica de contenidos en un servidor Web. El significado de sus siglas es HyperText Preprocessor. Entre sus principales características cabe destacar su potencia, su alto rendimiento, su facilidad de aprendizaje y su escasez de consumo de recursos.

El código PHP puede incluirse dentro del código HTML de la página. Para delimitar la sección de código PHP podemos hacerlo de varias formas:

- Usando las etiquetas `<?php` y `<?`
- Usando las etiquetas `<? y ?>`
- Mediante `<script lenguaje="php"> </script>`

### 1.6.3 Servidor Apache

El Servidor Apache HTTP es un servidor Web de tecnología Open Source sólido y para uso comercial desarrollado por la Apache Software Foundation (<http://www.apache.org>). En él se ejecuta el código PHP y se devuelve el resultado en una página HTML.

### 1.6.4 Lenguaje de Programación JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas Web dinámicas.

Una página Web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

Entre las acciones típicas que se pueden realizar en Javascript se tienen dos vertientes. Por un lado los efectos especiales sobre páginas Web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, javascript permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que se pueden crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo.

Javascript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas, etc. Toda esta potencia de Javascript se pone a disposición del programador, que se convierte en el verdadero dueño y controlador de cada cosa que ocurre en la página.

### 1.6.5 Manejador de Bases de Datos. POSTGRESQL

**PostgreSQL** ha tenido una larga evolución, comenzando con el proyecto Ingres en la Universidad de Berkeley. Este proyecto, liderado por Michael Stonebraker, fue uno de los primeros intentos en implementar un motor de base de datos relacional. Después de haber trabajado un largo tiempo en *Ingres* y de haber tenido una experiencia comercial con el mismo, Michael decidió volver a la Universidad para trabajar en un nuevo proyecto sobre la experiencia de Ingres, dicho proyecto fue llamado **post-inges** o simplemente **POSTGRES**.

El proyecto **post-inges** pretendía resolver los problemas con el modelo de base de datos relacional que habían sido aclarados a comienzos de los años 1980. El principal de estos problemas era la incapacidad del modelo relacional de comprender "tipos", es decir, combinaciones de datos simples que conforman una única unidad. Actualmente estos son llamados objetos. Se esforzaron en introducir la menor cantidad posible de funcionalidades para completar el soporte de tipos. Estas funcionalidades incluían la habilidad de definir tipos, pero también la habilidad de describir relaciones - las cuales hasta ese momento eran ampliamente utilizadas pero mantenidas completamente por el usuario. En **POSTGRES** la base de datos "comprendía" las relaciones y podía obtener información de tablas relacionadas utilizando *reglas*.

## Características

### Alta concurrencia

Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

### Amplia variedad de tipos nativos

PostgreSQL provee nativamente soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR
- Direcciones MAC.
- Arrays.
- Disparadores (*triggers*).

Un *disparador* o *trigger* se define en una acción específica basada en algo ocurrente dentro de la base de datos. En PostgreSQL esto significa la ejecución de un procedimiento almacenado basado en una determinada acción sobre una tabla específica. Ahora todos los disparadores se definen por seis características:

- El nombre del trigger o disparador
- El momento en que el disparador debe arrancar
- El evento del disparador deberá activarse sobre...
- La tabla donde el disparador se activara
- La frecuencia de la ejecución
- La función que podría ser llamada

Entonces combinando estas seis características, PostgreSQL le permitirá crear una amplia funcionalidad a través de su sistema de activación de disparadores (*triggers*).

- Vistas.
- Integridad transaccional.
- Herencia de tablas.
- Tipos de datos y operaciones geométricas.

**PL/pgSQL (Procedural Language/PostgreSQL Structured Query Language)** es un lenguaje imperativo provisto por el gestor de base de datos PostgreSQL. Permite ejecutar comandos SQL mediante un lenguaje de sentencias imperativas y uso de funciones, dando mucho más control automático que las sentencias SQL básicas.

Desde PL/pgSQL se pueden realizar cálculos complejos y crear nuevos tipos de datos de usuario. Como un verdadero lenguaje de programación, dispone de estructuras de control repetitivas y condicionales, además de la posibilidad de creación de funciones que pueden ser llamadas en sentencias SQL normales o ejecutadas en eventos de tipo disparador (*trigger*).

Una de las principales ventajas de ejecutar programación en el servidor de base de datos es que las consultas y el resultado no tienen que ser transportadas entre el cliente y el servidor, ya que los datos residen en el propio servidor. Además, el gestor de base de datos puede planificar optimizaciones en la ejecución de la búsqueda y actualización de datos.

Las funciones escritas en PL/pgSQL aceptan argumentos y pueden devolver valores de tipo básico o de tipo complejo (por ejemplo, registros, vectores, conjuntos o incluso tablas), permitiéndose tipificación polimórfica para funciones abstractas o genéricas (referencia a variables de tipo objeto).



## Capítulo 2 Entorno del Problema y Requerimientos del Sistema.

### 2.1 Entorno del Problema

Como se menciona en la introducción, se tenía un sistema previo de historias clínicas el cual funcionaba muy bien para realizar las consultas de primera vez. Cabe mencionar que el alumno me proporcionó el sistema en forma electrónica, es decir la base de datos y los archivos php. De aquí en adelante nos referiremos a este sistema como *Sistema Inicial*.

Al reunirme con el doctor Martín del Campo en el Hospital General, me puso al tanto de cómo se manejaban las consultas, seguimientos e interconsultas en el Depto de salud Mental. Ahí se me proporcionó la hoja de historia clínica breve de primera vez, que como su nombre lo indica es la que se utiliza para anotar los datos de los pacientes que van a su primer visita. Estas visitas al Hospital General fueron de gran ayuda ya que permiten tener un panorama de cómo se realizan las consultas y seguimientos dentro del departamento, ver realmente que datos son los más importantes y la forma en que son capturados, de igual manera se pudo ver con que equipos de cómputo dispone el depto.

El Dr. me comentó que surgió la necesidad de tener un sistema electrónico en donde se lograra capturar toda la información, y los datos estuvieran almacenados y seguros en una base de datos, ya se tenía ese primer sistema pero le faltaban los módulos de seguimiento de pacientes e interconsulta.

Comparando este formato de primera vez con el formato de altas del sistema inicial observe algunos cambios, como el de la supresión de las pruebas de laboratorio, la división del nombre del paciente en apellido Paterno, Materno y Nombre (s) – que esto fue de gran ayuda como se verá mas adelante – y algo muy importante: la inclusión de un **expediente interno** que se asigna al paciente a nivel departamento para tener un mayor control de él.

Con toda esta información se comenzó a analizar el sistema inicial y como funcionaba,

En mi segundo encuentro con el Dr. Martín del Campo me explico mas a detalle la forma en que se realiza el seguimiento de pacientes, - aquí también recibí la ayuda del Dr. Bustamante -jefe del depto de salud mental- quien esta muy interesado en la implementación del sistema- se me proporcionó un formato de seguimiento de pacientes para saber que datos contiene.

Para el seguimiento se tiene un sistema que los doctores llaman SOAP y consiste en S= subjetivo, estado subjetivo del paciente, O = objetivo, estado objetivo del paciente, cambios físicos, evolución del tratamiento, dolor, entre otros, A = análisis

de la situación, y P = plan a seguir, plan terapéutico; en cada visita se sigue este sistema que consiste en áreas de texto para poder escribir todas estas partes del SOAP.

También se tienen datos que vienen en el formato de primera vez tales como tabla de medicamentos, escala de Missouri, signos vitales y plan terapéutico a seguir, debido a que en cada visita subsiguiente se revisarán estos datos para realizar por ejemplo cambios en la dosis o si se presentaron o ausentaron parámetros en la escala Missouri, por mencionar algunos ejemplos.

En el tercer encuentro con el doctor me fue explicado el funcionamiento de la interconsulta.

Una interconsulta tiene lugar cuando un paciente de algún otra área del hospital general necesita atención del departamento de salud mental, puede ser que el medico de salud mental tenga que ir a ver al paciente de la otra área o viceversa.

Esto significa que el paciente ya tiene un numero de expediente que llamaremos de origen, para cuando se realiza la interconsulta se le asigna un numero de expediente local es decir para el departamento de salud mental además de un numero de expediente interno con el propósito de tener un mejor control en los pacientes. Desde luego también un dato importante es el área o servicio de origen así como también el lugar en donde se ubica. Con estos datos se procede a realizar la interconsulta que consiste en aplicar la escala Missouri para poder establecer un diagnostico, recetar algún psicofármaco y si es necesario dar un plan terapéutico.

Al final se indica si el paciente será dado de alta, o seguirá en observación, en este último caso se le programará una nueva cita.

## **2.2 Requerimientos del Sistema**

### **2.2.1 Requerimientos de Software**

#### **Lenguaje HTML**

Se utilizó este lenguaje para la realización de toda la interfaz grafica del sistema. Es de los más utilizados para la realización de páginas Web. En él se puede introducir código en PHP y en javascript lo cual permite realizar consultas a la base de datos, validaciones y todo lo que se necesite realizar en cada pantalla del sistema

### **Lenguaje de Programación PHP.**

Trabajé con la versión 4. Utilicé el software llamado PHP triad que contiene el servidor apache que se necesita para ejecutar el código en PHP. También contiene una versión de mysql que en nuestro caso no se utilizara debido a que trabajé con otro DBMS.

### **Manejador de Bases de datos.**

Utilice el manejador PostgresSQL porque es estable, seguro y además es de código abierto, trabajé con la versión 8.2 que fue la mas estable que encontré cuando lo bajé. Se puede obtener de la pagina [www.postgres.org](http://www.postgres.org) y ahí se puede encontrar también la documentación del manejador y todo lo relacionado a éste.

Otra ventaja de este manejador es que existen versiones tanto para Windows como para Linux, esto es muy importante por si en alguna circunstancia no se lograra adquirir la licencia para el uso de Windows.

### **Lenguaje de Programación JavaScript**

EL lenguaje javascript se utilizó para realizar las validaciones necesarias para los campos de los formularios; por mencionar algunos usos de este lenguaje en el sistema se utilizó para saber si algún campo del formulario esta vacío y requiere ser llenado, el lenguaje entonces despliega una ventana diciendo que falta llenar ese campo y ubica el cursor en él. También se utilizó para que en el campo edad se escribieran solo valores enteros, de igual forma en el campo teléfono se empleó javascript para que el usuario solo pueda teclear números.

Estas validaciones se pudieron haber realizado en lenguaje PHP pero la ventaja que ofrece javascript es que el código se ejecuta en el mismo navegador del usuario a diferencia del código php que tiene que ejecutarse en el servidor por lo que al utilizar javascript se reduce el tiempo de ejecución y se mejora el rendimiento del sistema.

### **Sistema Operativo**

El sistema lo desarrollé en un servidor que tiene el sistema operativo Windows Server 2003, dicho servidor se encuentra en el Departamento de Procesamiento de Señales Digitales en el edificio de Posgrado de la Facultad de Ingeniería.

Todas las pruebas hechas al sistema fueron realizadas sobre el servidor. Desde el hospital realicé la conexión remota hacia el servidor, así como también las inserciones, borrada y actualizaciones que realiza el sistema.

Es pertinente mencionar que se tiene pensado que una computadora funja como servidor ahí mismo en el hospital para que ahí sea donde se aloje la base de datos y a donde se conecten todas las computadoras cliente de cada uno de los médicos.

En dicha computadora se puede tener el sistema operativo Windows - si se puede adquirir la licencia por parte del hospital – de lo contrario se puede instalar Linux) que es software libre) como sistema operativo y ahí alojar el sistema de administración de historias clínicas.

### **2.2.2 Requerimientos de Hardware.**

Como mencioné anteriormente se tiene pensando tener una computadora central en la que se alojen todos los datos. También se necesitan computadoras que cuenten con conexión a Internet.

Para la computadora central (servidor) se pedirían los siguientes requerimientos mínimos:

Procesador Intel Core Duo  
2Gb de memoria RAM  
Tarjeta de red  
Disco duro de 120Gb  
Unidad de cd-rom

Para las computadoras cliente

Procesador Intel Pentium 4 o superior  
  
1gb de memoria RAM  
Tarjeta de red  
Disco duro de 80 GB  
Unidad de CD-ROM

### Capítulo 3. Desarrollo del Seguimiento del Paciente y la Interconsulta

#### 3.1 Migración de datos de MySQL a PostgreSQL. Análisis del sistema.

Para esta versión del sistema se hicieron varios cambios para lograr tener el seguimiento de pacientes, módulo de interconsulta así como también un módulo para los datos de los médicos.

Dichos cambios implicaron el rediseño de la base de datos, muchas de las tablas tuvieron que ser modificadas y también se crearon nuevas.

En esta etapa de la tesis surgió la dificultad de comprender y asimilar lo que hacía el sistema tal y como estaba, esto requirió un tiempo importante ya que se buscaba realizar el mínimo de cambios posibles y lograr que el sistema siguiera funcionando de forma adecuada.

El primer problema que se encontró fue que la base de datos estaba hecha en el DBMS Mysql así que se migró al manejador PostgreSQL versión 8.2.4, ahí se montó el archivo .sql de la base de datos inicial y se comenzó la depuración de las tablas. Esta etapa no requirió mucho tiempo debido a que los cambios a realizar fueron mínimos, lo más importante fue que se logró conservar todos los constraints, nombres de tablas y de columnas.

El siguiente paso fue modificar el archivo de conexión, se trata de una función en el lenguaje de programación PHP, que contiene todos los parámetros para realizar las conexiones a la base de datos: nombre de la base, puerto de conexión, dirección host, usuario y contraseña, dicha función era una conexión al manejador de bases de datos mysql pero tuvo que ser cambiada por una de postgres debido a que se tomó la decisión de emplear PostgreSQL como DBMS.

Teniendo estas 2 etapas realizadas se pudo visualizar el sistema inicial (a nivel local) y ver como funcionaba cada pantalla, se observó que era lo que se tenía que suprimir y que cosas conservar.

Como se ha mencionado el sistema funcionaba muy bien para las consultas de primera vez, sin embargo para el seguimiento se tuvieron que hacer varios cambios.

El primer objetivo era lograr que el sistema aceptara 1, 2, y hasta n visitas, debido a que se tenían los casos en que un paciente puede ir solo una vez y ser dado de alta o ir hasta 20 veces a terapias y/o revisiones por mencionar un ejemplo, así que el sistema tenía que ser flexible en cuanto al número de visitas.

Se creó una tabla llamada VISITA, que contiene 3 campos: idvisita, numexpediente y fecha. Esta tabla es muy importante debido a que contiene un id para las visitas, se asigna uno cada que el paciente va a consulta, este id es

único, el numexpediente es el que se le asigna al paciente desde la primera vez, y la fecha es en la que el paciente fue a consulta.

Estos 3 datos me parecieron los más importantes para que el sistema pudiera soportar múltiples visitas y no cambiar por completo el diseño de la base de datos así se pudo relacionar de una mejor manera esta tabla VISITA con la de PACIENTE

También se creó una tabla llamada SEGUIMIENTO que entre otras cosas contiene los campos del sistema de seguimiento SOAP que se describió en el capítulo anterior, también contiene la evolución del paciente que son de los datos que se registran a partir de la segunda visita que es cuando comienza propiamente el seguimiento. Otros 2 campos importantes de la tabla son los de alta y seguimiento ya que aquí es donde se puede saber si algún paciente sigue en observación o si ya está dado de alta clínicamente. Para esto se utilizó un tipo de dato entero y tiene 2 valores: 0 para cuando el paciente está en observación y 1 para cuando ya ha sido dado de alta.

Otra tabla que se agregó fue la de EFECTOS SECUNDARIOS que también se requieren a partir de la segunda visita, aquí se almacena la información cuando se llega a presentar 1 o más reacciones a causa de los psicofármacos prescritos en la visita anterior así como también su intensidad que puede ser leve, moderado, severo o muy severo.

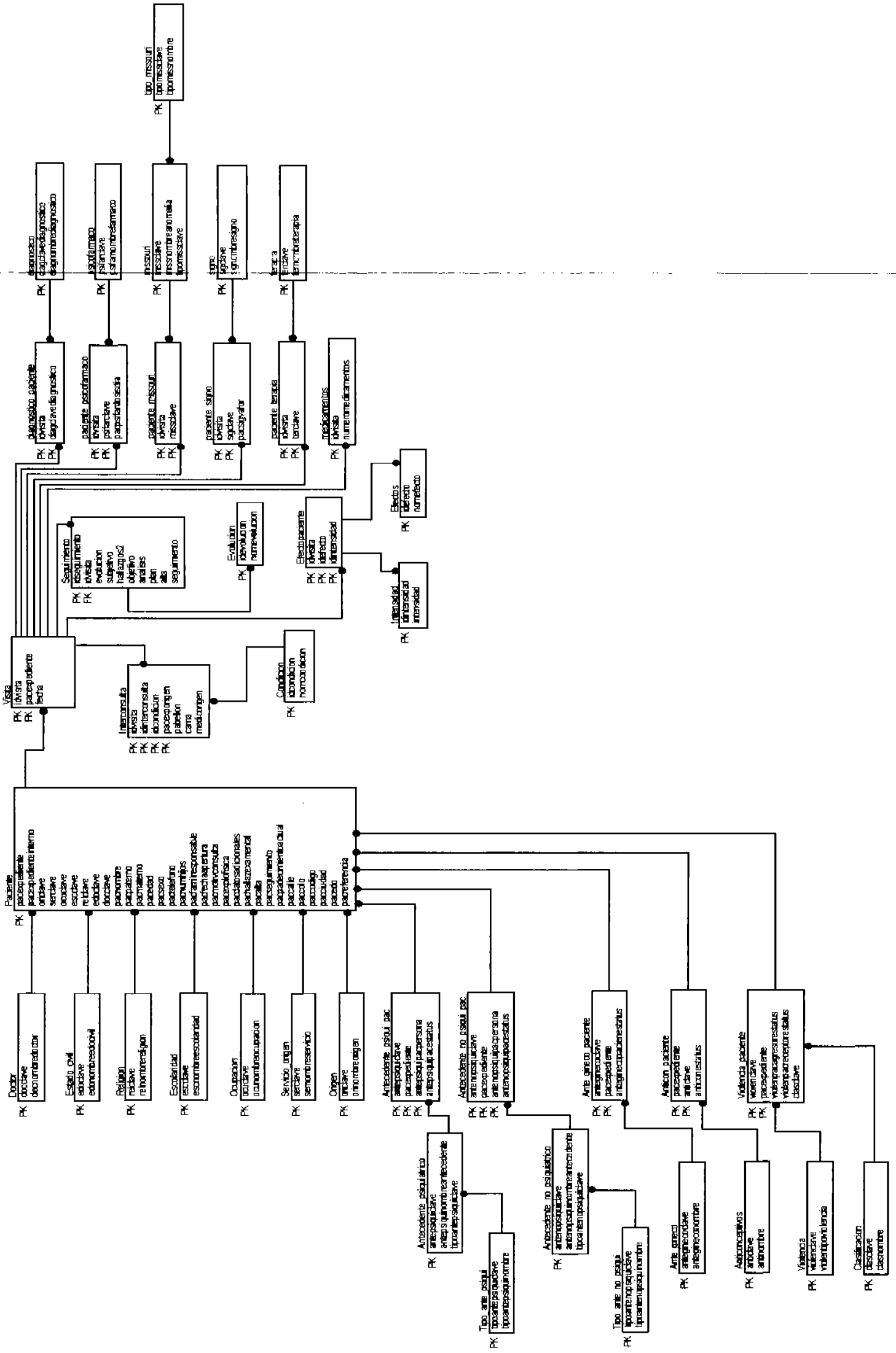
Como se menciona el diccionario de datos, el idvisita es llave primaria ya que es un valor único, es decir no se repite

La tabla VISITA está relacionada con la tabla PACIENTE a través de numexpediente, ya que aquí es llave primaria y en VISITA es foránea. Estas dos tablas son de las principales porque contienen los datos esenciales de altas bajas y seguimiento de pacientes.

Este fue de los cambios más significativos porque así se tuvo la posibilidad de manejar las visitas en una tabla aparte y no dentro de la misma tabla paciente que además ya tiene varios campos como se puede observar en el diagrama E/R. Debido a este cambio las tablas que también resultaron afectadas fueron las de SIGNOS VITALES, ESCALA MISSOURI, DIAGNÓSTICOS, PSICOFÁRMACOS Y TERAPIA, que son tablas que se usan desde la primera visita en adelante, el cambio consistió en sustituir el campo numexpediente por el idvisita para poder saber en qué visita se insertaron los datos de cada tabla,

Los catálogos de estas tablas se encuentran relacionados a través de sus llaves primarias. Los antecedentes psiquiátricos quedaron relacionados con la tabla paciente a través del número de expediente.

### 3.2 Modelo Entidad Relación



Se crearon además las tablas de ANTICONCEPTIVOS y ANTICONPACIENTE, para complementar los antecedentes ginecobstétricos cuando el paciente es una mujer.

La mayor parte del tiempo trabajé con la herramienta PgAdmin III la cual está contenida en la versión 8.2 de Postgres, en ella se pueden crear mucho más rápido las tablas y relaciones sin tener que escribir el código SQL también es más fácil detectar un error en alguna sintaxis, de igual manera las consultas SQL se pueden efectuar ahí, en este caso el editor se divide en 2, una ventana para escribir las consultas en lenguaje SQL y en la otra se muestran los resultados.

Para crear las tablas solo se proporciona el nombre, los nombres de las columnas, el tipo de dato y los constraints sin tener que teclear el código sql, de igual manera al borrar una tabla solo se tiene que oprimir la tecla supr.

También permite visualizar tanto las tablas como sus respectivas columnas, muestra todas sus propiedades incluyendo sus relaciones con otras tablas y finalmente también ofrece la posibilidad de generar el diccionario de datos de cada tabla.

Al principio fue un poco tardado entender el código SQL de la base de datos original pero repasándolo y ejecutándolo en partes fue posible comprender bien como funcionaba la base y las relaciones que existían.



