

**Gestión de Datos**

# **Clínica FRBA**

**Trabajo práctico  
2do cuatrimestre 2016**

**Grupo 52 – WASD**

**Leonel Fishkel  
Ezequiel Gauna  
Jeanette Mamani Gutierrez  
Hernán Trigo Mogro**

## *ÍNDICE*

- Migración
- Decisiones de diseño
- Modelo de datos
- Aplicación desktop

### **Migración:**

Al momento de realizar la migración de los datos desde la tabla Maestra, se encontraron las siguientes inconsistencias o problemas:

- Las fechas de compras de todos los bonos siempre son en el futuro respecto de la fecha de su uso. Se decidió migrar los datos tal y como se encuentran, ya que elegir cualquier otra fecha para la compra sería totalmente arbitrario.
- Existen registros de turnos para los domingos, en los cuales la clínica se encuentra cerrada. Se decidió migrar los datos tal y como se encuentran, ya que es imposible determinar en qué fechas ocurrieron realmente esos turnos, aunque se asume que se trata de datos de hechos que efectivamente ocurrieron.
- Los afiliados no presentan información de sexo ni estado civil. Se decidió migrar estos datos con valores especiales, “X” y “Migrado/a” respectivamente, considerando instruir a los administrativos de la clínica para que actualicen estos datos a medida que los pacientes vayan siendo atendidos post-migración.
- Los profesionales, al igual que los afiliados, no presentan información respecto del sexo. Se tomó la misma decisión que en el caso anterior (migrar con valor “X”).
- Los afiliados no presentan información acerca de sus grupos familiares. Se decidió migrar a todos los afiliados existentes como “principales” sin grupo familiar, ya que es imposible determinar filiaciones de ningún tipo sobre la base de los datos informados.
- Los profesionales no presentan número de matrícula. A los fines prácticos se decidió utilizar el número de DNI de cada uno como número de matrícula.

### **Decisiones de diseño:**

A continuación, se detallan diversas decisiones de diseño y consideraciones que se tomaron respecto del funcionamiento del sistema:

- La fecha del sistema se establece como 1° de enero de 2016 a las 00:00 horas, entendiendo que la migración se realiza en ese momento ya que los datos presentes en la tabla maestra llegan hasta el día anterior inclusive.
- Se utiliza el número de documento como nombre de usuario para las altas de nuevos usuarios, y “w23e” como contraseña por defecto.
- Se asignaron las siguientes funcionalidades:

	Afiliado	Administrativo	Profesional
ABM Roles		X	
ABM Afiliados		X	
Ver Estadísticas		X	
Registrar llegada		X	
Comprar bonos	X	X	
Pedir Turno	X		
Cancelar Turno	X		X
Registrar Agenda			X
Diagnosticar			X

- Se desnormalizó el sexo y el tipo de documento en las tablas de Afiliado y Profesional, manteniendo sin embargo una restricción CHECK en esas columnas para evitar la carga de datos erróneos.
- Se utilizó el campo Plan\_Med\_Codigo de la tabla Maestra como PK de la tabla de planes médicos del modelo.
- Se utilizó el campo Turno\_Numero de la tabla Maestra como PK de la tabla de turnos del modelo.
- Se utilizó el campo Especialidad\_Codigo de la tabla Maestra como PK de la tabla de especialidades médicas del modelo.
- Se utilizó el campo Tipo\_Especialidad\_Codigo de la tabla Maestra como PK de la tabla de tipos de especialidad médica del modelo.
- Se asignaron los tres roles existentes al usuario admin pedido por el enunciado, para que el mismo contase con todas las funcionalidades existentes, como fuera solicitado.

### **Modelo de datos:**

A continuación, detallamos las tablas que se crearon para modelar cada uno de los puntos solicitados del enunciado.

#### **- ABM de Rol**

Se creó la tabla Rol, con una PK autoincremental, un nombre y un campo bit para determinar si se encuentra activo o no (borrado lógico).

Se creó la tabla Funcionalidad, con una PK autoincremental y un nombre.

Se creó la tabla intermedia FuncionalidadPorRol para modelar la relación muchos a muchos entre los roles y las funcionalidades, con una FK a la PK de Rol y otra FK a la PK de Funcionalidad. La PK está compuesta por ambas FK.

#### **- Login y seguridad**

Se creó la tabla Usuario, con una PK autoincremental, un nombre, una contraseña, un contador de intentos de logueo, un bit para determinar si se encuentra activo o no, y un nombre a mostrar por pantalla.

Se creó la tabla RolPorUsuario, con una FK a la PK de Usuario y otra FK a la PK de Rol. La PK está compuesta por ambas FK.

#### **- ABM de afiliados**

Se creó la tabla Afiliado, con una PK autoincremental, FKs a Usuario, EstadoCivil, PlanMédico y a sí misma (para modelar los grupos familiares). Además se incluyeron todos los datos personales del modelo de negocios, más campos para el borrado lógico (afiliado\_activo), y la cuenta de bonos utilizados (afiliado\_cantidad\_bonos\_usados).