### 3.3 Diccionario De Datos

**Tabla ANTE\_GINECO**

| Nombre           | Tipo de dato | Llave Primaria | Llave Foránea | Descripción                            |
|------------------|--------------|----------------|---------------|--|
| anteginecoclave  | Char(2)      | Sí             | No            | Clave de identificación de antecedente |
| antegineconombre | Varchar(100) | No             | No            | Nombre del antecedente                 |

**Tabla ANTE\_GINECO\_PACIENTE**

| Nombre                | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|-----------------------|-------------|----------------|---------------|----------|--|
| pacExpediente         | Varchar(15) | Sí             | No            | NOT NULL | Clave que identifica a un paciente                 |
| anteginecoclave       | Char(2)     | Sí             | No            | NOT NULL | Clave que identifica a un antecedente ginecológico |
| anteginepacienestatus | Varchar(30) | No             | No            | NOT NULL | Valor del antecedente de un paciente               |

**Tabla ANTECEDENTE\_NO\_PSQUI\_PAC**

| Nombre                 | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|------------------------|-------------|----------------|---------------|----------|--|
| antenopsiquiclave      | Char(2)     | Sí             | No            | NOT NULL | Clave que identifica a un antecedente no psiquiátrico                                    |
| pacExpediente          | Varchar(15) | Sí             | No            | NOT NULL | Clave que identifica a un paciente   |
| antenopsiquipacpersona | Integer     | Sí             | No            | NOT NULL | Número entero que determina si el antecedente no es psiquiátrico es del paciente o de su |

|                        |         |    |    |             |  |
|------------------------|---------|----|----|-------------|--|
|                        |         |    |    |             | familia. 0 =<br>Antecedente del<br>paciente. 1 =<br>Antecedente de la<br>familia del paciente            |
| antenopsiquipacestatus | Integer | No | No | NOT<br>NULL | Numero que<br>determina si el<br>antecedente no<br>psiquiátrico existe,<br>0= No existe<br>1 = Si Existe |

**Tabla ANTECEDENTE\_NO\_PSIQUIATRICO**

| Nombre                       | Tipo        | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción   |
|------------------------------|-------------|-------------------|------------------|-------------|---|
| Antenopsiquiclave            | Char(2)     | Sí                | No               | NOT<br>NULL | Clave que<br>identifica a un<br>antecedente<br>no<br>psiquiátrico                     |
| Antenopsiquinombreamecedente | Varchar(50) | Sí                | No               | NOT<br>NULL | Nombre del<br>antecedente<br>no<br>psiquiátrico                                       |
| tipoantenopsiquiclave        | Char(2)     | No                | No               | NOT<br>NULL | Clave única<br>que identifica<br>q un grupo de<br>antecedentes<br>no<br>psiquiátricos |

**Tabla ANTECEDENTE\_PSIQUI\_PAC**

| Nombre          | Tipo        | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción   |
|-----------------|-------------|-------------------|------------------|-------------|---|
| antepsiquiclave | Char(2)     | Sí                | No               | NOT<br>NULL | Clave única que<br>identifica a un<br>antecedente<br>psiquiátrico |
| pacExpediente   | Varchar(15) | Sí                | No               | NOT<br>NULL | Clave única que<br>identifica al paciente                         |

|                      |         |    |    |          |  |
|----------------------|---------|----|----|----------|--|
| antepsiquipacpersona | Integer | Sí | No | NOT NULL | Número entero que determina si el antecedente psiquiátrico es del paciente o de su familia. 0 = Antecedente del paciente. 1 = Antecedente de la familia del paciente |
| antepsiquipacestatus | Integer | No | No | NOT NULL | Numero que determina si el antecedente psiquiátrico existe, 0= No existe 1 = Sí Existe   |

**Tabla ANTECEDENTE\_PSIQUIATRICO**

| Nombre                      | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-----------------------------|-------------|----------------|---------------|----------|---|
| antepsiquiclave             | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un antecedente psiquiátrico            |
| antepsiquinombreantecedente | Varchar(50) | No             | No            | NOT NULL | Nombre del antecedente psiquiátrico                                 |
| tipantenopsiquiclave        | Char(2)     | No             | No            | NOT NULL | Clave única que identifica q un grupo de antecedentes psiquiátricos |

**Tabla ANTICON\_PACIENTE**

| Nombre        | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                              |
|---------------|-------------|----------------|---------------|----------|--|
| pacexpediente | Varchar(15) | Sí             | No            | NOT NULL | Clave única que identifica a un paciente |

|                |         |    |    |             |  |
|----------------|---------|----|----|-------------|--|
| anticlave      | Integer | Si | No | NOT<br>NULL | Clave única que identifica a un nombre de anticonceptivo                               |
| anticonestatus | Integer | No | No | NOT<br>NULL | Número que determina el uso del anticonceptivo, 0 = No lo utiliza<br>1 = Sí lo utiliza |

**Tabla ANTICONCEPTIVOS**

| Nombre     | Tipo        | Llave Primaria | Llave Foránea | Null        | Descripción  |
|------------|-------------|----------------|---------------|-------------|--|
| anticlave  | Integer     | Si             | No            | NOT<br>NULL | Clave única que identifica a un nombre de anticonceptivo |
| antinombre | Varchar(30) | No             | No            | NOT<br>NULL | Nombre del anticonceptivo utilizado                      |

**Tabla CLASIFICACION**

| Nombre     | Tipo        | Llave Primaria | Llave Foránea | Null        | Descripción   |
|------------|-------------|----------------|---------------|-------------|---|
| clasclave  | Varchar(2)  | Si             | No            | NOT<br>NULL | Clave única que identifica a una clasificación de violencia |
| clasnombre | Varchar(10) | No             | No            | NOT<br>NULL | Nombre de la clasificación de violencia                     |

**Tabla CONDICION**

| Nombre        | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|---------------|-------------|----------------|---------------|----------|--|
| idcondicion   | Integer     | Sí             | No            | NOT NULL | Clave única que identifica a la condición de un paciente |
| nomecondicion | Varchar(20) | No             | No            | NOT NULL | Nombre de la condición de un paciente                    |

**Tabla DIAGNOSTICO**

| Nombre                | Tipo       | Llave Primaria | Llave Foránea | Null     | Descripción  |
|-----------------------|------------|----------------|---------------|----------|--|
| diagclavediagnostico  | Varchar(8) | Sí             | No            | NOT NULL | Clave única que identifica a un diagnóstico de un paciente |
| diagnombrediagnostico | Text       | No             | No            | NOT NULL | Nombre del diagnóstico                                     |

**Tabla DIAGNOSTICO\_PACIENTE**

| Nombre               | Tipo       | Llave Primaria | Llave Foránea | Null     | Descripción  |
|----------------------|------------|----------------|---------------|----------|--|
| diagclavediagnostico | Varchar(8) | Sí             | No            | NOT NULL | Clave única que identifica a un diagnóstico de un paciente |
| idvisita             | Integer    | Sí             | No            | NOT NULL | Clave única de visita de un paciente                       |

**Tabla DOCTOR**

| Nombre          | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                            |
|-----------------|-------------|----------------|---------------|----------|--|
| docclave        | Varchar(4)  | Sí             | No            | NOT NULL | Clave única que identifica a un doctor |
| docnombredoctor | Varchar(80) | No             | No            | NOT NULL | Nombre completo del doctor             |

**Tabla EFECTOPACIENTE**

| Nombre       | Tipo    | Llave Primaria | Llave Foránea | Null     | Descripción   |
|--------------|---------|----------------|---------------|----------|---|
| idvisita     | Integer | Sí             | No            | NOT NULL | Clave única de visita de un paciente                            |
| idefecto     | Integer | Sí             | No            | NOT NULL | Clave única que identifica un efecto secundario                 |
| idintensidad | Integer | Sí             | No            | NOT NULL | Número único que identifica la intensidad del efecto secundario |

**Tabla EFECTOS**

| Nombre    | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                                     |
|-----------|-------------|----------------|---------------|----------|---|
| idefecto  | Integer     | Sí             | No            | NOT NULL | Clave única que identifica un efecto secundario |
| nomefecto | Varchar(50) | No             | No            | NOT NULL | Nombre del efecto secundario                    |

**Tabla ESCOLARIDAD**

| Nombre               | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|----------------------|-------------|----------------|---------------|----------|--|
| esclave              | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un nivel de escolaridad |
| escnombreescolaridad | Varchar(30) | No             | No            | NOT NULL | Nombre del nivel de escolaridad                      |

**Tabla ESTADO\_CIVIL**

| Nombre            | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                                  |
|-------------------|-------------|----------------|---------------|----------|--|
| edoclave          | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un estado civil |
| edonombreedocivil | Varchar(20) | No             | No            | NOT NULL | Nombre del estado civil                      |

**Tabla EVOLUCION**

| Nombre       | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|--------------|-------------|----------------|---------------|----------|--|
| idevolucion  | Integer     | Sí             | No            | NOT NULL | Clave única que identifica a la evolución de un paciente |
| nomevolucion | Varchar(20) | No             | No            | NOT NULL | Nombre de la evolución de un paciente                    |

**Tabla INTENSIDAD**

| Nombre       | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|--------------|-------------|----------------|---------------|----------|--|
| idintensidad | Integer     | Sí             | No            | NOT NULL | Clave única que identifica a la intensidad del efecto secundario |
| intensidad   | Varchar(15) | No             | No            | NOT NULL | Nombre de la intensidad del efecto secundario                    |

**Tabla INTERCONSULTA**

| Nombre          | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-----------------|-------------|----------------|---------------|----------|---|
| idvisita        | Integer     | Sí             | No            | NOT NULL | Clave única de visita de un paciente                        |
| idinterconsulta | Integer     | Sí             | No            | NOT NULL | Clave única que identifica a un paciente de interconsulta   |
| pacexporigen    | Varchar(20) | Sí             | No            | NOT NULL | Clave única que identifica al paciente de una interconsulta |
| idcondicion     | Integer     | Sí             | No            | NOT NULL | Clave única que identifica a la condición de un paciente    |
| pabellon        | Varchar(10) | No             | No            | NOT NULL | Número de pabellón de un paciente de interconsulta          |
| cama            | Varchar(10) | No             | No            | NOT NULL | Número de cama de un paciente de interconsulta              |
| medicoorigen    | Varchar(50) | No             | No            | NOT NULL | Nombre completo del   |



|  |  |  |  |  |                               |
|--|--|--|--|--|-------------------------------|
|  |  |  |  |  | médico del servicio de origen |
|--|--|--|--|--|-------------------------------|

**Tabla MEDICAMENTOS**

| Nombre             | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                                      |
|--------------------|-------------|----------------|---------------|----------|--|
| idvisita           | Integer     | Sí             | No            | NOT NULL | Clave única de visita de un paciente             |
| numeromedicamentos | Varchar(20) | No             | No            | NOT NULL | Numero de psicofármacos prescritos a un paciente |

**Tabla MISSOURI**

| Nombre              | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|---------------------|-------------|----------------|---------------|----------|--|
| misslave            | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a una anomalía dentro de la Escala de Missouri          |
| missnombreaanomalía | Varchar(50) | No             | No            | NOT NULL | Nombre de una anomalía dentro de la Escala de Missouri                             |
| tipomisslave        | Char(2)     | No             | No            | NOT NULL | Clave única que identifica a un grupo de anomalías dentro de la Escala de Missouri |

**Tabla OCUPACION**

| Nombre             | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                                |
|--------------------|-------------|----------------|---------------|----------|--|
| ocuclave           | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a una ocupación |
| ocunombreocupacion | Varchar(25) | No             | No            | NOT NULL | Nombre de la ocupación                     |

**Tabla ORIGEN**

| Nombre          | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-----------------|-------------|----------------|---------------|----------|---|
| oriclave        | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a la Entidad Federativa de donde es originario el paciente |
| orinombreorigen | Varchar(30) | No             | No            | NOT NULL | Nombre de la Entidad Federativa de donde es originario el paciente                    |

**Tabla PACIENTE**

| Nombre               | Tipo        | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción  |
|----------------------|-------------|-------------------|------------------|-------------|--|
| pacexpediente        | Varchar(15) | Sí                | No               | NOT<br>NULL | Clave única que<br>identifica a un<br>paciente   |
| pacexpedienteinterno | Varchar(10) | No                | No               | NOT<br>NULL | Clave única que<br>identifica a un<br>paciente dentro del<br>Departamento de<br>Salud Mental |
| oriclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave de la Entidad<br>Federativa de donde<br>es originario un<br>paciente                   |
| serclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave del servicio de<br>origen de un paciente   |
| ocuclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave de la<br>ocupación de un<br>paciente   |
| escclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave de la<br>escolaridad de un<br>paciente   |
| relclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave de la religión<br>de un paciente   |
| edoclave             | Char(2)     | No                | No               | NOT<br>NULL | Clave del estado civil<br>de un paciente   |
| dooclave             | Varchar(4)  | No                | No               | NOT<br>NULL | Clave del doctor que<br>atiende a un paciente  |
| pacnombre            | Varchar(80) | No                | No               | NOT<br>NULL | Nombre de un<br>paciente   |
| pacpaterno           | Varchar(80) | No                | No               | NOT<br>NULL | Apellido paterno de<br>un paciente   |
| pacmaterno           | Varchar(80) | No                | No               | NOT<br>NULL | Apellido materno de<br>un paciente   |
| pacedad              | Integer     | No                | No               | NOT<br>NULL | Edad de un paciente  |
| pacsexo              | Char(1)     | No                | No               | NOT<br>NULL | Sexo del paciente, M<br>para masculino, F<br>para femenino                                   |
| pactelefono          | Varchar(13) | No                | No               | NOT<br>NULL | Teléfono particular<br>de un paciente  |
| pacnumhijos          | Integer     | No                | No               | NOT<br>NULL | Número de hijos de<br>un paciente  |
| pacfamilresponsable  | Varchar(80) | No                | No               | NOT         | Nombre de un   |

Manual para Mantenimiento y Seguimiento de  
Historias Clínicas para el Departamento de Salud Mental del Hospital General de México

|                       |             |    |    |          |   |
|-----------------------|-------------|----|----|----------|---|
|                       |             |    |    | NULL     | familiar responsable de un paciente   |
| pacfechaapertura      | Date        | No | No | NOT NULL | Fecha de apertura del expediente clínico  |
| pacmotivconsulta      | Text        | No | No | NOT NULL | Motivo que tiene el paciente para una consulta médica   |
| pacexplofisica        | Text        | No | No | NOT NULL | Observaciones hechas de la exploración física hecha a un paciente   |
| pacdatosadicionales   | Text        | No | No | NOT NULL | Información adicional que puede escribir el doctor sobre un paciente  |
| pachallazexamental    | Text        | No | No | NOT NULL | Hallazgos encontrados por el médico en el examen mental hecho a un paciente                                       |
| pacalta               | Integer     | No | No | NOT NULL | Número entero que determina si el paciente fue dado de alta clínica, 0 = Seguirá en Observación, 1 = Alta Clínica |
| pacseguimiento        | Integer     | No | No | NOT NULL | Número entero que determina si el paciente seguirá en observación, 0= Alta Clínica, 1 = Seguirá en Observación    |
| pacpadecimientoactual | Text        | No | No | NOT NULL | Descripción del padecimiento que presenta el paciente en una consulta médica                                      |
| pacelfamilresponsable | Varchar(13) | No | No | NOT NULL | Teléfono del familiar responsable de un paciente  |
| pacalle               | Varchar(50) | No | No | NOT NULL | Calle de un paciente  |
| paccolo               | Varchar(30) | No | No | NOT NULL | Colonia de un paciente  |

|               |             |    |    |          |   |
|---------------|-------------|----|----|----------|---|
| paccodigo     | Integer     | No | No | NOT NULL | Código Postal de un paciente  |
| pacciudad     | Varchar(20) | No | No | NOT NULL | Ciudad en la que vive un paciente                                       |
| pacedo        | Varchar(15) | No | No | NOT NULL | Entidad Federativa donde vive actualmente el paciente                   |
| pacreferencia | Varchar(60) | No | No | NOT NULL | Lugar a donde se enviará a un paciente para una revisión más exhaustiva |

**Tabla PACIENTE\_ MISSOURI**

| Nombre    | Tipo       | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-----------|------------|----------------|---------------|----------|---|
| idvisita  | Integer    | Sí             | No            | NOT NULL | Clave única de visita de un paciente                                      |
| missclave | Varchar(2) | Sí             | No            | NOT NULL | Clave única que identifica a una anomalía dentro de la Escala de Missouri |

**Tabla PACIENTE\_PSIKOFARMACO**

| Nombre            | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-------------------|-------------|----------------|---------------|----------|---|
| idvisita          | Integer     | Sí             | No            | NOT NULL | Clave única de visita de un paciente                    |
| psifarclave       | Varchar(3)  | Sí             | No            | NOT NULL | Clave única que identifica a un psicofármaco            |
| pacpsifardosisdia | Varchar(20) | No             | No            | NOT NULL | Dosis diaria de un psicofármaco prescrita a un paciente |

**Tabla PACIENTE\_SIGNO**

| Nombre      | Tipo        | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción   |
|-------------|-------------|-------------------|------------------|-------------|---|
| idvisita    | Integer     | Sí                | No               | NOT<br>NULL | Clave única de<br>visita de un<br>paciente              |
| sigclave    | Varchar(2)  | Sí                | No               | NOT<br>NULL | Clave única<br>que identifica a<br>un signo vital       |
| pacsigvalor | Varchar(20) | No                | No               | NOT<br>NULL | Valor del signo<br>vital que<br>presenta el<br>paciente |

**Tabla PACIENTE\_TERAPIA**

| Nombre   | Tipo       | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción   |
|----------|------------|-------------------|------------------|-------------|---|
| idvisita | Integer    | Sí                | No               | NOT<br>NULL | Clave única de<br>visita de un<br>paciente                        |
| terclave | Varchar(2) | Sí                | No               | NOT<br>NULL | Clave única<br>que identifica<br>una terapia de<br>rehabilitación |

**Tabla PSICOFARMACO**

| Nombre              | Tipo        | Llave<br>Primaria | Llave<br>Foránea | Null        | Descripción   |
|---------------------|-------------|-------------------|------------------|-------------|---|
| psifarclave         | Char(3)     | Sí                | No               | NOT<br>NULL | Clave única<br>que identifica a<br>un<br>psicofármaco |
| psifarnombrefarmaco | Varchar(50) | No                | No               | NOT<br>NULL | Nombre del<br>psicofármaco                            |

**Tabla RELIGION**

| Nombre          | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                               |
|-----------------|-------------|----------------|---------------|----------|---|
| relclave        | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a una religión |
| relnombreligion | Varchar(20) | No             | No            | NOT NULL | Nombre de la religión                     |

**Tabla SEGUIMIENTO**

| Nombre        | Tipo    | Llave Primaria | Llave Foránea | Null     | Descripción  |
|---------------|---------|----------------|---------------|----------|--|
| idseguimiento | Integer | Sí             | No            | NOT NULL | Clave única de seguimiento de un paciente  |
| idvisita      | Integer | No             | Sí            | NOT NULL | Clave única de visita de un paciente   |
| evolucion     | Integer | No             | No            | NOT NULL | Clave única que identifica a la evolución de un paciente que esta en seguimiento |
| subjetivo     | Text    | No             | No            |          | Descripción de la etapa subjetivo del sistema SOAP de seguimiento                |
| hallazgos2    | Text    | No             | No            | NOT NULL | Descripción de los hallazgos del examen mental realizado a un paciente           |
| objetivo      | Text    | No             | No            | NOT NULL | Descripción de la etapa Objetivo del sistema SOAP de seguimiento                 |
| analisis      | Text    | No             | No            | NOT NULL | Descripción de la etapa Análisis del   |

|             |         |    |    |          |  |
|-------------|---------|----|----|----------|--|
|             |         |    |    |          | sistema SOAP de seguimiento  |
| Plan        | Text    | No | No | NOT NULL | Descripción de la etapa Plan del sistema SOAP de seguimiento   |
| alta        | Integer | No | No | NOT NULL | Número entero que determina si el paciente de seguimiento fue dado de alta clínica, 0 = Seguirá en Observación, 1 = Alta Clínica |
| seguimiento | Integer | No | No | NOT NULL | Número entero que determina si el paciente de seguimiento seguirá en observación, 0= Alta Clínica, 1 = Seguirá en Observación    |

**Tabla SERVICIO\_ORIGEN**

| Nombre            | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|-------------------|-------------|----------------|---------------|----------|--|
| serclave          | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un servicio médico de origen de un paciente |
| sernombreservicio | Varchar(50) | No             | No            | NOT NULL | Nombre del servicio de origen del paciente                               |



**Tabla SIGNO**

| Nombre         | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|----------------|-------------|----------------|---------------|----------|---|
| sigclave       | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un tipo de signo vital |
| signombresigno | Varchar(30) | No             | No            | NOT NULL | Nombre del signo vital                              |

**Tabla TERAPIA**

| Nombre           | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|------------------|-------------|----------------|---------------|----------|--|
| terclave         | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a una terapia de rehabilitación |
| ternombreterapia | Varchar(50) | No             | No            | NOT NULL | Nombre de una terapia de rehabilitación                    |

**Tabla TIPO\_ANTE\_NO\_PSIQUI**

| Nombre                 | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|------------------------|-------------|----------------|---------------|----------|--|
| tipoantenopsiquiclave  | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un grupo de antecedentes no psiquiátricos |
| tipoantenopsiquinombre | Varchar(50) | No             | No            | NOT NULL | Nombre del grupo de antecedentes no psiquiátricos                      |

**Tabla TIPO\_ANTE\_PSIQUI**

| Nombre               | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción   |
|----------------------|-------------|----------------|---------------|----------|---|
| tipoantepsiquiclave  | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un grupo de antecedentes psiquiátricos |
| tipoantepsiquinombre | Varchar(50) | No             | No            | NOT NULL | Nombre del grupo de antecedentes psiquiátricos                      |

**Tabla TIPO\_MISSOURI**

| Nombre        | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|---------------|-------------|----------------|---------------|----------|--|
| tipomisclave  | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un grupo de anomalías dentro de la Escala de Missouri |
| tipomisnombre | Varchar(50) | No             | No            | NOT NULL | Nombre del grupo de anomalías dentro de la Escala de Missouri                      |

**Tabla VIOLENCIA**

| Nombre             | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                                       |
|--------------------|-------------|----------------|---------------|----------|---|
| violenclave        | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un tipo de violencia |
| violentipoviolenca | Varchar(20) | No             | No            | NOT NULL | Nombre del tipo de violencia                      |

**Tabla VIOLENCIA\_PACIENTE**

| Nombre                    | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción  |
|---------------------------|-------------|----------------|---------------|----------|--|
| violenclave               | Char(2)     | Sí             | No            | NOT NULL | Clave única que identifica a un tipo de violencia  |
| pacexpediente             | Varchar(50) | No             | No            | NOT NULL | Clave única que identifica a un paciente   |
| violencpacagresorestatus  |             |                |               |          | Número entero que determina si el paciente presenta el tipo de violencia como agresor, 0 = No lo presenta, 1 = Si lo presenta  |
| violencpacreceptorestatus |             |                |               |          | Número entero que determina si el paciente presenta el tipo de violencia como receptor, 0 = No lo presenta, 1 = Si lo presenta |
| clasclave                 |             |                |               |          | Clave única que identifica a una clasificación de violencia  |

**Tabla VISITA**

| Nombre        | Tipo        | Llave Primaria | Llave Foránea | Null     | Descripción                              |
|---------------|-------------|----------------|---------------|----------|--|
| idvisita      | Integer     | Sí             | No            | NOT NULL | Clave única de visita de un paciente     |
| pacexpediente | Varchar(15) | No             | Sí            | NOT NULL | Clave única que identifica a un paciente |
| fecha         | Date        | No             | No            | NOT NULL | Fecha de visita de un paciente           |

Se tiene también la tabla USUARIOS que no aparece en el diagrama entidad relación debido a que no influye en el diseño de la base de datos. En la tabla se incluye la información personal de los usuarios. Es aquí donde se pueden dar de alta nuevos usuarios, darlos de baja o actualizar su información personal.

**Tabla USUARIOS**

| Nombre    | Tipo         | Llave Primaria | Llave Foránea | Null     | Descripción   |
|-----------|--------------|----------------|---------------|----------|---|
| idusuario | Integer      | Sí             | No            | NOT NULL | Clave única de identificador de un usuario                                      |
| usuario   | Varchar(15)  | No             | Sí            | NOT NULL | Nombre de usuario del Sistema   |
| password  | Varchar(15)  | No             | No            | NOT NULL | Contraseña del Usuario del sistema  |
| nombre    | Varchar(100) | No             | No            | NOT NULL | Nombre real del Usuario   |
| Email     | Varchar(30)  | No             | No            | NOT NULL | Correo Electrónico del Usuario  |
| nivel     | Integer      | No             | No            | NOT NULL | Número que define el nivel de usuario, esto incluye restricciones y privilegios |
| rfc       | Varchar(10)  | No             | No            | NOT NULL | Es el Registro Federal de Contribuyente del Usuario                             |

### 3.4 Elaboración de la Interfaz Grafica

La imagen de fondo para todas las pantallas del sistema se muestra en la figura 3.1. Utilicé el lenguaje PHP para realizar las consultas y despliegue de datos.

Para realizar las pantallas se utilizó código HTML y trabajé en el editor DREAMWEAVER el cual permite visualizar el código y como va quedando la página, esto es muy útil ya que cualquier cambio en el código puede verse reflejado inmediatamente en la página.

En el menú principal del sistema se agregaron las opciones de SEGUIMIENTO, INTERCONSULTA y MEDICOS quedando de la siguiente forma:

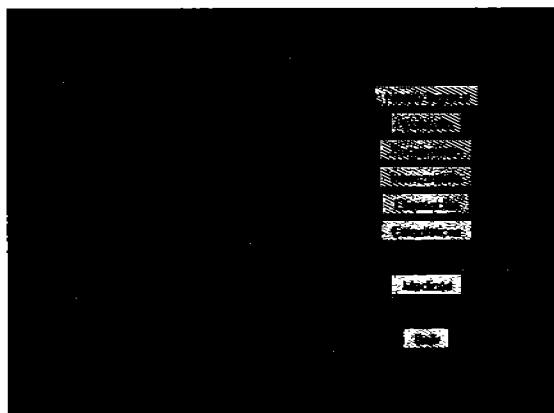


Figura 3.1 Menú Principal del Sistema

Lo primero fue rediseñar la página de altas, ya que como se mencionó con anterioridad hubo cambios en la hoja de historia clínica breve.

Se realizó el formulario con HTML, los primeros campos son nombre del medico tratante, expediente y expediente interno.

La fecha de apertura se muestra en la pantalla para evitar que el médico la teclee, de esta forma se elimina el riesgo de error. Esto fue realizado en PHP, existe una función - date - que proporciona la fecha actual pero en idioma inglés, para solucionar esto implementé un arreglo para traducir la fecha y desplegarla en español con el formato de día, mes y año como se puede observar en la siguiente pantalla:

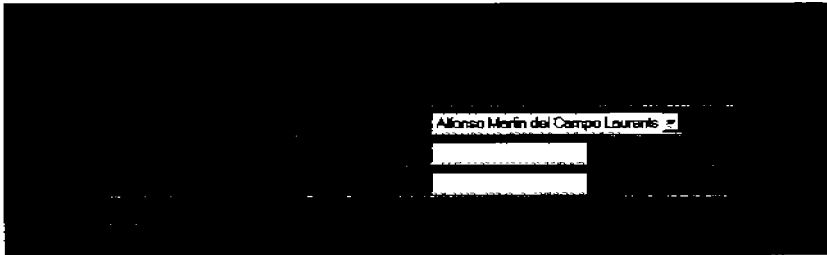


Figura 3.2 Ejemplo del despliegue de la fecha

A continuación viene la ficha de identificación del paciente.

El nombre del paciente esta dividido en apellido paterno, materno y nombre con el fin de facilitar la búsqueda de pacientes ya que como veremos mas adelante se pueden realizar búsquedas por cada uno de estos campos por separado esto ayuda a localizar mas rápidamente algún paciente en especifico o ir filtrando las búsquedas.

Para los campos de edad y el sexo se utilizó la herramienta text y radio respectivamente, el campo radio permite elegir solo una de varias opciones, en este caso 2, masculino y femenino. El domicilio se dividió en calle y numero, colonia, código postal, ciudad y estado.

A continuación vienen los campos de teléfono, lugar de nacimiento, estado civil, servicio de origen, escolaridad religión y ocupación. De éstos con excepción del campo teléfono consiste en menús desplegarles, es decir son catálogos ya preestablecidos que fueron insertados en la base de datos y el usuario elige solo una opción de cada uno de ellos.

Finalmente en esta primera pantalla de nuevo ingreso de pacientes se tienen los campos de familiar responsable y el teléfono de éste mismo, con el fin de tener más referencias del paciente.

Se tienen tres botones que se repiten en las pantallas subsecuentes y son

Siguiente formulario que es para pasar a la siguiente pantalla, cancelar alta y borrar datos es decir para limpiar el formulario y comenzar de nuevo a llenarlo.

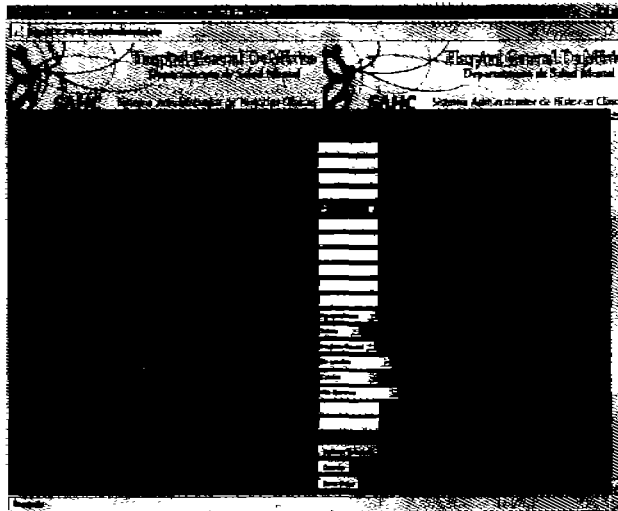


Figura 3.3 Pantalla de Altas

En la siguiente pantalla, se encuentra el campo motivo de la consulta que es de tipo texto debido a que es en donde el medico escribe todos los síntomas que presenta el paciente por lo que no puede ser un campo limitado. Postgres ofrece ese tipo de dato texto que permite tener un campo más amplio de escritura.

El campo padecimiento actual tiene las mismas características que el anterior.

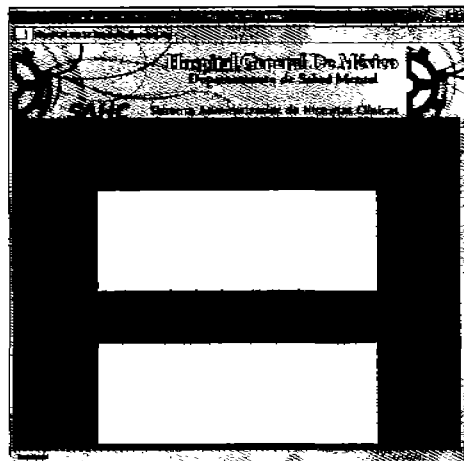


Figura 3.4 Motivo de Consulta y Padecimiento Actual

Se tiene a continuación la parte de antecedentes ginecobstétricos (para cuando el paciente es mujer) en donde se utiliza la herramienta text y checkbox ésta última se explicará más abajo.

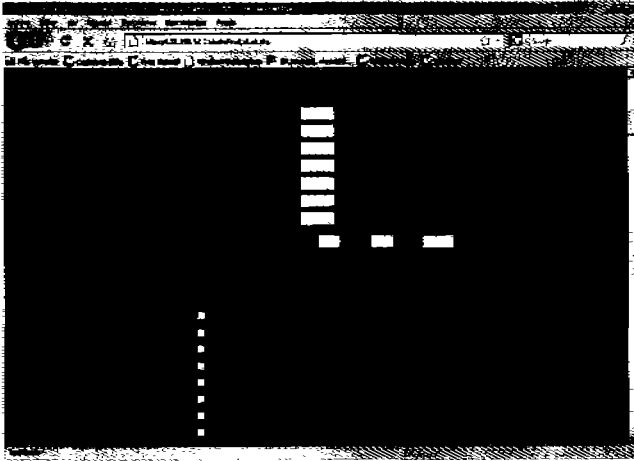


Figura 3.5 Antecedentes Ginecobstétricos

A continuación está la parte de antecedentes personales psiquiátricos y no psiquiátricos.

Se trata de un menú hecho con la herramienta checkbox del lenguaje html. Permite tener 1 de 2 valores que es cuando se selecciona y cuando no, además se le puede asignar un nombre a cada uno de los campos que componen el menú, así se tiene la ventaja de manejar cada valor por separado.

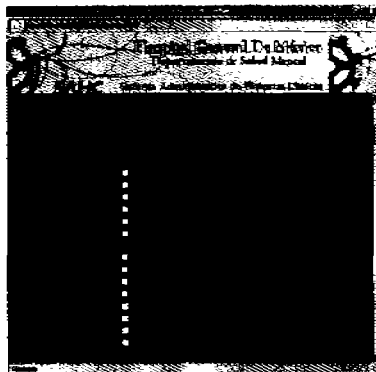


Figura 3.6 Antecedentes Personales Psiquiátricos



Para nuestro caso le dimos el valor de "1" cuando se selecciona cada opción. Esto con el fin de validar este número a la hora de hacer la secuencia de inserción en el lenguaje sql.

De igual forma se realizó el menú para los antecedentes familiares psiquiátricos y no psiquiátricos.

Se tiene en la siguiente pantalla el campo de violencia familiar que se compone de clasificación, tipo y tiempo. En el campo tipo se encuentran 2 opciones de checkbox para cada una de las 4 clasificaciones: receptor y agresor.

Se decidió utilizar checkbox porque puede darse el caso en que sea receptor, agresor, ambas o ninguna, así que solo se debe palomear cuando se da la violencia.

Para el caso de tiempo, se utilizó un option value el cual permite elegir solo una opción de una lista. Las opciones fueron: Actual, anterior o ambas.

Todo se aplico para cada tipo clasificación de violencia: Física, Psicoemocional, sexual y abandono.

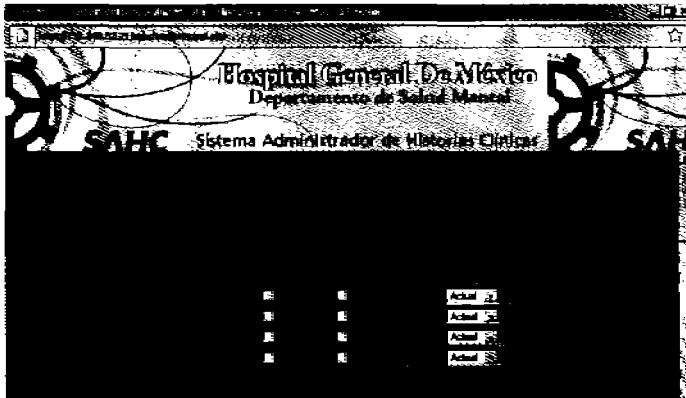


Figura 3.7 Violencia Familiar

Después en la misma pantalla viene la escala Missouri la cual es muy importante ya que al aplicarla el medico puede determinar un diagnostico preciso.

Esta escala la va llenando el medico a partir de la observación que le hace al paciente, va palomeando las señales que el primero detecte.

Aquí convino nuevamente utilizar checkbox para cada elemento de la escala, respetando las clasificaciones que contiene y funciona de igual manera que en los

antecedentes: asignando el valor de 1 en caso de ser seleccionado y a cada campo ponerle su nombre como variable.

Viene a continuación el campo de texto examen mental que funciona igual que el de motivo de consulta.

Después viene la parte de signos vitales que como su nombre lo indica es donde se capturan todas las mediciones hechas al paciente, esto se hace en cada visita y son datos muy importantes para los médicos. Los campos son peso, pulso, talla, presión arterial y respiración. Todos son campos de carácter ya que por ejemplo en el peso puede incluirse un punto para cuando la medida no es exacta.

Finalmente en esta pantalla vienen los campos de hallazgos en la exploración física que es de tipo texto también y un campo llamado datos adicionales en el que el médico puede escribir notas extras o algo que haya omitido anteriormente. Este campo puede quedarse vacío en caso de que no sea necesario escribir algo más.

En la siguiente pantalla se tienen los diagnósticos. Esta lista es la correspondiente al ICD-10 (International Classification of Diseases) que contiene todos los diagnósticos numerados, por lo que los inserté directamente en la base de datos. En la interfaz los coloqué dentro de un option value. Como puede verse es un listado grande así que se redujo el tamaño de la letra para que cupiera en la pantalla. Se puede tener uno o más diagnósticos, por esta razón se tienen varios campos de diagnóstico numerados del 1 hasta el 10.

Posteriormente viene el campo de psicofármacos. Al igual que en diagnósticos se tiene ya un catálogo definido de los posibles fármacos que se pueden prescribir. Por supuesto que se pueden agregar o quitar elementos de la lista, eso se realizaría desde la base de datos y lo haría la persona encargada de la administración del sistema.

Los psicofármacos también se encuentran contenidos en option values y cada uno tiene el campo de la dosis que se expresa en mg. por día, este campo es de tipo texto por lo que se debe teclear la cantidad de dosis administrada. Al igual que en diagnósticos se tienen varios campos de psicofármacos y también se incluye el campo del número de psicofármacos prescritos.

Se tiene después el plan terapéutico a seguir que es un listado de todas las posibles terapias a las que se le puede mandar al paciente, igualmente están en option values, y puede recomendarse una o varias terapias.

A continuación viene el campo de referencia que indica si el paciente debe ser canalizado a otro hospital, ahí se escribirá entonces la ubicación de éste, en dado caso de que no se haga alguna referencia puede escribirse "ninguna".

Finalmente se debe indicar si el paciente se dará de alta o si se le dará una nueva cita para comenzar su seguimiento. Esto lo realice con la herramienta de radio que permite elegir solo una opción entre varias, a diferencia de la herramienta checkbox que elige una o varias.

### 3.5 Seguimiento de pacientes.

Una vez que se realizó la consulta de primera vez y en el caso de que se haya dado una nueva cita, el paciente pasa a seguimiento.

En el menú principal aparece la opción Seguimiento y a continuación una pantalla en la que se pide se proporcione el número de expediente, expediente interno, nombre, apellido paterno o materno. Para que se comience el seguimiento el paciente debe existir en la base de datos, es decir al menos debe haber ido una vez a consulta. En caso contrario aparecerá un mensaje de que el paciente no existe y se tendrá que registrar en la opción de nuevo ingreso. Para realizar esto se hace una consulta a la base de datos buscando los registros mas parecidos a los que se escribieron en los campos correspondientes para esto se utilizó una consulta en lenguaje SQL utilizando la palabra LIKE .

Si el paciente existe entonces aparecen el o los nombres que se hayan encontrado en la búsqueda pero para diferenciarlos esta el número de expediente y también el expediente interno ya que como se mencionó anteriormente estos valores son únicos.

Al seleccionar el paciente que va a seguimiento se pasa a la siguiente pantalla en la que se muestra su nombre, num. de expediente, y el interno. Para tener siempre presente sus datos.

Después se muestra la fecha actual, es decir del día en que se esta realizando el seguimiento, al igual que en el formato de primera vez no es necesario teclear la fecha ya que el sistema la despliega y la inserta por default.

Después viene el campo de evaluación que es un conjunto de opciones hechas con el elemento radio, es decir solo se escoge una, las opciones son: excelente, muy buena, buena, regular y mala.

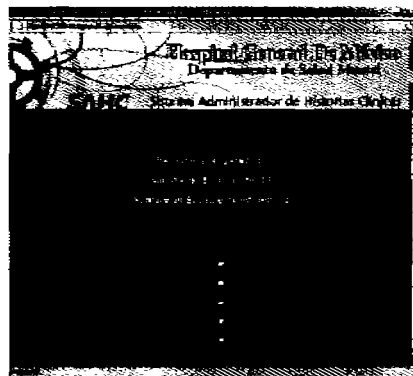


Figura 3.8 Evolución del Paciente

La siguiente parte es la de signos vitales, aquí se miden nuevamente los signos del paciente, aparecerán escritos los signos vitales del paciente en su visita inmediata anterior esto con el fin de que el doctor los compare e incluso si alguno coincide dejarlo así como está o de lo contrario escribir el nuevo valor.

Cabe aclarar que no se trata de un UPDATE sino de una visualización de los datos de la visita anterior y se hará una nueva inserción en la visita actual, la cual a su vez será visualizada en la siguiente visita si la hay.

Para el campo de evolución sucede algo similar es decir aparece la opción seleccionada de la visita anterior, excepto para la segunda visita ya que como es la primer consulta de seguimiento es la primera vez que se va a elegir alguna opción de evolución.

#### Subjetivo

Esta es la primera etapa del sistema SOAP utilizado para el seguimiento, es un campo de texto para que el doctor escriba todo lo relacionado a este espacio, al igual que los signos vitales también aquí aparece a partir de la tercera visita los datos de la visita anterior.

#### Hallazgos del examen mental, Objetivo y Análisis

También son campos de texto para los médicos, el Objetivo y el Análisis son la segunda y tercer etapa respectivamente del método SOAP y también aparecen los datos de la visita inmediata anterior (a partir de la segunda)

#### Escala Missouri

A continuación viene la escala Missouri. Es la misma que se aplicó en la primer visita y de igual manera aparecen palomeados los de la visita anterior con el fin de que si se vuelve a presentar algún aspecto de la escala se deja seleccionando de lo contrario se le da clic para quitar la selección y por supuesto también se pueden seleccionar nuevos síntomas de la escala.

#### Efectos secundarios

Esta tabla es para saber si existió alguna reacción a algún psicofármaco y en que grado se presentó ésta. En la segunda visita (primer seguimiento) aparece sin llenar debido a que apenas se va a saber si hubo reacciones secundarias a los psicofármacos recetados en la primera vez, ya en las siguientes visitas aparecerán mostrados los datos de la anterior.

## Diagnósticos

En la siguiente pantalla viene la parte de diagnósticos que es un formato similar al de la primera vez, con 10 campos y en cada uno de ellos el listado de todos los posibles diagnósticos.

## Plan

Viene después la última parte del SOAP, el Plan, que consiste en un cuadro de texto en el que los doctores escriben todo lo referente a esa parte del SOAP.

## Medicamentos y Plan Terapéutico

Después viene el campo de psicofármacos (como en la consulta de primera vez) junto con el campo para escribir la dosis. A continuación viene el campo número de medicamentos y finalmente el plan terapéutico a seguir, en todos estos campos mencionados. Aparecerán elegidos los que se hayan seleccionado en la visita anterior. Por poner un ejemplo en el plan terapéutico aparecen preseleccionados algunos y en esta visita se podrán quitar en caso de que ya no sean necesarios o también se podrá agregar uno nuevo y se guardara todo en esa visita.

Finalmente viene el campo para elegir si el paciente será dado de alta o continuara en observación, y después el botón para guardar la historia clínica.

Aparecerá una leyenda mencionando el nombre del paciente y su situación.

Se tiene por último el botón regresar el cual lleva al menú principal.

Para poder lograr que aparecieran seleccionadas las opciones de las visitas anteriores en las opciones fue necesario hacer una consulta en la base datos y el resultado serviría como valor del elemento del formulario. Como se ve en el siguiente ejemplo para la escala Missouri:

```
<td><input type="checkbox" name="perseverante" value="1" <?
for($i=0;$i<$num1; $i++) if($datos[$i][0] == 'Perseverante') echo "checked";
?> >
  Perseverante</td>
```

Este código nos indica primero que se trata de un elemento checkbox, su nombre de identificador es perseverante y su valor es 1 cuando se selecciona. Después

viene la validación con el código en PHP que indica que si alguno de los resultados de la consulta realizada previamente a la base datos - cuyos resultados fueron asignados a un arreglo - es igual a la variable Perseverante entonces la selecciona, de lo contrario se deja sin seleccionar. Así se realizó la comparación para cada elemento de la escala Missouri.

Para el caso de la tabla de efectos secundarios se realizaron 2 comparaciones. La primera fue para saber si el efecto secundario se presentó en la visita pasada, esto se realizó así:

```
<td> <input type="checkbox" name="nausea" value="1" <? $va = 0;
    for($i=0;$i<$num2; $i++) {
        if($datos2[$i][0] == "Náusea") {
            echo "checked";
            $va = 1;
        }
    }
?> > Nausea </td>
```

Que es similar a la hecha para la escala Missouri

Para la intensidad en que se presenta dicho efecto se tiene:

```
<?
$queryuno = "select * from Intensidad";
$resultbe= pg_exec($conexion, $queryuno);
$nume= pg_numrows($resultbe);
//echo $num;
$bandera = true;
for($i=0;$i<$nume; $i++)
{
    $datosal=pg_fetch_array($resultbe);

?>

<option value= " <? echo $datos["idintensidad"] ?> " <? if ($bandera && $v
= 1 && $datos1[$cont][0] == $datos["idintensidad"])
```

```

    {
        echo "selected";

        $cont = $cont + 1; //aumento

        $bandera = false;

    }
    <? > <? echo
    $datos["intensidad"]; > </option>
}
    <?
    >

```

La primera parte indica que se trata de un conjunto de opciones, después viene una sentencia if de comparación que revisa si en el resultado de la consulta existe el valor idintensidad, si existe se selecciona la opción y se aumenta un contador en uno, el contador es para ir revisando uno por uno cada uno de los resultados de la consulta, esto fue debido a que el numero de efectos secundarios es variable así que por ejemplo si se tienen solo 5 efectos secundarios el tope del contador es 5 se inicia en 1 y en el momento que se haya presentado el efecto el contador aumenta en 1, en el momento en que llegue a 5 se termina de comparar y termina la validación de la parte de efectos secundarios.