Para modelar los estados civiles se optó por una tabla separada, EstadoCivil, consistente en una PK autoincremental, y un campo de texto para el nombre del estado civil. No así para el modelado de los sexos y tipos de documento: estos datos fueron desnormalizados por considerar que el aumento en el tamaño de las tablas que los incluyeran sería negligible.

Se creó la tabla HistorialPlan, con una PK autoincremental, una FK a la PK del afiliado y un campo de texto libre para indicar el motivo del cambio y de qué plan a qué plan fue el mismo.

#### **- ABM de profesional**

Se creó la tabla Profesional, con una PK autoincremental, una FK a la PK del usuario, y campos para todos los datos personales del modelo de negocios.

#### **- ABM de especialidades médicas**

Se creó la tabla TipoEspecialidad, con una PK autoincremental y un nombre de tipo de especialidad.

Se creó la tabla Especialidad, con una PK autoincremental, una FK a la PK del tipo de especialidad y un nombre de especialidad.

Se creó la tabla intermedia EspecialidadPorProfesional para modelar la relación muchos a muchos entre los profesionales y sus especialidades, con una FK a la PK de profesional y otra FK a la PK de especialidad. La PK está compuesta por ambas FK.

#### **- ABM de plan**

Se creó la tabla PlanMedico, con una PK autoincremental, y campos para todos los datos del modelo de negocios.

#### **- Registrar agenda profesional**

Se creó la tabla Hora, con una PK autoincremental, y un campo TIME para modelar el

comienzo de cada bloque de 30 minutos que compone los turnos dentro del horario de atención de la clínica.

Se creó la tabla Agenda, con una PK autoincremental, FKs a la PK del profesional, de la especialidad, y a las horas de comienzo y final de cada jornada laboral. Además se incluyeron dos campos fecha para modelar el rango de aplicabilidad de la agenda, y un campo agenda\_dia de tipo CHAR para especificar el día de la semana correspondiente a cada registro. De esta manera se logra una granularidad total en la agenda del profesional: permite que atienda cualquier especialidad que él o ella posea, en cualquier día de la semana, en cualquier franja horaria y para cualquier rango de fechas.

- Compra de bonos

Se creó la tabla Compra, con una PK autoincremental, una FK a la PK del afiliado que realizó la compra, y campos para almacenar la fecha y hora de la compra, así como la cantidad de bonos adquirida y el monto total de la transacción.

Se creó la tabla Bono, con una PK autoincremental, y FKs al plan médico al que corresponde, al afiliado que lo compró, a la compra en la que fue adquirido y al afiliado que finalmente lo utilizó. El número de consulta médica, al ser propio de cada afiliado, se almacena en el registro correspondiente al mismo en la tabla Afiliado.

- Pedido de turno

Se creó la tabla Turno, con una PK autoincremental, FKs al afiliado, al profesional, al tipo de especialidad y a la posible cancelación del turno. También se incluye un campo de fecha y otro de hora, y un campo BIT para marcar cuando el afiliado efectivamente se presenta en la clínica para su atención.

- Registro de llegada para atención médica

Se creó una tabla ConsultaMedica, con una PK autoincremental, una FK a la PK del bono utilizado para efectivizar la consulta, un campo DATETIME para almacenar la fecha y la hora en la que el afiliado se presenta para su atención, y un campo BIT para que el profesional indique si la consulta ocurrió.

- Registro de resultado para atención médica

Se creó la tabla Diagnostico, con una PK autoincremental, una FK a la PK de la consulta médica, y un campo de texto libre para indicar el diagnóstico propiamente dicho.

Se creó la tabla Sintoma, con una PK autoincremental, una FK a la PK de la consulta médica, y un campo de texto libre para indicar el síntoma propiamente dicho.

- Cancelar atención médica

Se creó la tabla TurnoCancelado, con una PK autoincremental, una FK a la PK del tipo de cancelación, un campo de texto libre para el motivo de la misma, y un campo CHAR para indicar si la cancelación fue realizada por un afiliado o un profesional.

Se creó la tabla TipoCancelacion, con una PK autoincremental y un campo CHAR para indicar el nombre del tipo de cancelación.

### **Aplicación Desktop:**

A continuación, detallamos las tecnologías y patrones utilizados para el desarrollo de la interfaz de usuario.

- Para el desarrollo de la UI se utilizó C# Windows forms como estipulaba el enunciado.
- Se utilizó inyección de dependencias para manipular una única instancia de cada formulario.
- También se creó una DLL aparte para el acceso a la base de datos cuyas únicas instancias de objetos de acceso a datos también son inyectadas en los formularios.
- Para la inyección de dependencias se utilizó el framework Autofac 4.1.1.
- Para el acceso a datos y mapeo de las tablas (modelo), views, functions y store procedures se utilizó el framework EntityFramework 6.1.3.
- Todos los métodos de los objetos de acceso a datos (DAO) invocan exclusivamente store procedures, functions o views. De ese modo la lógica de negocio queda volcada en la base de datos.