Para el caso de diagnósticos fue igual que para la intensidad del efecto secundario. Aquí un ejemplo.

```

<?
    $query1 = "select * from diagnostico order by
diagNombreDiagnostico";
    $resul= pg_exec($conexion, $query1);
    $num= pg_numrows($resul);
    $bandera = true;
    for($i=0;$i<$num; $i++)
    {
        $datos=pg_fetch_array($resul);
        >

<option value="<? echo $datos["diagclavediagnostico"] >? "
<? if ($cont < $tope && $bandera && $var ==1 && $arreglo3[$cont][0] ==
    $datos["diagclavediagnostico"])
    {

```

```
        echo "selected";

        $cont = $cont + 1;

        $bandera = false;

    }

?> >
<? echo $datos["diagnombrediagnostico"] ?> </option>
```

Se tiene el contador para saber en que momento termina de comparar.

La bandera se inicializa en true antes de la validación y en le momento de la comparación y asignación de valor se cambia a false para que deje de comparar los demás valores, ya que como se ve en el código esta contenido dentro de una sentencia for que va barriendo todo el arreglo que contiene el resultado de la consulta sql.

Esta misma forma de recuperar los datos se realizó para el campo de psicofármacos y de terapias.

Para los campos de texto, es decir los de nombre, expediente, expediente interno, los del sistema SOAP, se validó si existía algún elemento en el resultado de la consulta, de ser cierta la condición se imprime en el campo dicho resultado sino se deja vacío.

Como ejemplo se tiene el código para recuperar los signos vitales del paciente de la visita anterior.

Se busca en la base de datos:

```
$query3 = "select * from paciente_signo where ldivisita =
'$valor'";

$result= pg_exec($conexion, $query3);
$num3= pg_numrows($result);

if($num3)
{
    $i = 0;
    if ($result)
    {
        while ($lineaC = pg_fetch_row($result) )
        {
```



```
$arregloC[$i++] = $lineaC;
    }
}

$peso = $arregloC[0][2];
$pulso = $arregloC[1][2];
$talla = $arregloC[2][2];
$tas = $arregloC[3][2];
$stad = $arregloC[4][2];
$respiracion = $arregloC[5][2];

?> }
```

Se le asignan variables a cada elemento del ArregloC  
En los campos del formulario se escribe el contenido preguntando primero si existe en el value.

```
<input type="text" name="peso" size="5" maxlength="6" value = "<?
if($num3) echo $peso; ?>" >
```

```
<input type="text" name="pulso" size="5" maxlength="6" value = "<?
if($num3) echo $pulso; ?>" onKeyPress="javascript:return
admitir_numeros();">
```

```
<input type="text" name="talla" size="5" maxlength="6" value = "<?
if($num3) echo $talla; ?>" >
```

```
<input type="text" name="TA_S" size="5" maxlength="5" value = "<?
if($num3) echo $tas; ?>" onKeyPress="javascript:return
admitir_numeros();">
```

```
<input type="text" name="TA_D" size="5" maxlength="5" value = "<?
if($num3) echo $stad; ?>" onKeyPress="javascript:return
admitir_numeros();">
```

```
<input type="text" name="respiracion" maxlength="5" size="5" value = "<?
if($num3) echo $respiracion; ?>" onKeyPress="javascript:return
admitir_numeros();">
```

Si no existe el resultado entonces aparece vacío el campo.

### 3.6 Interconsulta

La interconsulta como se mencionó anteriormente se realiza cuando se requiere el servicio del depto. de salud mental en algún otra área del hospital, por lo que el manejo de los números de expediente para este caso es de suma importancia,

En el menú principal se tiene la opción de interconsulta y al elegirla aparece el formato para ingresar los datos del paciente.

La primera parte se integra de numero de expediente de origen, es decir el que se le asignó al paciente en el área que solicitó la interconsulta, el expediente local y el interno también local del dpto de salud mental, la fecha que como en los otros formatos ya la da el sistema automáticamente, el nombre del medico del servicio de origen y el nombre del medico del depto. de salud mental.

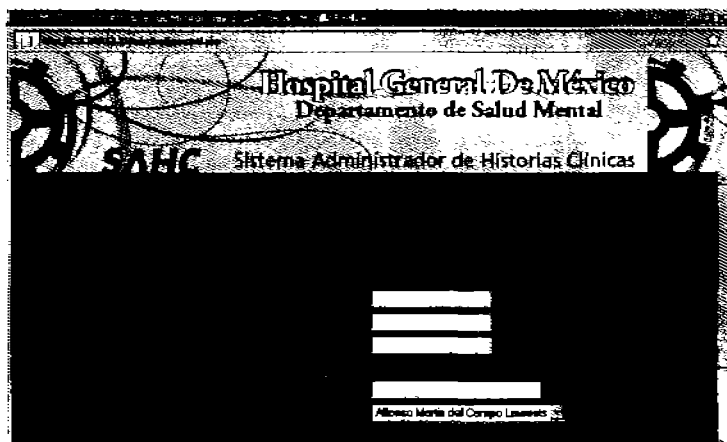


Figura 3.9 Identificación del paciente de interconsulta

Estos primeros campos son para la identificación del paciente.

A continuación vienen los datos del paciente:

Nombre, Apellido Paterno, Apellido Materno, Edad, Sexo.

El siguiente campo es servicio de origen que consiste en una lista de opciones de todos los servicios del hospital. Finalmente aparece número de pabellón y número de cama para complementar los datos de identificación del paciente.

Después viene un campo de texto de motivo de consulta.

En al siguiente pantalla se tiene la escala Missouri que se le aplicara al paciente y función de igual forma que en los otros formatos, es decir palomeando los síntomas que se presenten.

A continuación viene el cuadro de diagnósticos, tal y como aparece en los formatos de primera vez y de seguimiento, de igual forma también se tiene la parte de psicofármacos y plan terapéutico.

Al final de la pantalla se tiene la opción de dar de alta al paciente o darle otra cita. En cualquier caso en la siguiente pantalla aparecerá impresa la situación del paciente y también se guardaran todos los datos en la base del sistema.

Como puede verse la interconsulta es un formato corto debido a que no es un paciente que pertenezca al depto. de salud mental. Se incluyeron partes de los otros formatos como son los diagnósticos, planes terapéuticos y psicofármacos. Las diferencias con los demás pacientes son los datos de antecedentes psiquiátricos, signos vitales, entre otros.

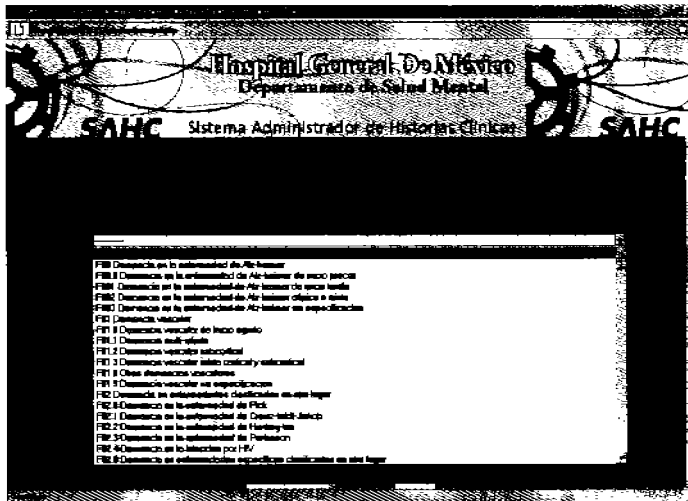


Figura 3.10 Lista de Diagnósticos.

### 3.7 Borrado de Expediente de un paciente.

En esta parte del sistema se borra toda la información del paciente, esto incluye todas las visitas a las que haya asistido, para borrar se utilizó la instrucción SQL DELETE.

Primero se debe saber cuantas y cuales son las visitas del paciente con ayuda del número de expediente interno. Después estos números de idvisita se almacenan en un arreglo. Teniendo esto, se procede a borrar de cada tabla todos los datos donde el idvisita sea igual a todos los contenidos en el arreglo. Para recorrer el arreglo se usa la sentencia FOR.

Para los datos de consulta de primera vez se utiliza solamente el número de expediente el cual se utiliza para indicar a las tablas que se borre la información en donde el número de expediente sea igual al que se desea borrar.

```
$query = "select idvisita from visita where pacexpediente =  
"$numExpediente";
```

```
$resul = pg_exec($conexion, $query);  
$num = pg_numrows($resul);  
//echo $num;  
  
$i=0;  
while ($datos = pg_fetch_row($resul))  
{  
    $arreglo[$i++] = $datos;  
}
```

Los números idvisita se obtienen de la tabla visita a través del número de expediente y se guardan en el \$arreglo y para borrar se usa la instrucción en lenguaje sql DELETE, todo va dentro de un ciclo for para recorrer el arreglo.

En el siguiente ejemplo se borrará la información de terapias y signos vitales del paciente en todas sus visitas.

```
for ($i=0; $i<$num; $i++)  
{  
    $query4 = borrar($conexion, "delete from paciente_terapia where  
idvisita = ".$arreglo[$i][0]);  
  
    $query5 = borrar($conexion, "delete from paciente_signo where idvisita  
= ".$arreglo[$i][0]);}
```

Para borrar datos de la consulta de primera vez solos e compara el numero de expediente. Por ejemplo para borrar los antecedentes psiquiátricos del paciente se realizó de la siguiente forma:

```
$query11 = borrar($conexion,"delete from antecedente_psiqui_pac  
where pacExpediente = '$numExpediente');
```

### 3.8 Módulo de Médicos

Esta opción también se le agregó para que los médicos pudieran cambiar su contraseña, Es muy recomendable cambiarla periódicamente por razones de seguridad, ya que los médicos tiene los privilegios de insertar información en la base datos así como también consultarla y borrarla por lo que no cualquier persona puede entrar al sistema.

Al entrar a la pantalla se pide nuevamente al medico que ingrese su nombre de usuario y contraseña para confirmar su identidad.

Si la información es incorrecta se indicara en la pantalla y se le pedirá al medico que ingrese nuevamente sus datos.

Si son correctos viene una opción aparecerán desplegados en pantalla junto con una opción para cambiar los datos, si se elige ésta ultima entonces aparecerán el nombre del médico, su RFC, el nombre de usuario, correo electrónico y la contraseña en formato password por seguridad, aquí es donde se pueden cambiar o corregir algún dato del médico y finalmente se guardarán todos los cambios.

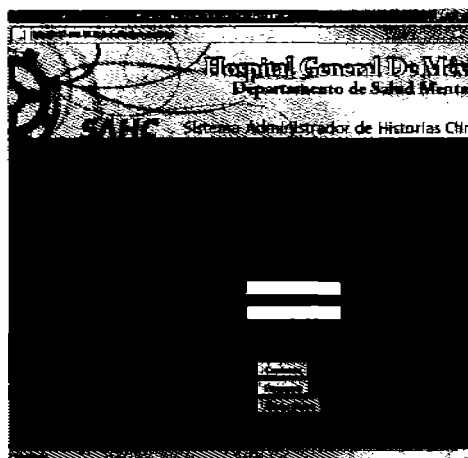


Figura 3.11 Primer Pantalla del Menú Médicos

Dado que la información personal de los médicos no tiene relación con los pacientes, esta tabla no se incluye en el diagrama entidad relación.

Para mostrar los datos del médico se hace una consulta a la tabla Usuarios y los resultados se colocan en un arreglo, de ahí se guardan en variables cada uno de los elementos para poder desplegarlos en el campo donde les correspondan.

Para realizar los cambios se utilizó el comando SQL UPDATE que permite sobrescribir los valores de la tabla.

## Conclusiones

Se realizó un Sistema Administrador de Historias Clínicas con el que es posible dar de alta nuevos pacientes del Departamento de Salud Mental así como también llevar un seguimiento de todos ellos. Con el sistema también es posible realizar interconsultas de pacientes y los registros de pacientes que ya hayan sido dados de alta clínica o que ya no regresaron, pueden ser borrados.

Una de las principales ventajas del sistema es que el médico puede ver toda la información de alguna consulta previa del paciente, así puede darse cuenta de la evolución de éste y comparar los datos en el momento en que está ingresando la nueva información.

Se realizó un manual de utilización del sistema el cual explica cada una de las partes que lo componen, el manual está dirigido a los médicos y al personal del departamento.

También se anexó un apéndice con algunas de las funciones en lenguaje PHP utilizadas en la realización del sistema como ejemplo de aplicación y apoyo a alumnos que cursan la materia de Bases de Datos

El sistema ya fue presentado a los médicos del departamento de salud mental y sólo falta presentarlo a las autoridades del Hospital para que pueda ser puesto en marcha.

Actualmente el sistema está alojado en un servidor que se encuentra en la Facultad de Ingeniería, se hicieron pruebas al sistema desde el Hospital General mediante una conexión remota a la dirección IP que tiene el servidor.

Se le agregó al sistema una opción en la que los médicos pueden ver y actualizar sus datos incluyendo la contraseña la cual se recomienda que la cambien periódicamente por razones de seguridad.

Este sistema como cualquier otro puede ser mejorado, se pretende agregarle elementos a la solicitud de primera vez y al formato de seguimiento, lo cual cambiaría un poco el diseño de la base de datos pero sin tener que rediseñarla completamente por lo que en ese aspecto el sistema es flexible.

También se tiene como objetivo (cuando se implemente el sistema en el departamento) realizar y desplegar gráficas de las estadísticas presentadas con el fin de que los médicos puedan visualizar mejor los datos, un objetivo más es exportar los datos de las estadísticas al paquete Microsoft Excel para realizar análisis estadístico utilizando esta herramienta.

Ventajas del Sistema:

- La principal ventaja del sistema es que se utilizó software libre (PHP y PostgreSQL por lo que no resultaría muy costoso implementarlo, los gastos estarían enfocados solamente al hardware.
- El manejador PostgreSQL es seguro, gratuito y además permite guardar imágenes.

Desventajas del Sistema:

- Si no se tiene en el Hospital una buena conexión de Internet las páginas tardarían un poco más de tiempo en abrirse; se piensa tener todo en una red local teniendo una computadora que reciba y guarde todos los datos y realice respaldos de información en periodos previamente establecidos.

Finalmente en este trabajo de tesis utilicé los conocimientos obtenidos a lo largo de la carrera para desarrollar y mejorar el sistema y lograr que tuviera un desempeño óptimo para beneficio de los pacientes del Departamento de Salud Mental del Hospital General de México.



## Glosario de Términos

**DBMS:** (Database Management System), Sistema de Gestión de Bases de Datos. Es un software con el cual se pueden crear bases de datos y trabajar con ellos. Existen varios manejadores de bases de datos, como por ejemplo PostgreSQL, Oracle, Sybase.

**HTML:** (HyperText Markup Language). Lenguaje de etiquetas que se utiliza para la creación de páginas Web.

**Checkbox:** Elemento de diseño Web del lenguaje Html, que permite al usuario hacer selecciones múltiples de un conjunto de opciones.

**Option Value:** Elemento Html de diseño Web que consiste en una lista desplegable de varias opciones. Puede permitir la selección de una sola opción o de múltiples opciones (si se mantiene presionada la tecla **Ctrl** durante dicha selección).

**Javascript:** Lenguaje de programación que se utiliza principalmente para crear páginas Web dinámicas.

**PHP:** (Hypertext Preprocessor) Lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo Web y puede ser embebido dentro de código HTML.

**PostgreSQL:** Sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la Universidad de Berkeley.

**SOAP:** (Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

**SQL:** (Structured Query Language), Lenguaje declarativo de acceso a Bases de Datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Además es un lenguaje no procedural que optimiza las sentencias de SQL.

**XML:** (Extensible Markup Language) Es un lenguaje que permite definir la gramática de lenguajes específicos, por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

### **Terminología Médica mencionada en la Tesis**

**Escala de Missouri:** Escala validada y publicada a nivel de la literatura internacional, donde se hace una evaluación del estado mental que permite detectar ciertos síntomas del paciente y a partir de la observación realizada dar un diagnóstico. Además, esta escala puede utilizarse de manera repetida y comparar la evolución del cuadro clínico.

**ICD-10:** (International Statistical Classification of Diseases and Related Health Problems 10th Revision). Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados a la Salud. Es un listado de los padecimientos, signos, síntomas, y causas externas de enfermedad, hecho por la Organización Mundial de la Salud. Lo utilizan los médicos para establecer un diagnóstico específico a un paciente.

**Interconsulta:** Consulta médica que realiza el departamento de salud mental a pacientes de otros departamentos dentro del Hospital General de México.

**Plan Terapéutico:** Terapia o terapias a las que puede ser canalizado un paciente para su tratamiento.

**SOAP:** Sistema de seguimiento de pacientes utilizado en el Departamento de Salud Mental, consiste en Subjetivo del paciente, Objetivo, Análisis y Plan.

## BIBLIOGRAFIA Y MESOGRAFIA

1. Ullman, Jeffrey D. Introducción a los Sistemas de Bases de Datos. México, 1999. Prentice Hall.
2. Date, C.J. Introducción a los Sistemas de Bases de Datos. México, 2001. Prentice Hall.
3. Gutiérrez Abraham. PHP 4 A través de ejemplos. México, 2004. Alfaomega.
4. <http://www.php.net>
5. <http://www.postgres.org>
6. <http://es.wikipedia.org/wiki/PL/PgSQL>
7. <http://www.webestilo.com/php/php00.phtml>
8. <http://www.manualdephp.com/manualphp/introduccion-php.html>
9. <http://es.wikipedia.org/wiki/WML>
10. <http://www.librosweb.es/javascript/capitulo1.html>
11. [http://es.wikipedia.org/wiki/C%C3%B3digo\\_HTML](http://es.wikipedia.org/wiki/C%C3%B3digo_HTML)
12. <http://www.desarrolloweb.com/articulos/490.php>
13. <http://www.gb.nrao.edu/pubcomputing/redhatELWS4/RH-DOCS/rhel-rg-es-4/ch-httpd.html>
14. <http://www.ignside.net/man/php/>
15. <http://www.mitecnologico.com/Main/DiagramasEntidadRelacionER>
16. [http://www.lsgames.com/tmp/SQL/ModeloER\\_Access.htm#paso\\_a\\_tablas](http://www.lsgames.com/tmp/SQL/ModeloER_Access.htm#paso_a_tablas)
17. [http://www.netpecos.org/docs/mysql\\_postgres/x15.html](http://www.netpecos.org/docs/mysql_postgres/x15.html)

## Apéndice A. MANUAL DE USUARIO.

### Página de inicio.

#### Autenticación del usuario

Escribir el nombre de usuario y contraseña

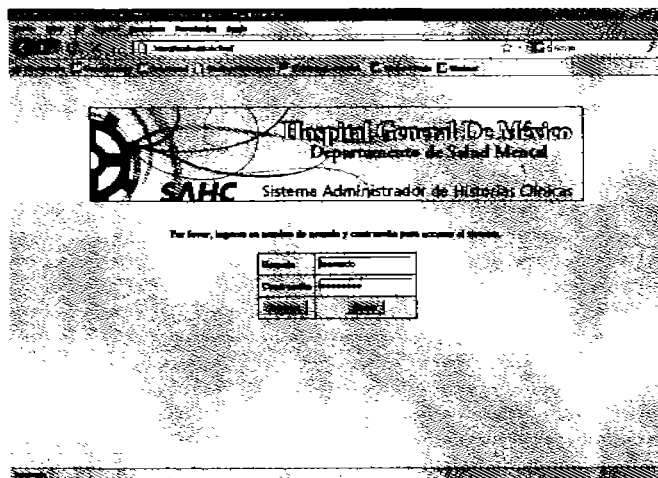


Figura. 1. Pantalla Principal

Si la contraseña es incorrecta aparecerá un mensaje y tendrá que volver a ingresar los datos.



Figura. 2 Datos incorrectos.

Si la contraseña es correcta se pasará al menú principal del sistema:

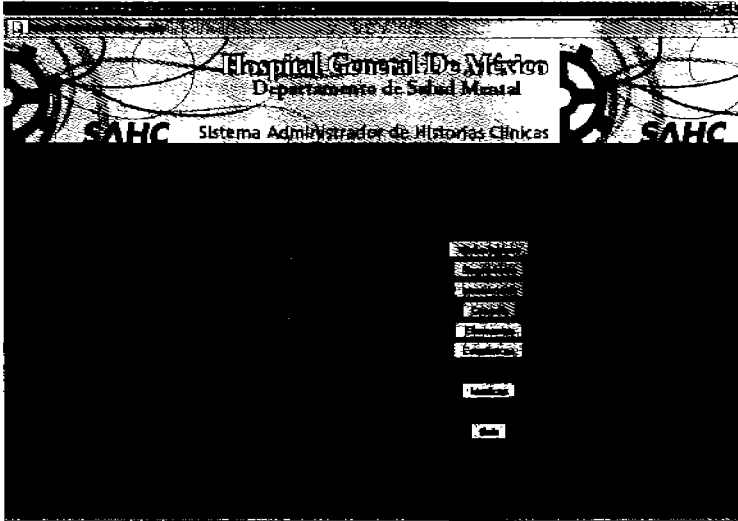


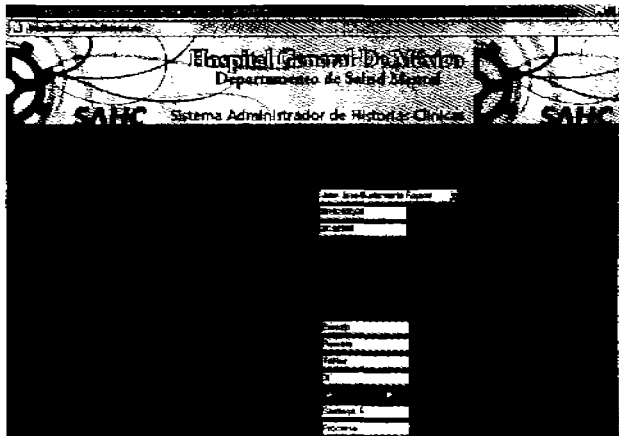
Figura. 3 Menú principal

### Ingreso de una nueva Historia Clínica

Se oprime el botón de Nuevo Ingreso y aparecerá la primer parte del formato de altas.

En esta parte se deberán ingresar los datos personales del paciente, la fecha correspondiente a ese día se mostrará en pantalla, no es necesario teclearla.

4a



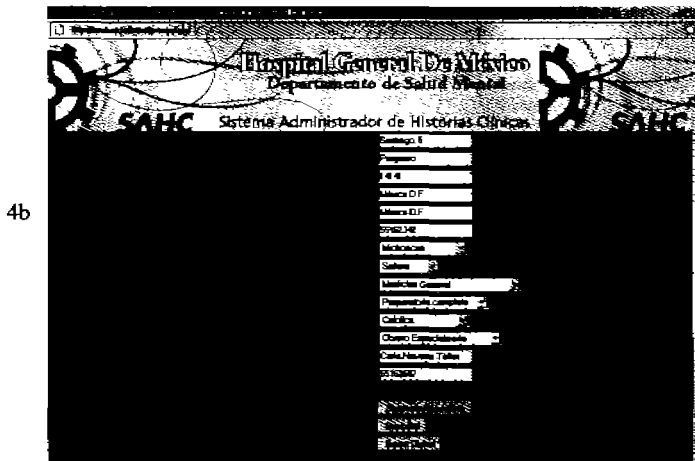
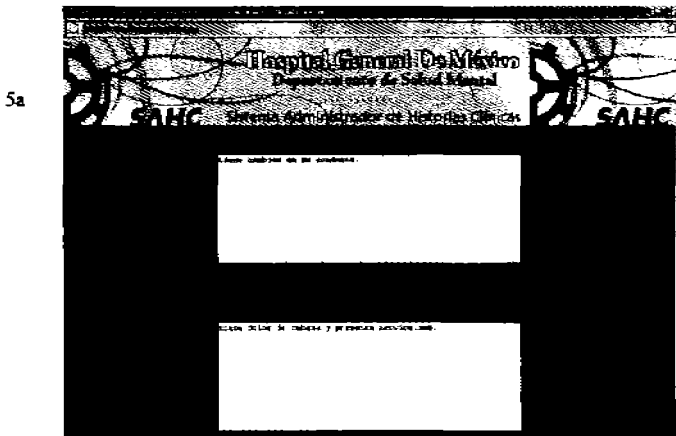


Figura. 4 Primera pantalla de Alta de pacientes (a) y (b)

Para pasar a la siguiente pantalla se oprime el botón de Siguiente Formulario. Si el número de expediente ya existe se le pedirá que lo cambie en la primer pantalla.

Si alguno de los campos de la primera pantalla se encuentra vacío aparecerá un mensaje que avisará que campo falta por escribir y se ubicará el cursor en dicho campo. Esto sucederá hasta que todos los campos se hayan llenado. En el campo Edad se le pedirá que ingrese un número entero, es decir no se admiten comas ni puntos.

En cualquier momento del alta del paciente se puede cancelar el proceso y también se pueden limpiar todos los campos en caso de que haya necesidad de volver a escribir o corregir alguna información del paciente.



En la segunda pantalla se escribe el motivo de la consulta y el padecimiento actual. También se seleccionan con un clic los antecedentes psiquiátricos y no psiquiátricos personales y familiares que presente el paciente.

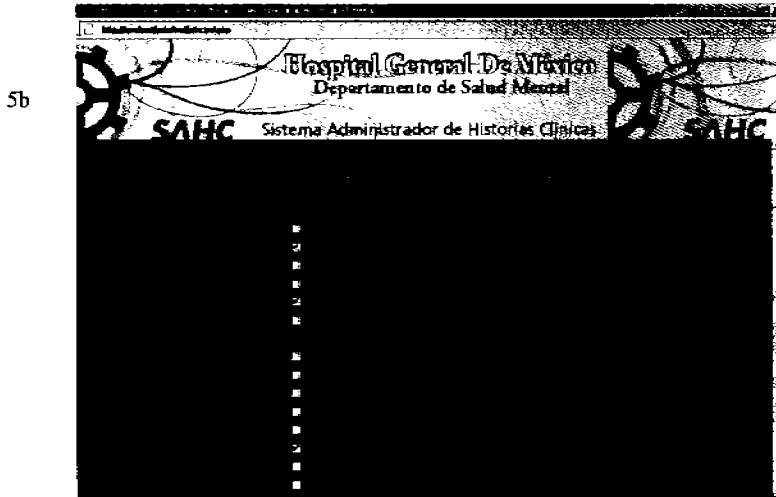


Figura. 5 Segunda pantalla de altas (a) y (b)

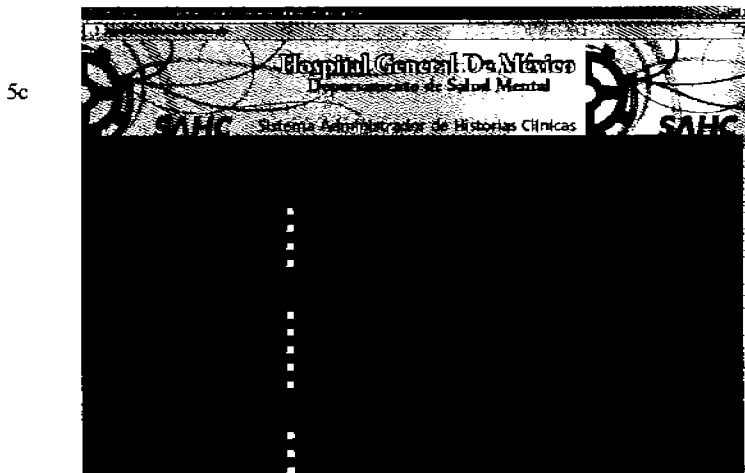


Figura 5c. Segunda pantalla. Antecedentes Fam. no psiquiátricos.

En la siguiente parte del formulario se debe llenar la parte de Violencia Familiar dando clic si se es receptor o agresor o ambas en todos los tipos de violencia. También si es el caso se selecciona el tiempo si es actual, anterior o ambas.

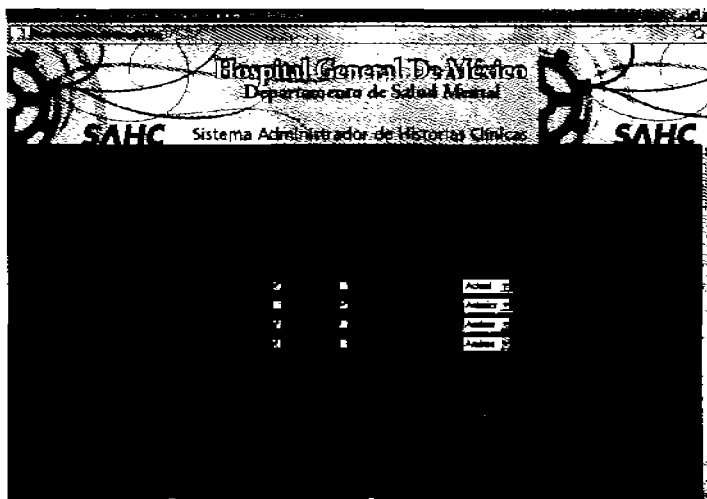


Figura 6. Tercera parte. Violencia Familiar.

Después de la violencia familiar Llenar la escala Missouri dando clic en las opciones donde se presente alguna anomalía.

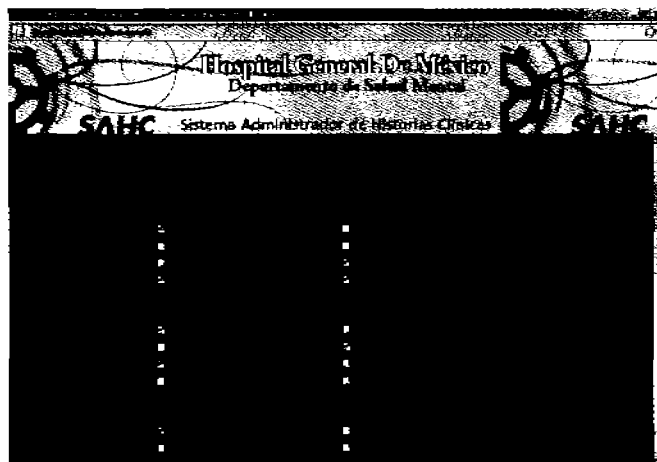


Fig.7a. Escala Missouri



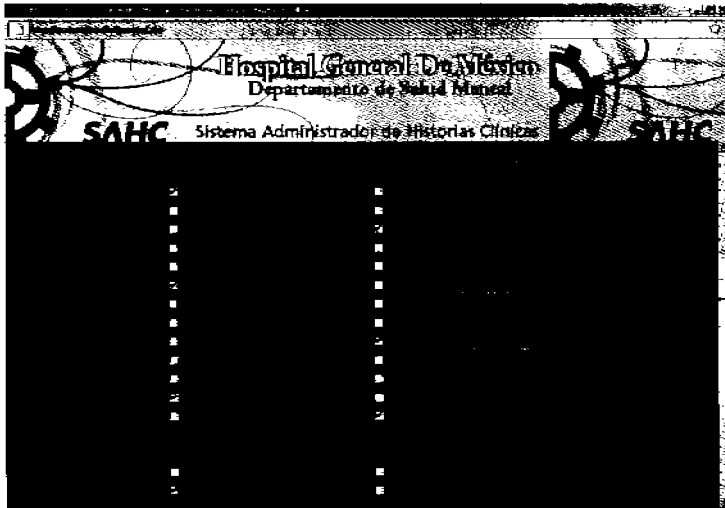


Figura 7b. Continuación de la Escala Missouri.

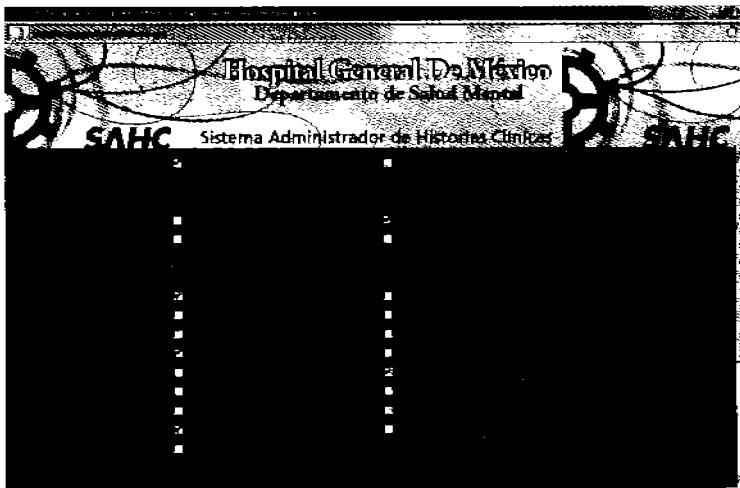


Figura 7d. Parte final de la Escala Missouri

Después de la escala Missouri se escribe lo observado en el examen mental y se toman los signos vitales del paciente para escribirlos en el formulario:

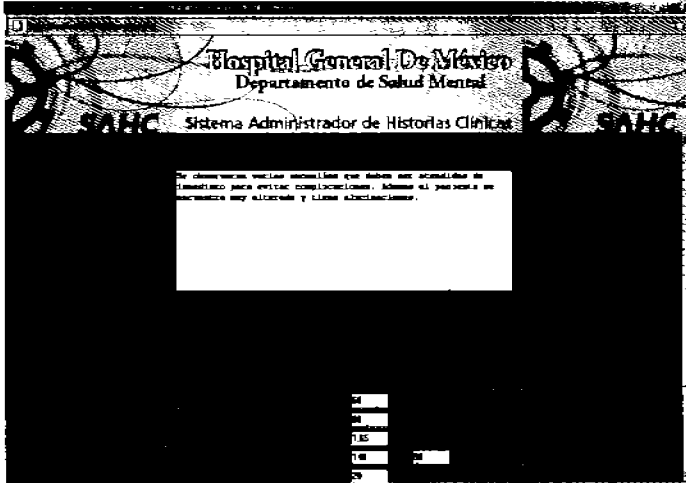


Figura 8. Examen Mental y Signos Vitales.

Al igual que en la primer pantalla si algún campo falta por llenar se le indicara al usuario mediante un mensaje. Excepto para la escala Missouri, ya que ahí pueden o no presentarse anomalías. Por ultimo se deben llenar los campos de exploración física y datos adicionales (este ultimo es opcional por si el medico necesita escribir alguna otra información importante).

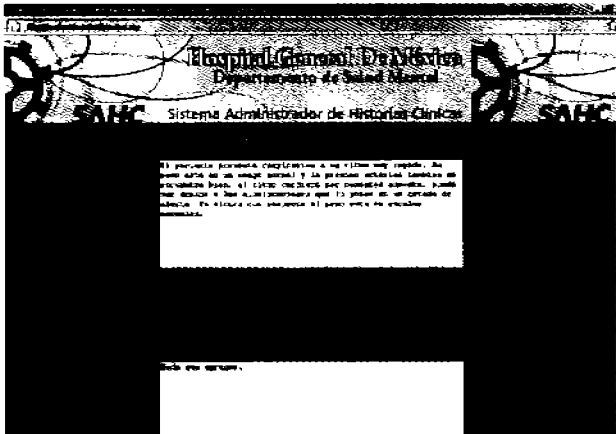


Figura 9. resultados de la Exploración Físicas y Datos Adicionales.

En la cuarta y última parte del formulario de altas se seleccionan él o los diagnósticos en la escala presentada en el menú desplegable.

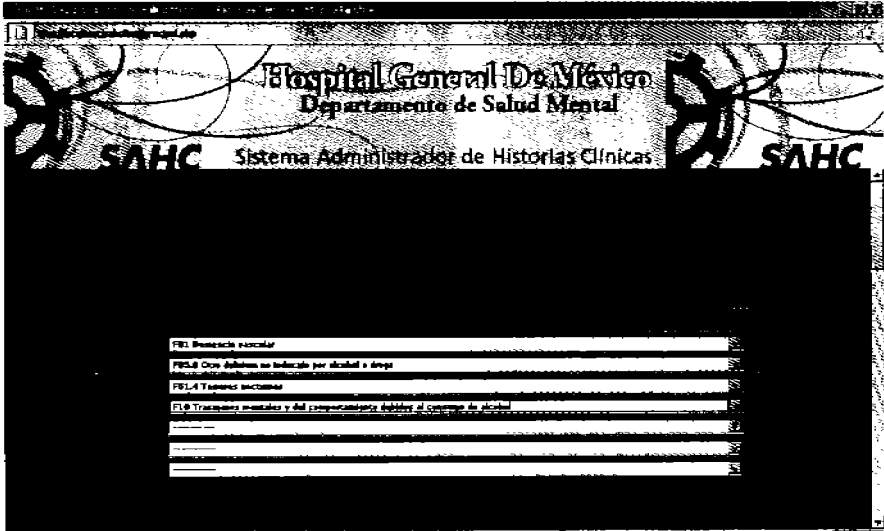


Figura 10. Menú de Diagnósticos

Después se seleccionan el o los psicofármacos que se recetarán junto con la dosis por día escrita con número así como también el número total de medicamentos recetados.

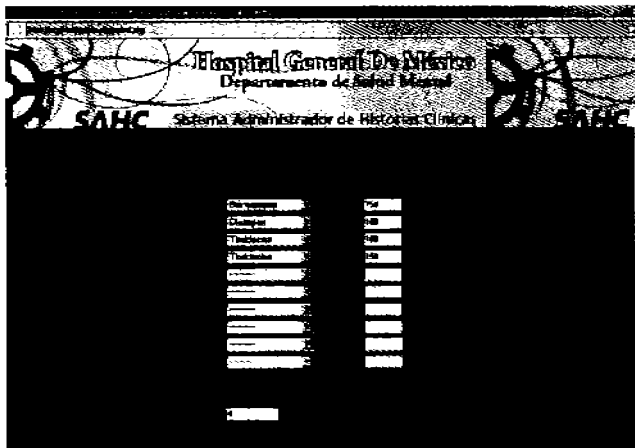


Figura 11. Psicofármacos, dosis/día y número de medicamentos.

En seguida de los diagnósticos se selecciona el plan terapéutico a seguir en el que se elige el tipo de terapia, aquí también pueden seleccionarse varias.

Por último viene el campo de referencia para indicar si el paciente debe ir a otra área o inclusive a otro hospital para realizar análisis clínicos.

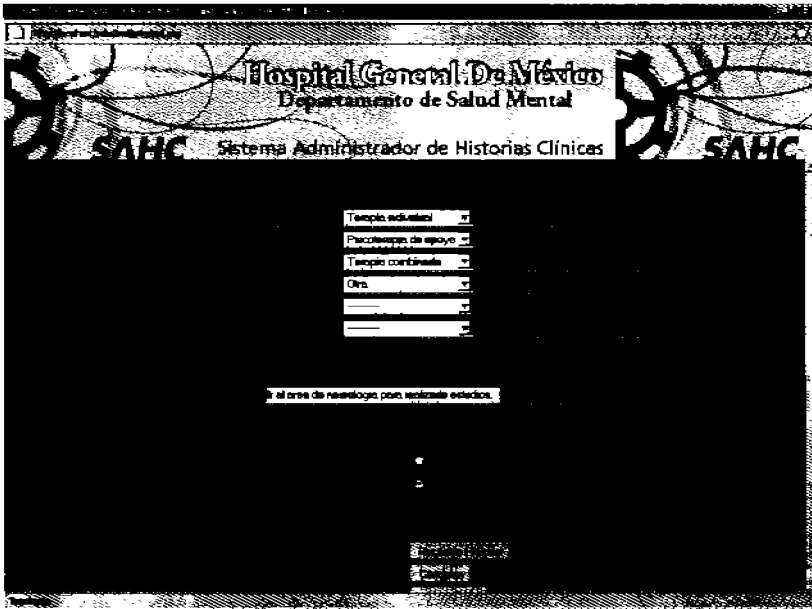


Figura 12. Plan Terapéutico y Referencia.

Como se puede observar en la figura 12 también se tienen dos opciones, una de alta clínica y otra de seguimiento que indica que el paciente tendrá que regresar a una segunda revisión, por lo que seguirá en observación. La de alta se elige cuando ya no es necesario que el paciente regrese a una nueva consulta.

Al final se presiona el botón Registrar historia para guardar toda la información del paciente en la base de datos del sistema y aparecerá la siguiente pantalla.

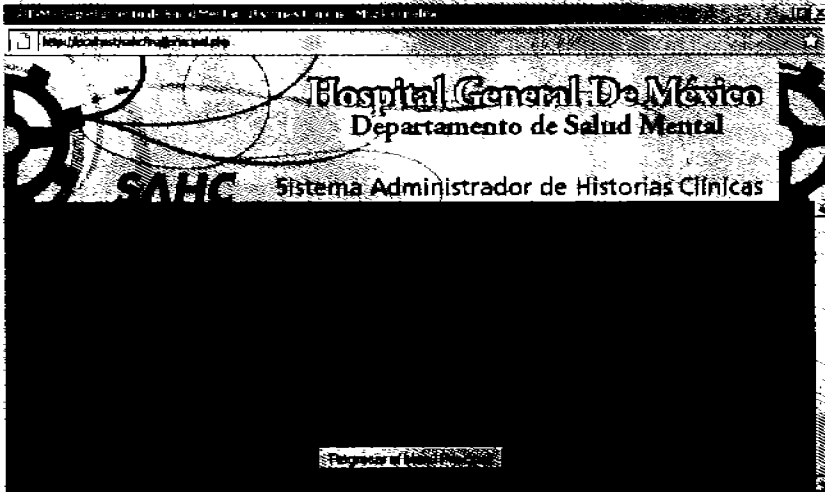


Figura 13. Información del almacenamiento de la información y situación del paciente.

## Seguimiento de Pacientes

Una vez que se tienen registrados los datos de consulta de primera vez de un paciente se le puede realizar el seguimiento (si no ha sido dado de alta clínica) lo primero q se necesita es encontrar al paciente en la base de datos, así que al oprimir el botón de seguimiento en el menú principal, se desplegará una pantalla para escribir algún dato del paciente que pueden ser el numero de expediente, el expediente interno, el nombre, el apellido paterno y el apellido materno. Se debe escribir sólo uno de los 4 datos y después dar clic en el botón Aceptar para comenzar la búsqueda del paciente.

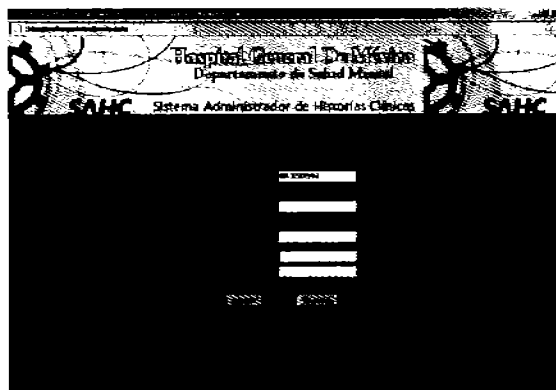


Figura 14. Pantalla de búsqueda de pacientes para el seguimiento.

Aquí pueden suceder 2 cosas, la primera que no exista el paciente o se haya escrito mal algún dato en la base. Cabe mencionar que los datos son sensibles a mayúsculas y minúsculas. Si se diera este primer caso se muestra el mensaje de error y se regresa a la pantalla de búsqueda.

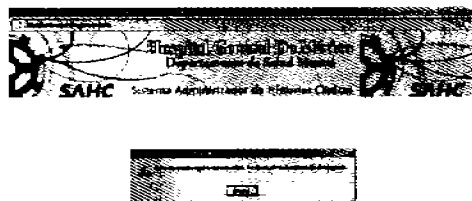


Figura 15. Mensaje de error en la búsqueda de un paciente.

Si la búsqueda es exitosa se mostrará el nombre completo del paciente junto con los números de expediente general e interno así como también la fecha de su visita más reciente.

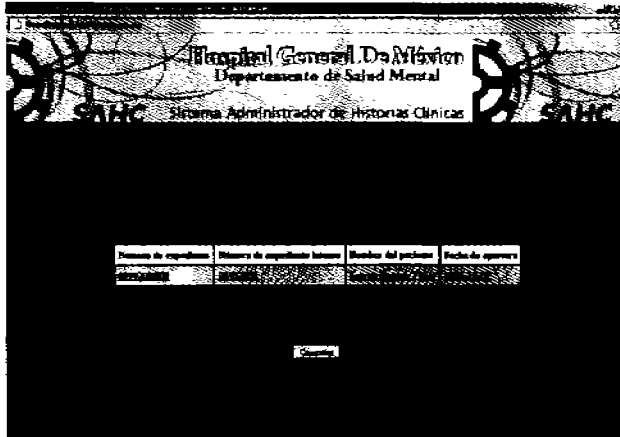


Figura 16. Resultado de la búsqueda de un paciente.

Se puede dar el caso que la búsqueda arroje varios resultados, por ejemplo cuando varios pacientes tiene el mismo apellido, para diferenciarlos se utiliza el numero de expediente que es único para cada uno de ellos y también la fecha puede servir como una referencia y por supuesto el nombre completo.

A continuación se da clic sobre el nombre del paciente buscado para comenzar a llenar el formato de seguimiento.

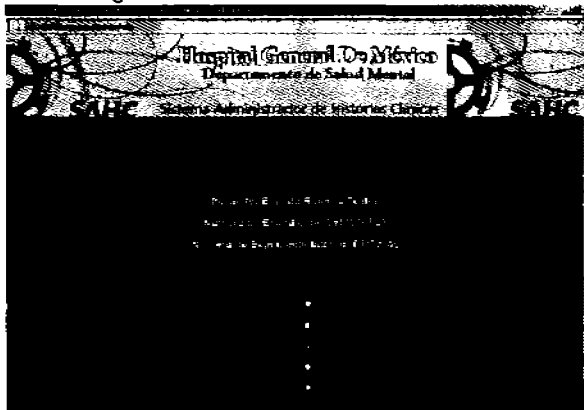


Figura 17. Seguimiento de Pacientes.

Se muestra la información del paciente (nombre, expedientes general e interno) para tener presente siempre sus datos.

Después viene la fecha en la que se está realizando el seguimiento, no es necesario teclearla ya que se desplegará la correspondiente a ese día. Se elige la evolución del paciente entre las 5 opciones. A continuación se tomarán los signos vitales. Se mostrarán desplegados los signos tomados en la visita inmediata anterior para tenerlas como referencia, esta información adicional es muy útil para el medico, y dicha ventaja también se tiene en el campo de diagnósticos, psicofármacos, terapias y la escala Missouri.

En los campos de evolución, SOAP y efectos secundarios se desplegará la información de la visita anterior a partir de la tercera vez que el paciente venga a consulta.

The screenshot shows the SAHC (Sistema Administrador de Historias Clínicas) interface. At the top, it displays 'Hospital General de México' and 'Departamento de Salud Mental'. Below this, there are two 'SAHC' logos. The main area contains a form with several input fields. On the left, there are five vertical fields labeled 'TA', 'FC', 'TA', 'FC', and 'C'. To the right of these fields, there is a dropdown menu with a small arrow icon. Below the form, there is a large white rectangular area with the text 'antes de escribir se deben de desplegar.' above it.

Figura 18. Signos Vitales y Subjetivo.

En esta parte se comienza a aplicar el sistema de seguimiento SOAP, Subjetivo, Hallazgos del Examen Mental y Objetivo y Análisis.

The screenshot shows the SAHC interface for entering SOAP notes. It features the same header as Figure 18. The main area contains two large white rectangular text boxes. The top box has the text 'antes de escribir se debe de desplegar.' above it. The bottom box has the text 'antes de escribir se debe de desplegar.' above it.



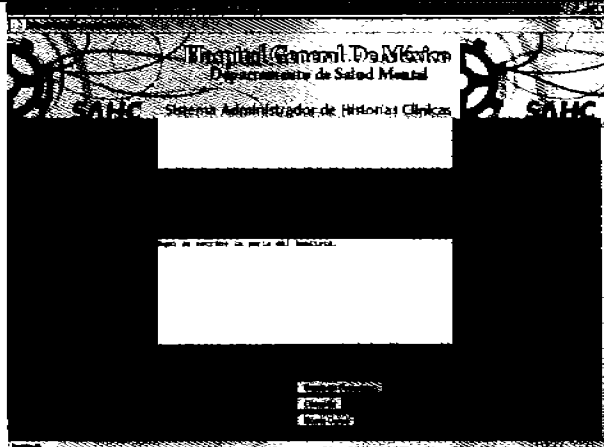


Figura 19. Subjetivo, hallazgos del examen mental, objetivo y Análisis.

En la siguiente pantalla se aplica al paciente la escala Missouri, se muestran seleccionadas las anomalías vistas en la visita anterior para que revise si se siguen presentando o desactivarlas en caso de que ya no se manifiesten.

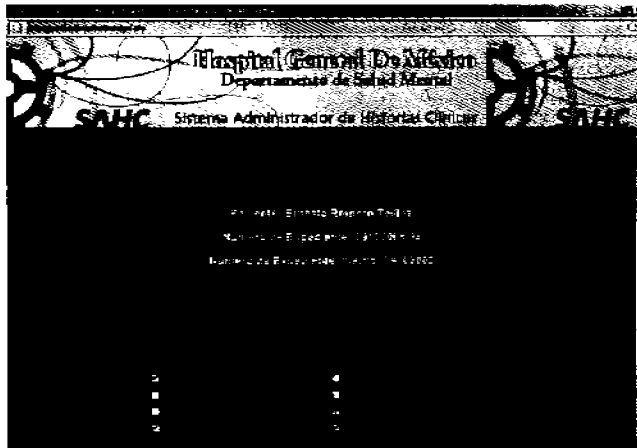


Figura 20. Escala Missouri.

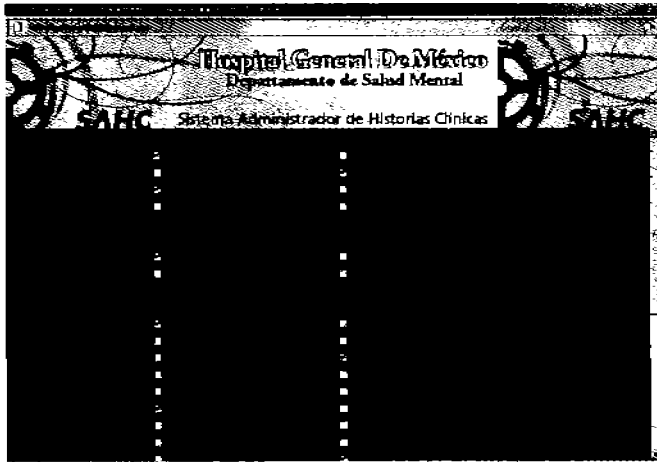


Figura 21 Continuación de la Escala Missouri

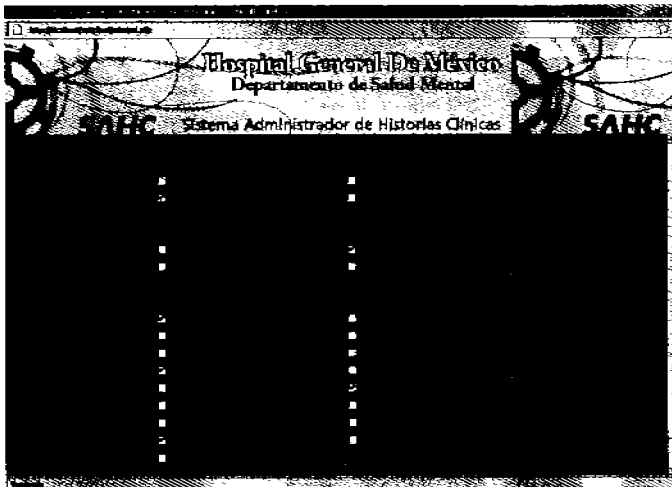


Figura 22. Parte final de la escala Missouri

En la siguiente pantalla se deben especificar los efectos secundarios causados por algún psicofármaco recetado. Para realizar esto se da clic en el efecto que se presente y a continuación se selecciona la intensidad en que se presentó, puede ser leve, moderado. Severo o muy severo.

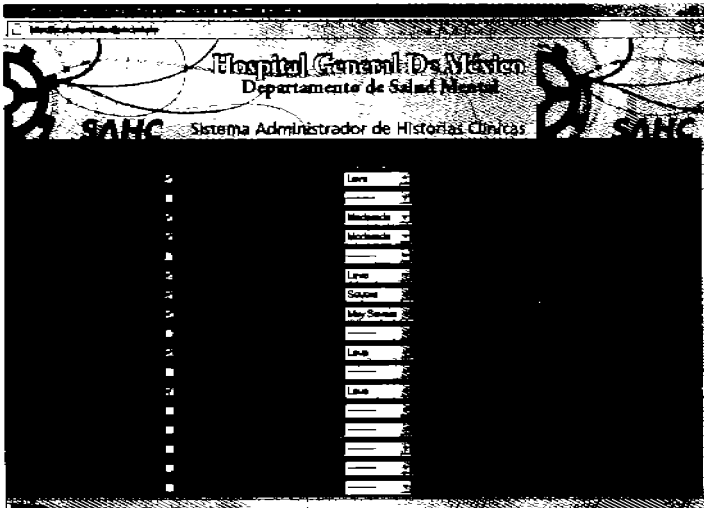


Figura 23. Tabla de Efectos secundarios.

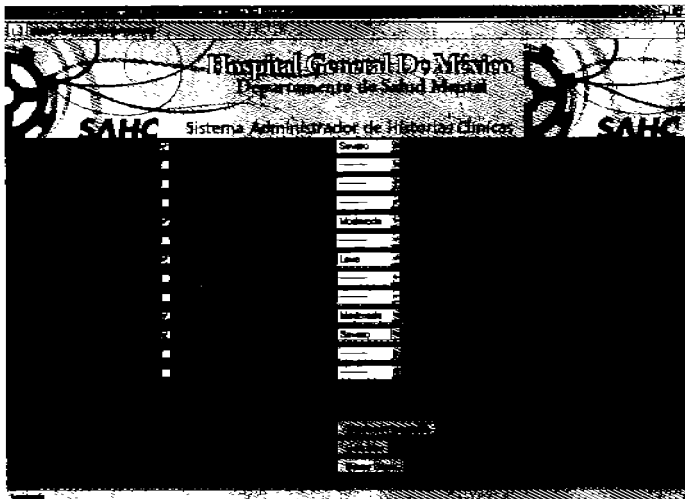


Figura 24. Continuación de la tabla de efectos secundarios.

En la siguiente pantalla se seleccionan él o los diagnósticos. Se muestran los de la visita anterior como referencia. Para agregar un diagnóstico se selecciona de la lista desplegable. Si se desea eliminar alguno se debe seleccionar la línea que está al principio del menú desplegable.

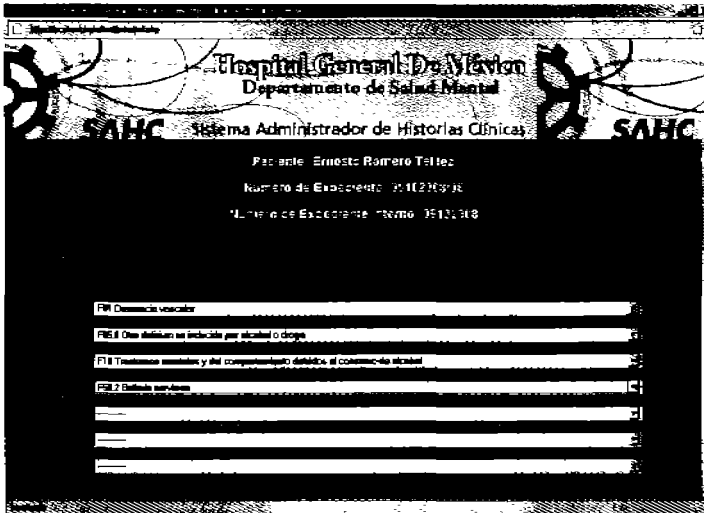


Fig. 25 Tabla de Diagnósticos.

Después se escribe el Plan (última parte del SOAP) en el campo de texto especificado.

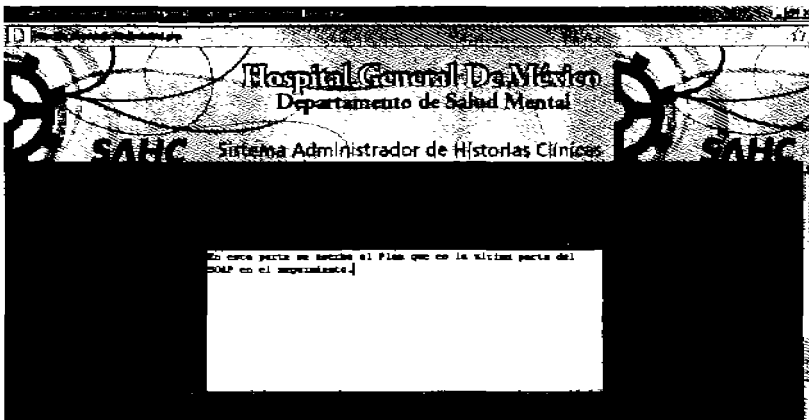


Figura 26. Plan



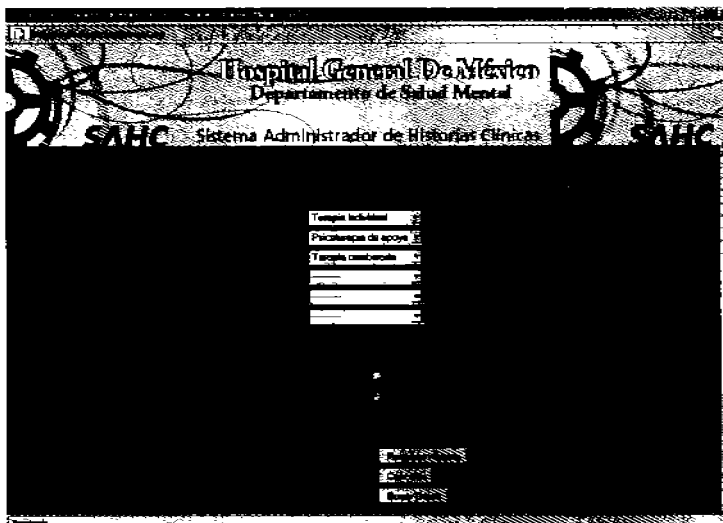


Figura 28. Plan Terapéutico a seguir y situación del Paciente.

Aparecerá enseguida una pantalla con los datos del paciente indicando su situación clínica y con esto se concluye el seguimiento del paciente.

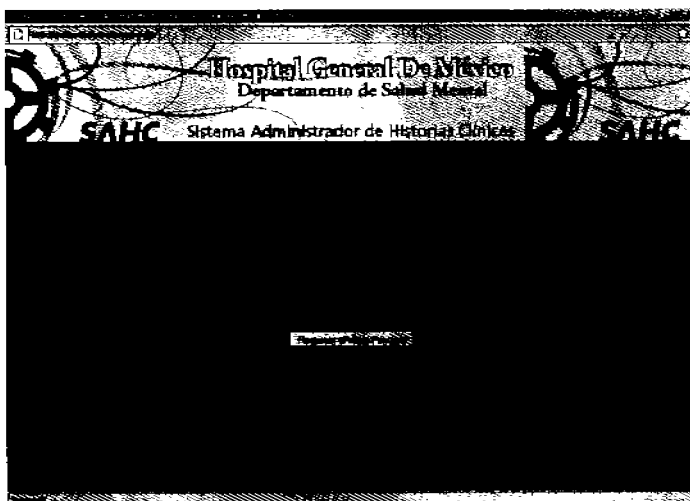


Figura 29. Información del paciente y situación clínica.

### Interconsulta

Para realizar una interconsulta, en el menú principal dar clic en el botón correspondiente. Aparecerá una pantalla para escribir los datos de identificación del paciente y del medico del servicio de origen.

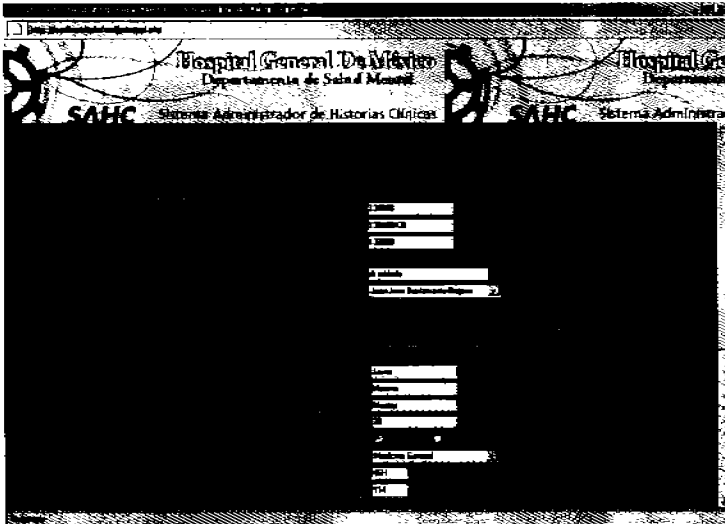


Figura 30. Formulario de Interconsulta.

A continuación se escribe el motivo de la consulta en el campo de texto correspondiente.

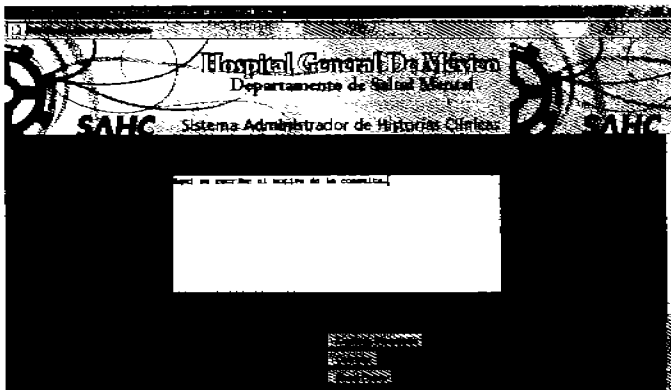


Figura 31. Campo de texto del motivo de la consulta

Después se le aplica al paciente la escala Missouri dando clic en donde se presente alguna anomalía.

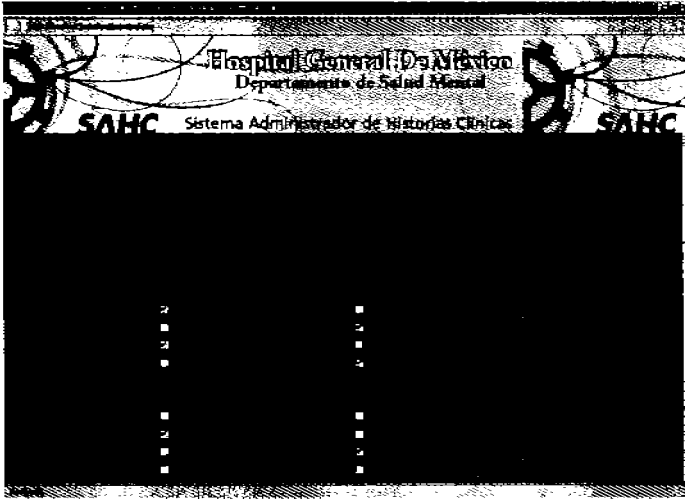


Figura 32 Escala Missouri para la Interconsulta

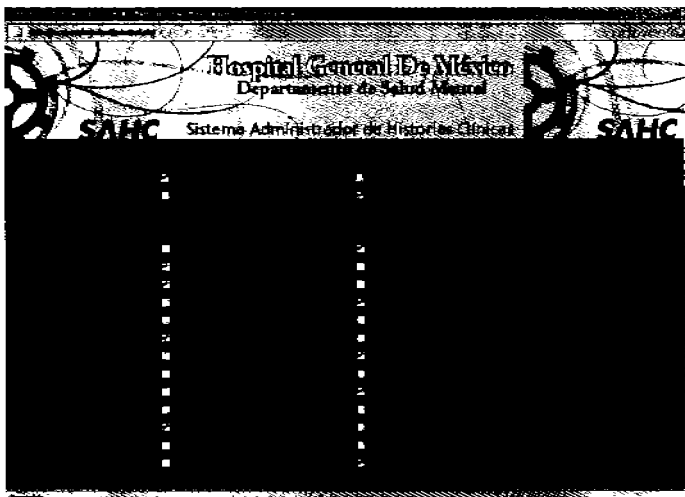


Figura 33. Continuación de la Escala Missouri



Después de aplicar la escala Missouri se elige el diagnóstico o diagnósticos de la ICD-10 (International Statistical Classification of Diseases and Related Health Problems 10th Revision) en el menú desplegable.

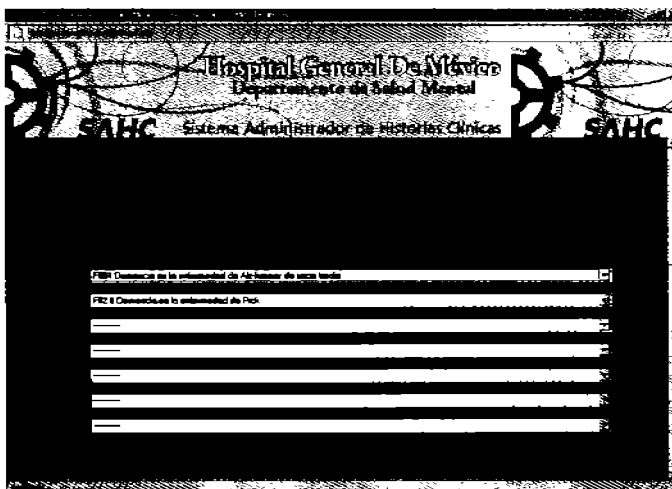


Figura 34. Menú desplegable de Diagnósticos de la ICD-10

Al terminar con los diagnósticos se elige él o los psicofármacos y se escribe con número la dosis/día en el campo indicado para ello.

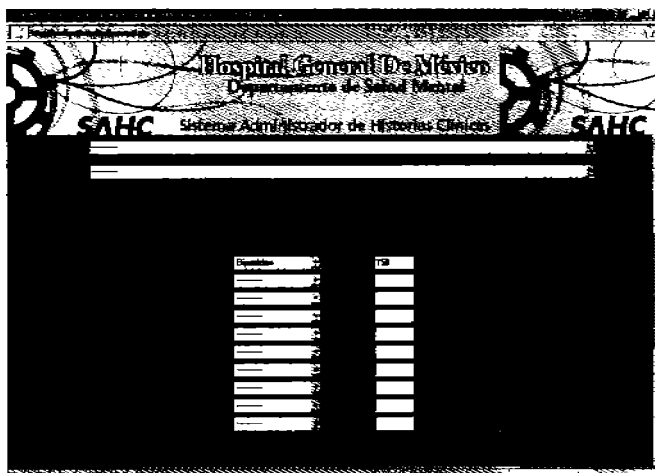


Figura 35. Tabla de Psicofármacos para Interconsulta

Como parte final de la Interconsulta se elige el plan terapéutico a seguir y la situación clínica del paciente de entre las opciones mostradas.

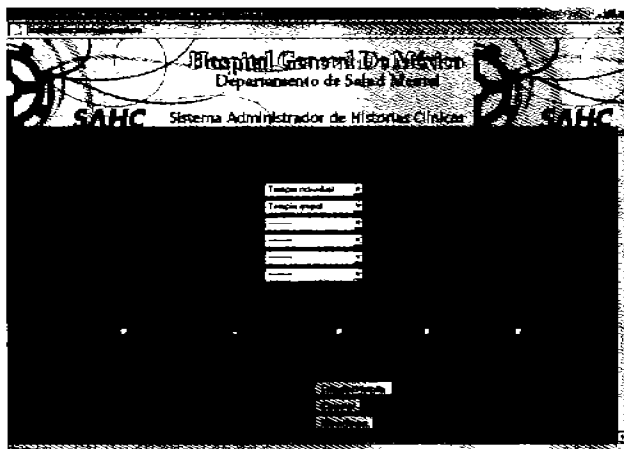


Figura 36. Plan Terapéutico a seguir y opciones de la situación clínica del paciente

En este caso se tienen varias opciones de la situación clínica:

**Alta clínica:** Para cuando el paciente ha sido de alta del depto. de salud mental

**Seguimiento:** El paciente regresa a una nueva consulta.

**Extraviado:** Si después de la primera interconsulta no se sabe nada del paciente.

**Ausente:** Si no regresa a una nueva consulta.

**Se ignora:** No se sabe la situación clínica del paciente.

Después de llenar el formulario se da clic en el botón Registrar Historia para guardar toda la información de la Interconsulta del paciente y se mostrará la pantalla final.

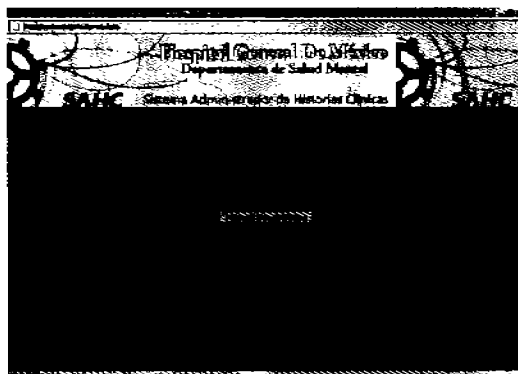


Figura 37. Pantalla con la información del paciente y su situación clínica.

### Consultar una Historia Clínica.

El sistema permite a los médicos revisar los datos guardados de un paciente. Para ello se tiene el botón correspondiente en el menú principal.

Al dar clic en éste botón se mostrará una pantalla similar a la de seguimiento en donde se pedirá escribir alguno de los campos solicitados (expediente, expediente interno, nombre, apellido paterno, o apellido materno) y a continuación se da clic en el botón Aceptar para comenzar la búsqueda del paciente en la base de datos del sistema.

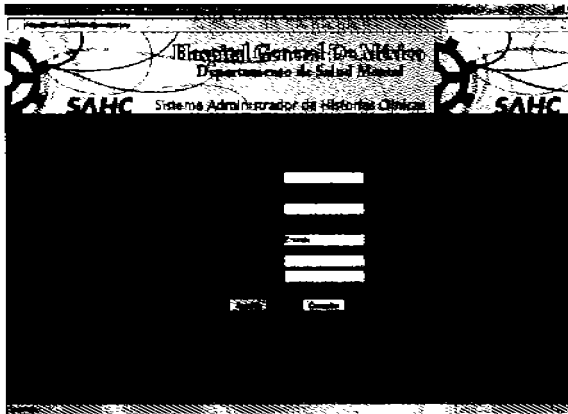
The screenshot shows a web browser window with the title "Hospital General de México Departamento de Salud Mental". The page header includes the SAHC logo and the text "Sistema Administrador de Historias Clínicas". The main content area contains a search form with four input fields stacked vertically, labeled "Expediente", "Expediente interno", "Nombre", and "Apellido". Below the fields are two buttons: "Aceptar" and "Cancelar".

Figura 38. Pantalla para escribir los datos del paciente a encontrar

Se mostrarán los resultados de la búsqueda, es importante mencionar que si no se encuentra ningún paciente con esos datos se le pedirá que corrija la información de búsqueda o también puede darse el caso que se esté buscando a un paciente que no exista en la base de datos.

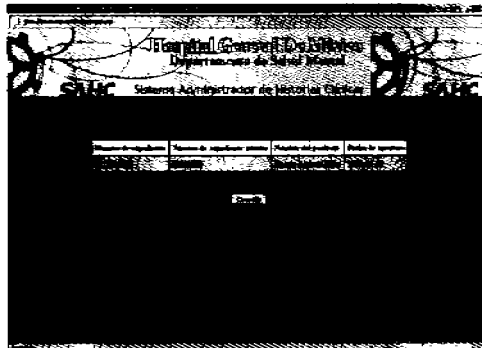
The screenshot shows the search results page in the SAHC system. The header is identical to Figure 38. Below the header, there is a table with four columns: "Expediente", "Expediente interno", "Nombre", and "Apellido". The table is currently empty. Below the table is a "Aceptar" button.

Figura 39. Resultados de la búsqueda.

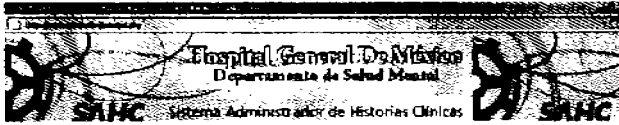


Figura 40. Mensaje de error en la búsqueda de un paciente.

Al encontrar al paciente solicitado se le da clic en su nombre para poder acceder a la siguiente pantalla donde se mostrará el nombre y los expedientes general e interno, el número de veces que ha ido a consulta y su situación clínica.

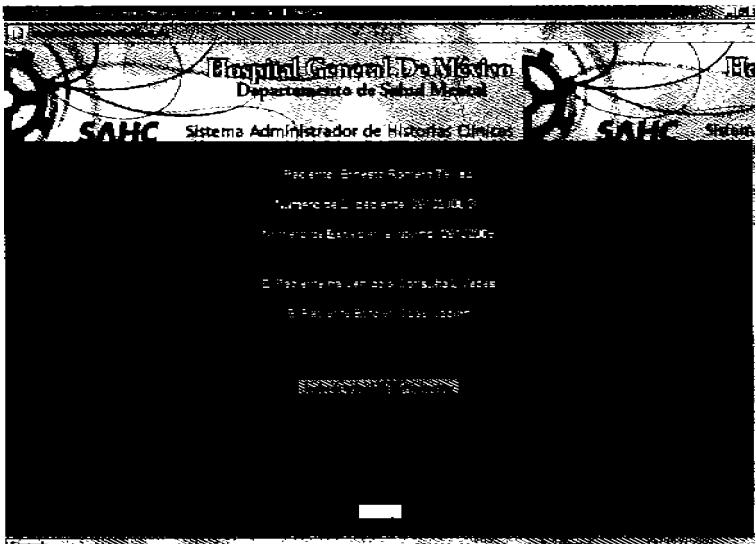


Figura 41. Información general del paciente

Se tiene un botón en donde se pueden revisar los datos de consulta de primera vez.

Para la segunda visita en adelante se debe teclear el número de visita (con dígitos) en el campo indicado.

a) Datos de consulta de primera vez.

Aquí se mostrará toda la información referente al formato de historia clínica breve, se divide en cuatro secciones.

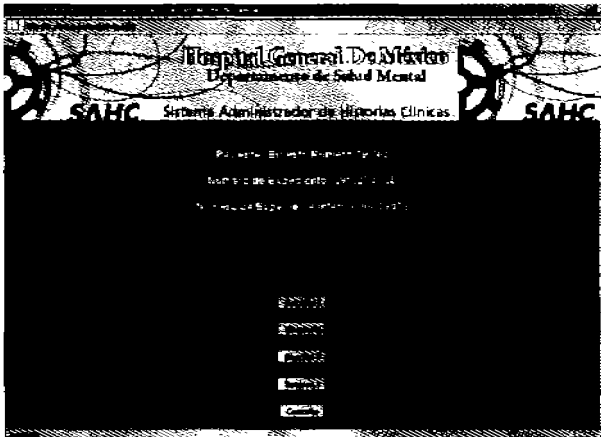


Figura 42. Submenú de los datos de consulta de primera vez.

Como muestra se revisará la información de la Sección 3 que se refiere a los datos de violencia, escala Missouri y signos vitales.

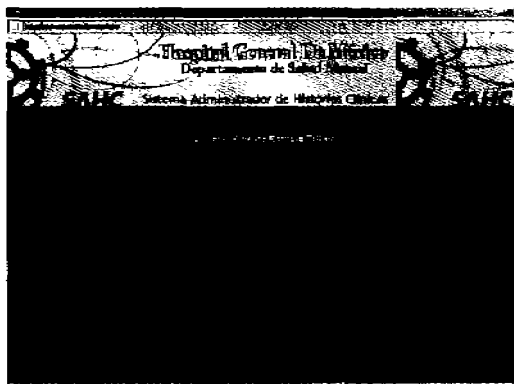


Figura 43. Datos de consulta de primera vez del paciente. Sección 3.

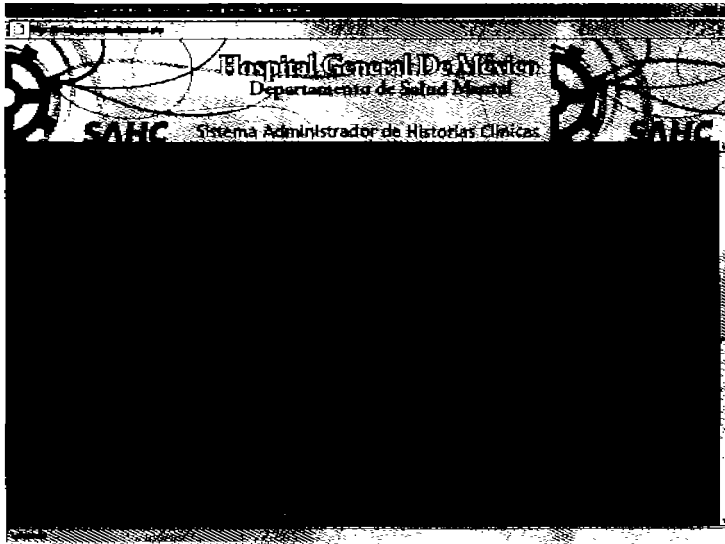


Figura 44. Continuación de la Sección 3.

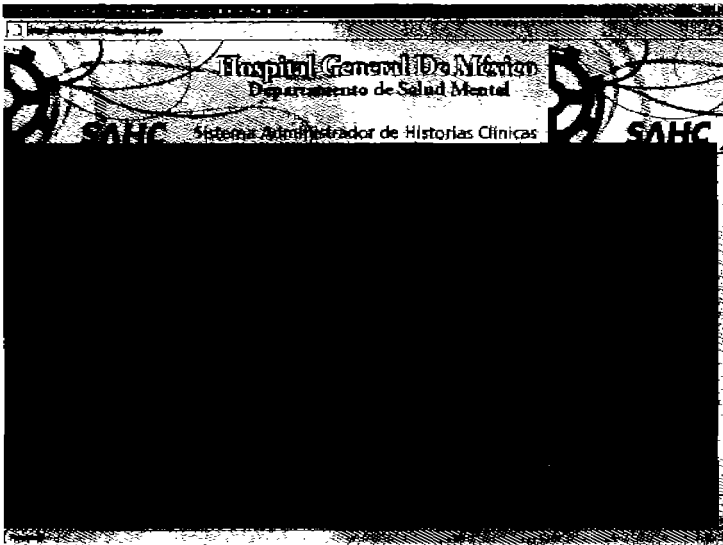


Figura 45. Parte Final de la Sección 3

b) Datos de consultas de seguimiento.

En esta parte se debe teclear el número de visita que se desea consultar a partir de la segunda y menor o igual a la indicada en la información del paciente. Si se teclea un número mayor o menor o igual a 1 se mostrará un mensaje de error y se tendrá que corregir el número solicitado.

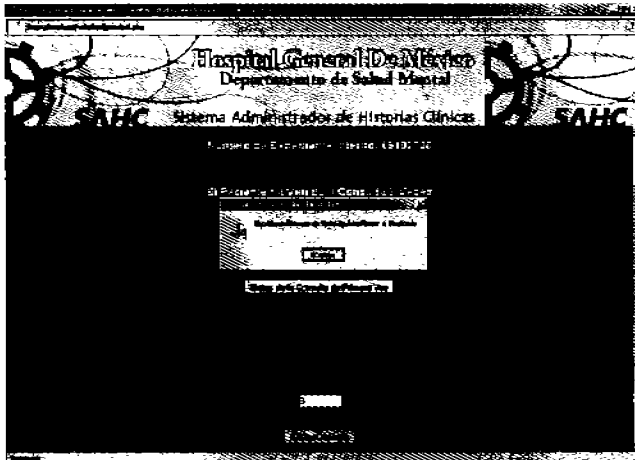


Figura 46. Mensaje de error al solicitar un número de visita incorrecto.

Si el número de visita está dentro del rango mostrado se pasará al submenú en donde están las secciones de los datos de seguimiento.

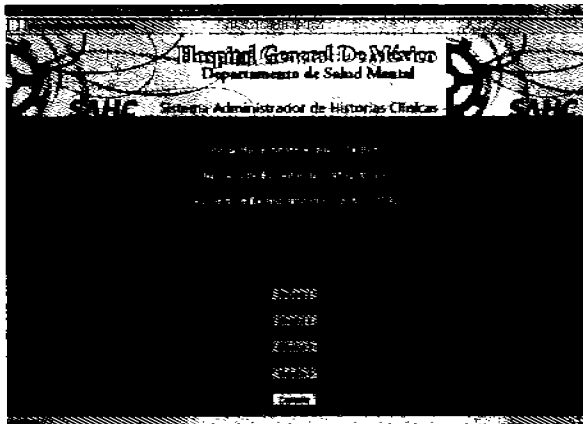


Figura 47. Secciones de los datos de seguimiento del paciente.

Como muestra se desplegarán los datos de la sección 3 que consiste en los efectos secundarios que sufrió el paciente, correspondientes a la segunda visita.

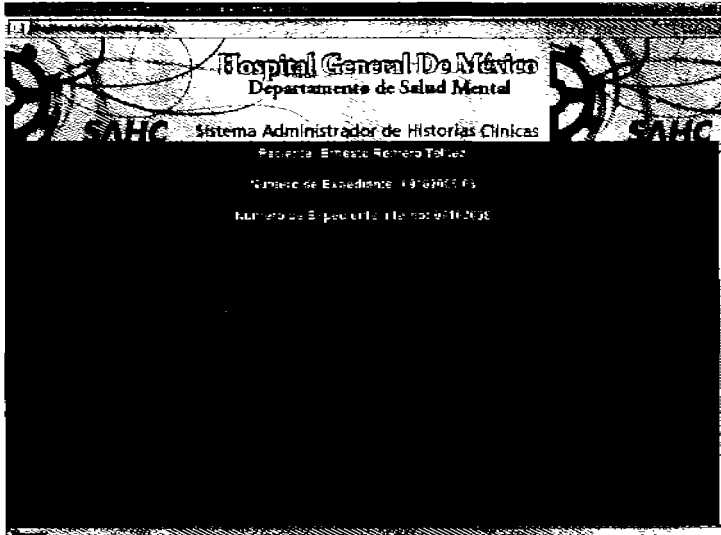


Figura 48. Información de seguimiento del paciente. Sección 3.

Todas las secciones que contiene la información del paciente se pueden imprimir, para ello se debe dar clic en el botón correspondiente y se abrirá la ventana de impresión en donde se podrán configurar las propiedades de la impresora que se tenga conectada y realizar las impresiones que se necesiten.



Figura 49. Propiedades de impresión de la información del Paciente.



### Eliminación de una Historia Clínica.

Cuando el paciente ya fue dado de alta y no se necesita su información puede borrarse de la base de datos, para realizar esto en el menú principal se selecciona la opción de Borrar una Historia Clínica, enseguida se mostrará en pantalla un campo para escribir el número de expediente general del paciente.

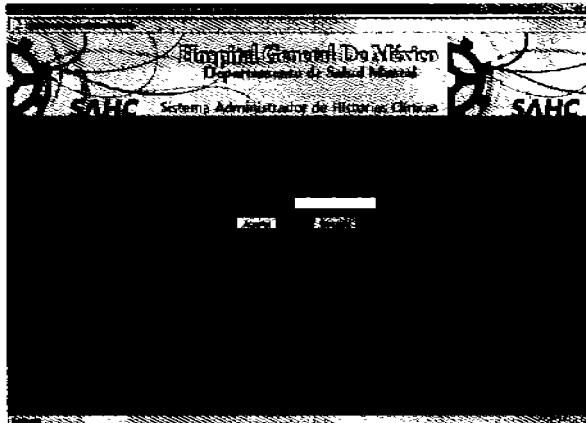


Figura 50. Pantalla para escribir el número de expediente a eliminar.

Si el número de expediente es correcto se desplegará la siguiente pantalla con el nombre completo del paciente, expediente general, interno y su fecha de apertura.

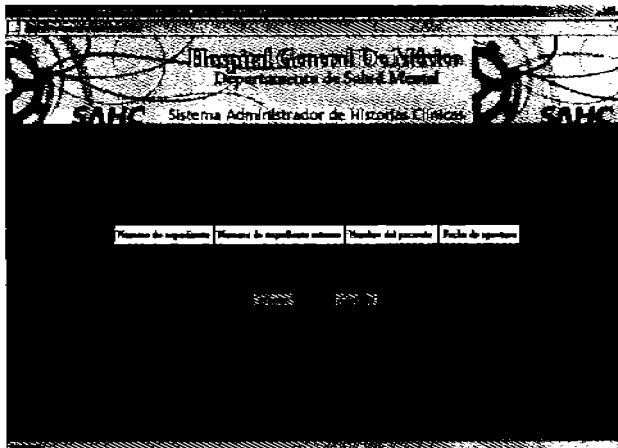


Figura 51. Datos del paciente que se eliminará

Al dar clic en el botón aceptar se borrará toda la información del paciente incluyendo todas sus visitas y la información no se podrá recuperar y se mostrará la siguiente pantalla.

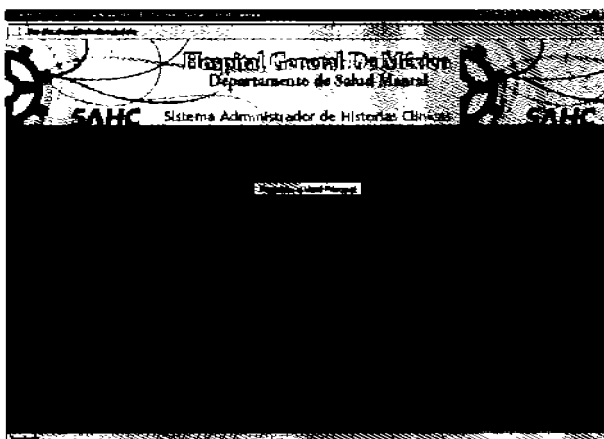


Figura 52. Parte final de la eliminación del paciente.

## Estadísticas.

El sistema contiene un conjunto de estadísticas para que los médicos puedan revisar todo lo concerniente a la actividad diaria del hospital. Se tienen estadísticas de varias categorías. Por ejemplo para ver la cantidad de pacientes registrados por día o el diagnóstico más común, entre otras.

Para acceder a esta información en el menú principal se le da clic al botón estadísticas y se desplegará un submenú con 2 opciones: Estadísticas Generales y Estadísticas Predeterminadas.

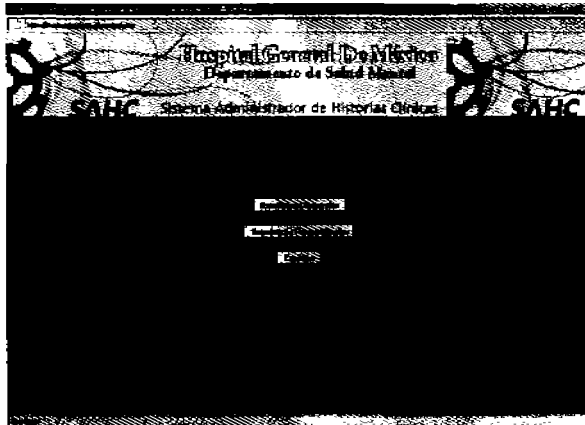


Figura 53. Submenú de Estadísticas.

### a) Estadísticas Generales

En esta parte se tienen estadísticas predeterminadas, referentes a los datos de pacientes, diagnósticos y demás características. Aquí se encuentran por ejemplo el número de pacientes registrado por día, mes y año. Pacientes por sexo, y otras estadísticas.

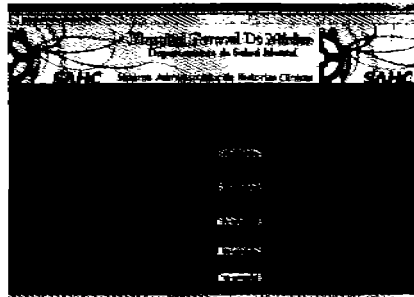


Figura 54. Submenú de Estadísticas del Departamento de Salud Mental

Para verlas se debe dar clic en el botón correspondiente y se desplegará la información en otra ventana. En la figura se muestra la estadística de pacientes registrados por día.

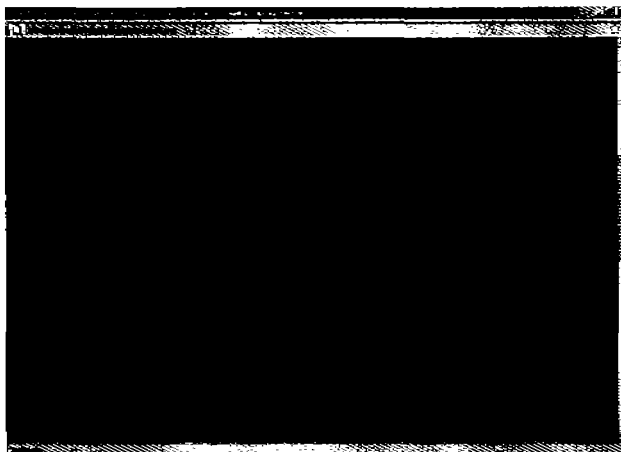


Figura 55. Pacientes Registrados por día

Las estadísticas pueden imprimirse, al final de cada una está el botón *Imprimir* y se abrirán las propiedad de la impresora que se tenga instalada y se procederá a comenzar la impresión.

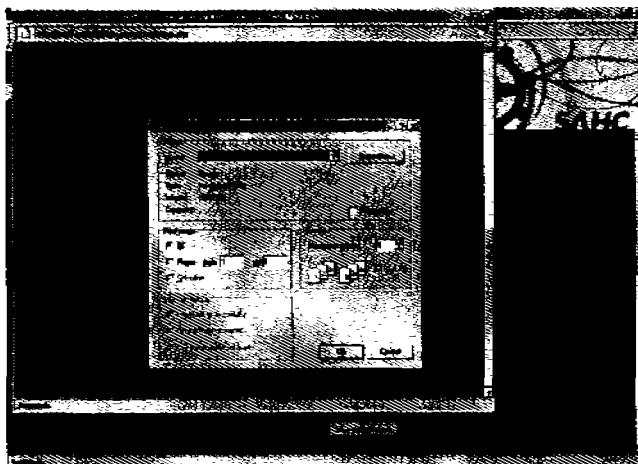


Figura 56. Pantalla de Impresión de Estadísticas.

## b) Estadísticas Personalizadas

Esta parte de las estadísticas es para que el administrador del sistema pueda ver desde la interfaz grafica la información en la base de datos sin tener que entrar directamente en ella. Aquí se tiene un campo para escribir sentencias en lenguaje SQL estándar comenzando con la palabra reservada **SELECT**, es por esto que es una parte que solo el administrador puede operar.

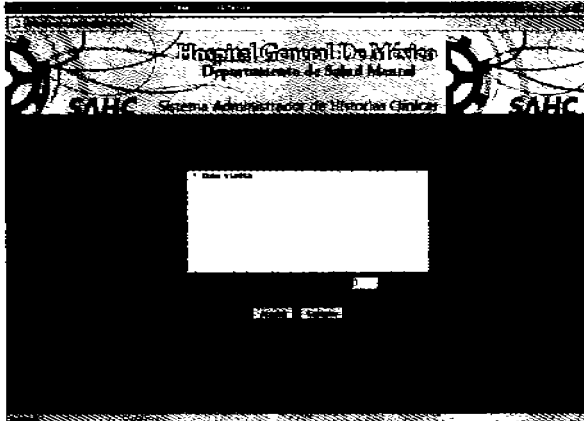


Figura 57. Estadísticas Personalizadas en lenguaje SQL

Después de escribir la sentencia SQL se teclea el número de columnas que se desea visualizar y se da clic en aceptar. Como ejemplo se hizo una consulta a la tabla visita y desplegará 3 columnas: idvisita, expediente y fecha. Al igual que en las estadísticas generales también se puede imprimir la información.

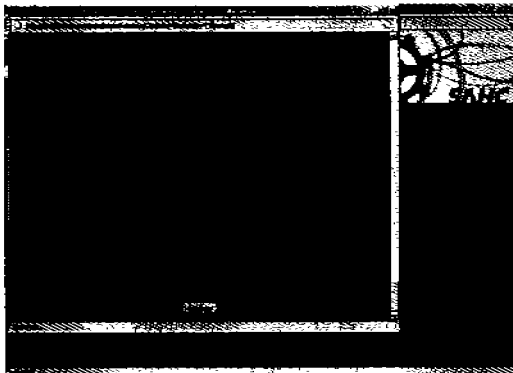


Figura 58. Resultado de una consulta personalizada

### Médicos.

En esta parte del menú principal los médicos pueden agregar y actualizar sus datos, es recomendable sobretodo cambiar su contraseña periódicamente por razones de seguridad así como también tener un correo electrónico en donde se les pueda mandar información acerca de cualquier cambio en el sistema.

Al dar clic en el botón Médicos se le pedirá que vuelva a escribir su nombre de usuario y su contraseña para comprobar su identidad.

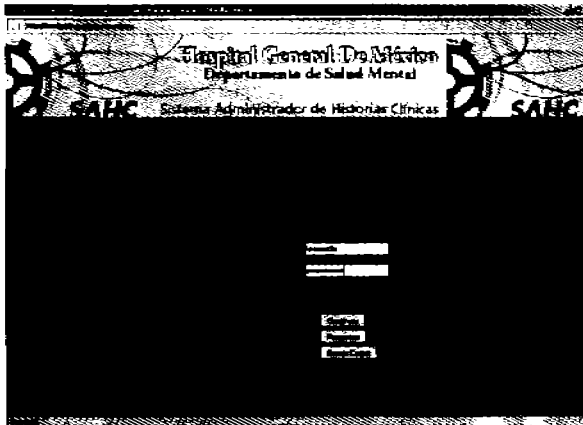


Figura 59. Autenticación de los Médicos.

Si los datos son incorrectos se le pedirá que escriba nuevamente su nombre de usuario y contraseña.

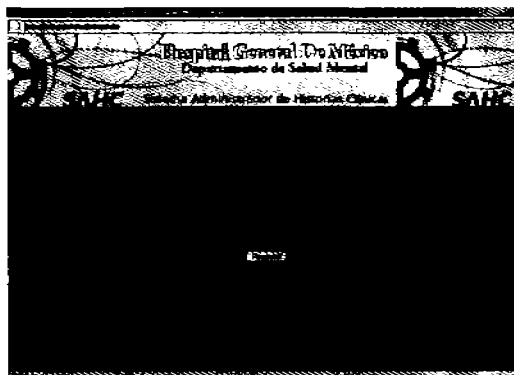


Figura 60. Mensaje de error cuando los datos son incorrectos.

Si los datos ingresados son correctos se mostrará la información actual del Médico. Si todo está bien y no hace falta agregar nada de clic en el botón regresar.

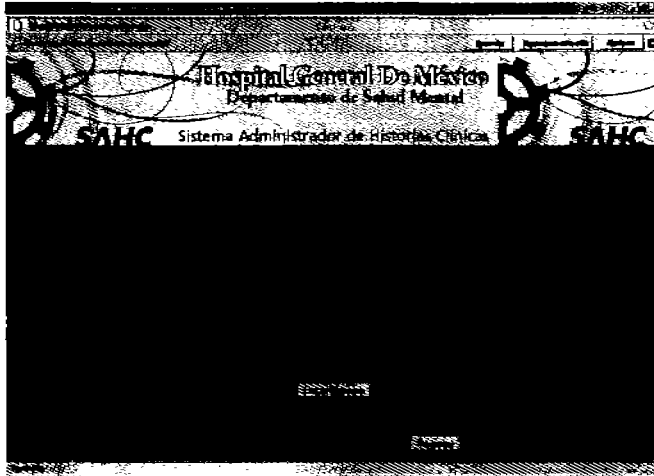


Figura 61. Datos actuales del Médico.

Si se desea modificar o agregar alguna información de clic en el botón *Hacer Cambios*. Aparecerá una pantalla con los datos actuales que podrán ser modificados.

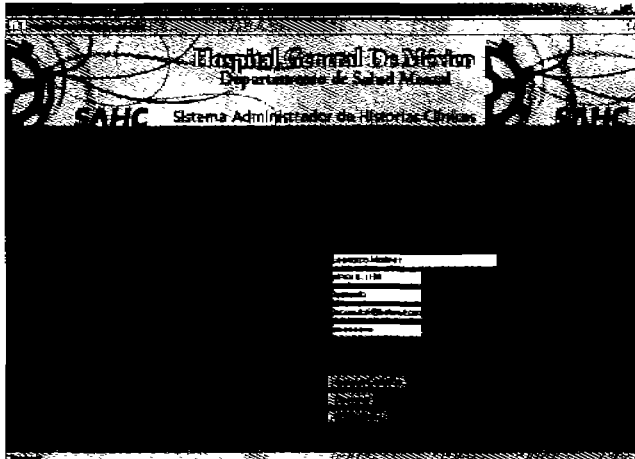


Figura 62. Modificación de los datos del Médico.

La contraseña aparece con puntos para proteger su privacidad. Al terminar de realizar todos los cambios de clic en el botón *Guardar Cambios*. Aparecerá una pantalla indicando que los cambios se guardaron satisfactoriamente.

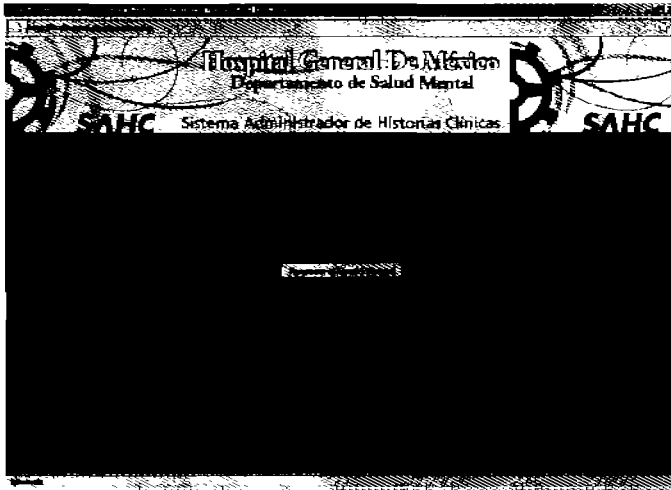


Figura 63. Pantalla de mensaje.



## Apéndice B. Funciones en Lenguaje PHP Utilizadas en el Sistema.

Conexión a la base de datos en Postgres:

```
<? php

    $user="postgres";
    $host = "localhost";
    $passwd = "postgres";
    $db = "base_datos";
    $port = "5432";

    $conexion = pg_connect("host=$host port=$port password=$passwd
    user=$user dbname=$db");
    /* Si la conexión a la base de datos falla: */
    if (!$conexion){
        echo "Conexion fallida!!";
        exit;
    }

?>
```

Cerrar conexión a la base de datos:

```
<?php
    pg_close($conexion);
?>
```

Función para cambiar la fecha del sistema al idioma en español.

```
<?
function fecha(){
    $mes = date("n");
    $mesArray = array(
        1 => "Enero",
        2 => "Febrero",
        3 => "Marzo",
        4 => "Abril",
        5 => "Mayo",
        6 => "Junio",
        7 => "Julio",
        8 => "Agosto",
```

```
9 => "Septiembre",
10 => "Octubre",
11 => "Noviembre",
12 => "Diciembre"
);

$semana = date("D");
$semanaArray = array(
"Mon" => "Lunes",
"Tue" => "Martes",
"Wed" => "Miercoles",
"Thu" => "Jueves",
"Fri" => "Viernes",
"Sat" => "Sábado",
"Sun" => "Domingo",
);

$mesReturn = $mesArray[$mes];
$semanaReturn = $semanaArray[$semana];
$día = date("d");
$año = date("Y");

return $semanaReturn." ".$día." de ".$mesReturn." de ".$año;
}
```

?>

Función para desplegar la fecha en la página

```
<?php
    require "fecha.php";
    $hoy= fecha();
    print_r($hoy);
?>
```

Función para obtener de la base de datos la lista de servicios de origen del Hospital General y contenerlos en un menú desplegable.

```
<?php
    $query1= "select * from servicio_origen";
    $resul= pg_exec($conexion, $query1);
    $num= pg_numrows($resul);
    for($i=0;$i<$num; $i++)
    {
```

```
        $datos=pg_fetch_array($resul);
        echo'<option
value=".$datos['serclave'].">'.$datos['semombreservicio'].'</option>';
    }
?>
```

Función para obtener la lista de médicos y contenerlas en un menú desplegable

```
<?php
$query1 = "select * from doctor order by docnombredocctor asc";
$resul= pg_exec($conexion, $query1);
$num= pg_numrows($resul);
//echo $num;
for($i=0;$i<$num; $i++)
{
    $datos=pg_fetch_array($resul);

    echo '<option value = ".$datos['docclave'].">
        '.$datos['docnombredocctor'].' </option>';
}
?>
```

Función para comprobar si un número de expediente ya existe en la base de datos.

```
<?php
    $query1 = "select * from paciente where pacExpediente =
    '$numExpediente'";
    $resul= pg_exec($conexion, $query1);
    $num= pg_numrows($resul);
    if ( $num != "" )
    {
        echo("<script language='javascript'>alert('Ese número de
        expediente ya existe. Por favor, corrija la información.');" history.back();</script>");
    }
?>
```

Función para insertar datos en la tabla visita

```
<?php
$querydos = "insert into visita (pacexpediente) values ('$numExpediente')";
$result= pg_exec($conexion, $querydos);
?>
```

Función para insertar datos en la tabla violencia

```
<?php
$query164 = "insert into violencia_paciente values ( '4', '$numExpediente',
'$abandonoAgr', '$abandonoRec', '$clasifiabandono' )";
$result=pg_exec($conexion, $query164);
?>
```

Función para insertar datos en la tabla de signos vitales

```
<?php
$query236 = "insert into paciente_signo values ( '$valor', '1', '$peso')";
$result=pg_exec($conexion, $query236);
?>
```

Función para insertar la anomalía "aislamiento" de la escala Missouri en la base de datos, preguntando primero si se dio clic en el valor.

```
<?php
    if ( $aislamiento == 1 )
    {
        $query219 = "insert into paciente_missouri values (
        '$valor', '52')";
        $result=pg_exec($conexion, $query219);
    }
?>
```

Función para insertar un diagnóstico en la base de datos preguntando si se eligió.

```
<?php
    if ( $diagClaveDiagnostico1 != '0' )
    {
        $query242 = "insert into diagnostico_paciente values
        ( '$diagClaveDiagnostico1', '$valor' )";
        $resul=pg_exec($conexion, $query242);
    }
?>
```

Código para encontrar a un paciente en la base de datos.

```
<?php
$query1 = "select * from paciente where pacExpediente like '%$numExpediente%'
or pacExpedienteInterno like '%$numExpedienteInterno%' or pacNombre like
 '%$pacNombre%' or pacPaterno like '%$pacPaterno%' or pacMaterno like
 '%$pacMaterno%'";
$resul= pg_exec($conexion, $query1);
$num= pg_numrows($resul);

$i = 0;
if ($resul)
{
    while ($lineaA = pg_fetch_row($resul) )
    {
        $arregloA[$i++] = $lineaA;
    }
}
?>
```

Código para almacenar los datos encontrados en una tabla

```
<?php
foreach($arregloA as $resul)
    {
        echo "<tr bgcolor='\#0092CD'"
onMouseOver='\setPointer(this, '\#FFCC66')\" onMouseOut='\setPointer(this,
'\#0092CD')\">";
        echo "<td><a
href='seguimiento3.php?numExpediente=$resul[0]'\>$resul[0]</a></td>";
        echo "<td><a
href='seguimiento3.php?numExpediente=$resul[0]'\>$resul[1]</td></a></td>";
        echo "<td><a
href='seguimiento3.php?numExpediente=$resul[0]'\>$resul[9] $resul[10]
$resul[11]</a></td>";
        echo "<td><a
href='seguimiento3.php?numExpediente=$resul[0]'\>$resul[17]</a></td>";
        echo "</tr>";
    }
?>
```

Código para obtener el idvisita de la última visita del paciente

```
<?php
$query = "select idvisita from visita where pacexpediente= '$numExpediente' order
by idvisita desc";
$resul = pg_exec($conexion, $query);
$num= pg_numrows($resul);

$i=0;
while ($datos = pg_fetch_row($resul))
{
    $arreglo[$i++] = $datos;
}

$valores = $arreglo [0][0];
?>
```

Teniendo este dato guardado en la variable `$valor` se puede obtener toda la información de la visita anterior del paciente, por ejemplo para los signos vitales del paciente tomados en su visita anterior:

```
<?php
$query3 = "select * from paciente_signo where idvisita = '$valor'";
$resul= pg_exec($conexion, $query3);
$num3= pg_numrows($resul);

if($num3)
{
    $i = 0;
    if ($resul)
    {
        while ($lineaC = pg_fetch_row($resul) )
        {
            $arregloC[$i++] = $lineaC;
        }

        $peso = $arregloC[0][2];
        $pulso = $arregloC[1][2];
        $talla = $arregloC[2][2];
        $tas = $arregloC[3][2];
        $tad = $arregloC[4][2];
        $respiracion = $arregloC[5][2];
    }
}

?>
```

Cada signo vital se guardó en una variable con el nombre que le corresponde para después desplegarla en su respectivo campo en el formulario.

```
<input type="text" name="peso" size="5" maxlength="6" value = "<? if($num3)
echo $peso; ?>" >
```

```
<input type="text" name="pulso" size="5" maxlength="6" value = "<? if($num3)
echo $pulso; ?>" onKeyPress="javascript:return admitir_numeros();">
```

```
<input type="text" name="talla" size="5" maxlength="6" value = "<? if($num3)
echo $talla; ?>" >
```

```
<input type="text" name="TA_S" size="5" maxlength="5" value = "<? if($num3)
echo $tas; ?>" onKeyPress="javascript:return admitir_numeros();">
```

```
<input type="text" name="TA_D" size="5" maxlength="5" value = "<? if($num3)
echo $tad; ?>" onKeyPress="javascript:return admitir_numeros();">
```

```
<input type="text" name="respiracion" maxlength="5" size="5" value = "<?
if($num3) echo $respiracion; ?>" onKeyPress="javascript:return
admitir_numeros();">
```

Así es como se recuperan todos los datos y se muestran en pantalla. Teniendo el idvisita se puede buscar la información de los pacientes en todas las tablas.

Para desplegar información (en los checkbox) guardada anteriormente, primero se hace la consulta a la base de datos, se almacenan en un arreglo los resultados y después se despliegan en el formulario en HTML

```
<?php
```

```
    $querya = "select missnombreaanomia
                from paciente_missouri as p, missouri as m
                where p.missclave = m.missclave and idvisita =
'$valor' and tipomissclave = '1'";
```

```
    $resul1 = pg_exec($conexion, $querya);
    $num1=pg_numrows($resul1);
    $i = 0;
    if ($resul1)
    {
        while ($lineaA = pg_fetch_row($resul1) )
        {
            $datos[$i++] = $lineaA;
        }
    }
    ?>
```



Para desplegarlos en el checkbox se pregunta en cada campo si se encontró en la base de datos y se selecciona, de lo contrario se deja el campo vacío. (Se muestran unos campos de ejemplo)

```
<input type="checkbox" name="bloqueo" value="1" <? for($i=0;$i<$num1; $i++)  
if($datos[$i][0] == 'Bloqueo' ) echo "checked" ; ?> >
```

```
<input type="checkbox" name="perseverante" value="1" <? for($i=0;$i<$num1;  
$i++) if($datos[$i][0] == 'Perseverante') echo "checked"; ?> >
```

```
<input type="checkbox" name="lentitud" value="1" <? for($i=0;$i<$num1; $i++)  
if($datos[$i][0] == 'Lentitud') echo "checked"; ?>>
```

```
<input type="checkbox" name="fugaIdeas" value="1" <? for($i=0;$i<$num1; $i++)  
if($datos[$i][0] == 'Fuga de ideas') echo "checked"; ?> >
```

```
<input type="checkbox" name="circunstancial" value="1" <? for($i=0;$i<$num1;  
$i++) if($datos[$i][0] == 'Circunstancial') echo "checked"; ?> >
```

Para recuperar y desplegar la información (en menús desplegables) del paciente guardada con anterioridad, se hace la consulta a la base de datos, se guardan en un arreglo y se muestra en el en el menú.

Por ejemplo para los psicofármacos.

```
<?php  
  
$query1="select * from psicofarmaco order by psifarNombreFarmaco";  
$resul= pg_exec($conexion, $query1);  
$num= pg_numrows($resul);  
$bandera = true;  
for($i=0;$i<$num; $i++)  
{  
    $datos=pg_fetch_array($resul);  
?  
    <option value="<? echo $datos['psifarclave'] ?>" <? if ($cont2 < $tope2 && $var2  
    == 1 && $bandera && $arefear[$cont2][0] == $datos['psifarclave'])  
  
        echo "selected" ;
```

```
$bandera = false;

$inc = 1 ;
    }

> <? echo $datos[psifamombrefarmaco] ?> </option>
?>
    <? }
?>
```

Código para determinar el número de veces que un paciente ha ido a consulta

```
$query = "select idvisita from visita where pacexpediente = '$numExpediente' order
by idvisita asc";
    $resul= pg_exec($conexion, $quer);
    $nume= pg_numrows($resul);
```

Código para determinar si un paciente ya ha sido dado de alta o está en seguimiento.

```
<?php

$queryb = "select distinct alta from seguimiento as s, visita as v where s.idvisita =
v.idvisita and v.pacexpediente = '$numExpediente'";

$resulta=pg_exec($conexion, $queryb);
$numero=pg_numrows($resulta);
$i = 0;
```

//VALIDACION DE TODOS LOS CASOS

```
if ($nume == 1)
{
    if ($pacalta == 1)
    {
        echo "El Paciente ha Sido Dado de Alta";
    }

    else
    {
        echo "El Paciente Está en Observación";
    }
}

else
{
    for ($i = 0; $i < $numero; $i++)
    {
        $datos=pg_fetch_array($resulta);

        if( $datos['alta'] == 1 and $nume > 1)
        {
            echo "El Paciente ha Sido Dado de Alta";
        }

        else
        {
            echo "El Paciente Está en Observación";
        }
    }
}

?>
```

Código para comprobar la contraseña del usuario:

```
<?php

$querr= "select usuario, password from usuarios where usuario='$usuario' and
password='$contrasena";
$resu=pg_exec($conexion,$querr);
$num=pg_numrows($resu);
pg_close($conexion);
if ( $num )
    {
        echo("<script
language='javascript'>window.open('principal.php','principal','toolbar=no
resizable=no width=900 height=600 scrollbars=yes menubar=no status=no
left=0 top=0');</script>");
    }

    else
        {
            echo("<script language='javascript'>alert('Los datos ingresados son
incorrectos.');
```

Función para eliminar los datos del paciente en todas sus visitas.

```
<?php

for ($i=0; $i<$num; $i++)
    {
        $quers = borrar($conexion, "delete from seguimiento
        where idvisita      =".$arreglo[$i][0]);

        $querr = borrar($conexion, "delete from visita where
        idvisita      =".$arreglo[$i][0]);
    }

?>
```

Los números idvisita del paciente se buscan previamente y se guardan en un arreglo. El ciclo for es para recorrer el arreglo y borrar la información en las tablas "visita" y "seguimiento" que tengan como idvisita los mismos que el arreglo.

Función para borrar los datos de un paciente en la tabla "Paciente"

```
<?php
```

```
$query1 = borrar($conexion,"delete from paciente where pacExpediente = '$numExpediente'");
```

Función para borrar todas las terapias de todas las consultas de un paciente.

```
for ($i=0; $i<$num; $i++)
```

```
{
    $query4 = borrar($conexion,"delete from paciente_terapia
    where Idvisita = ".$arreglo[$i][0]);
}
```

```
?>
```

Función para generar la estadística de diagnósticos hechos por doctor.

```
<?php
```

```
$numPsicofarmaco = 1;
$impresion = "";
$psicofarmacos = "SELECT a.docNombreDoctor, count( * ) FROM DOCTOR AS a,
PACIENTE AS b WHERE a.docClave = b.docClave GROUP BY
a.docNombreDoctor order by docnombredoctor";
```

```
$resul = pg_exec($conexion, $psicofarmacos);
$num= pg_numrows($resul);
$i = 0;
```

```
if ($resul)
{
    while ($lineaA = pg_fetch_row($resul) )
    {
        $arregloA[$i++] = $lineaA;
    }
}
```

```
foreach($arregloA as $resultado)
{
    echo "<tr><td> $resultado[0] </td><td>
    $resultado[1] </td></tr>";
    $Impresion = 1;
}

if ( $Impresion == "" )
{
    echo "<tr><td colspan='2'>La consulta no arrojó
    ningún dato.</td></tr>";
}

?>
```

Función para actualizar la información del médico.

```
<?php

$query = "UPDATE usuarios set usuario = '$usuario', password = '$pass', nombre
= '$nombre', email = '$mail', rfc = '$rfc' WHERE Idusuario = $Id";

$resul = pg_exec($conexion, $query);

?>
```