ClinicaFRBA – Estrategia

Contenido

[Decisiones tomadas 2](#_Toc372246706)

[Modelo de datos 2](#_Toc372246707)

[Decisiones de desarrollo 2](#_Toc372246708)

[Arquitectura de la aplicación .NET 3](#_Toc372246709)

[Diagrama de componentes 3](#_Toc372246710)

[Diagrama de clases de acceso a datos 5](#_Toc372246711)

[Diagrama de clases de Capa de presentación 6](#_Toc372246712)

[Uso de la aplicación 7](#_Toc372246713)

# Decisiones tomadas

* Decisión del sexo: Enum con los valores
  + 0 – Indefinido
  + 1 – Mujer
  + 2 – Hombre

Por defecto todos los valores que existan en la tabla maestra tienen sexo indefinido, pudiéndose modificar su valor desde la pantalla de modificación de afiliado o profesional.

* Grupo familiar: Se consideran que son del mismo grupo si tienen la misma dirección
* Login: El hash del password se crea desde .NET al momento de hacer el login y se compara contra el varbinary que tiene en el campo password de la tabla Usuario.
* Usuarios: El nombre de usuario es el número de documento y su contraseña es ‘afiliado’ y para los profesionales ‘profesional’
* Tipo documento: Todos quedan con DNI. En la aplicación pueden elegirse otros para las altas o modificaciones.
* No hay tabla de grupos de afiliados, en la tabla afiliado hay un número principal, que van de 100 en 100 y los secundarios de 1 en 1
* La fecha de prescripción es la fecha en que el médico crea la receta y esta fecha se guarda en el bono
* Al eliminar un turno de un profesional, se le cancela el turno al afiliado, sin buscarle reemplazo.
* Al cancelar un rango de días, el profesional puede seleccionar hasta 7 días corridos.

# Modelo de datos

Ver el archivo DER adjunto.

# Decisiones de desarrollo

Cada vez que se inicia la aplicación se genera un archivo de log en la carpeta donde este corriendo el binario. Dicho archivo de log guarda las llamadas a stored procedures que se hacen, sus parámetros y las excepciones.

Además para ayudar a las pruebas, hay dos barras de estado en toda la aplicación con la información del connectionstring, la fecha actual y el usuario identificado en la aplicación.

# Arquitectura de la aplicación .NET

## Diagrama de componentes



Para trabajar de forma desacoplada entre la presentación y los datos almacenados, la solución se compone de varios proyectos:

ClinicaFRBA: Es la capa de presentación, contiene los winforms que usarán los usuarios finales.

GestionDomain: Conoce las reglas del negocio de la clínica. Resuelve las peticiones de ClinicaFRBA.

GestionDAL: En esta capa se transforman las entidades de negocio en objetos compatibles con el Conector. Al recibir los DataSet del conector, los convierte mediante Builders en entidades que GestionDomain usa.

GestionConector: Es la capa más cercana a la base de datos. Se basa en ADO.NET y sabe como ejecutar stored procedures y devolver DataSets.

GestiónCommon: Contiene objetos comunes a todas las clases. Podemos dividir sus componentes en:

Entidades: Representan los objetos del negocio.

Helpers: Realizan tareas comunes. Por ejemplo FechaHelper se encarga del manejo de las fechas en el sistema.

## Diagrama de clases de acceso a datos



Para interactuar contra la base de datos creamos una capa llamada GestionConector que tiene la lógica para ejecutar stored procedures o queries, devolviendo objetos del tipo DataSet.

**Este conector usa el connectionstring del archivo de configuración de la aplicación.**

La capa DAL devuelve objetos que representan a las entidades, convirtiendo las filas del tipo DataRow mediante un objeto builder, que conoce como mapear cada campo contra cada propiedad.

Al conector lo usa la capa DAL, que es la encargada de transformar los objetos de las entidades en parámetros de los stored procedures y usa el conector para comunicarse con la base de datos.

A la capa DAL la conoce la capa de Domain que se encarga de realizar las validaciones de negocio necesarias y gestionar los errores que hayan surgido desde el modelo de datos para que sean legibles para el usuario final.

Para ello cada respuesta de la capa Domain es del tipo IResultado, que enmascara los errores y las devoluciones.

La capa de presentación conoce los datos a partir de invocaciones a la capa Domain.

## Diagrama de clases de Capa de presentación



Todos los formularios de ABM heredan de FormularioBase que conoce una lista de validaciones y sabe validar los datos antes de realizar alguna acción.

A su vez, los formularios de alta, baja y modificación heredan de formularios base que ya tienen los botones predefinidos para cada tipo de operación.

## Uso de la aplicación

En esta sección se describe cómo usar la aplicación

Abm de rol

Gestión de clínica-> Roles

Login y seguridad:

Archivo -> Sesión -> Iniciar Sesión

Archivo -> Sesión -> Cerrar Sesión

Registro de usuario:

Al crear un afiliado o profesional se le asigna un usuario automáticamente con el nro de documento como username.

Se consultan desde Gestión de Clínica -> Usuarios

ABM de afiliados

Gestión de afiliados -> Afiliados

ABM de profesional

Gestión de profesionales -> Profesionales

ABM de Especialidades médicas

Gestión de profesionales -> Especialidades médicas

ABM de planes

Gestión de clínica -> Planes

Registrar Agenda del médico

Agenda -> Registrar

Compra de bonos

Gestión de afiliados -> Compra de bonos

Pedir Turno

Gestión de afiliados -> Pedir turno

Registro de llegada para atención médica

Gestión de afiliados -> Registro de llegada

Registrar resultado para atención médica

Gestión de afiliados -> Registro de resultados

Cancelar atención médica

Gestión de clínica-> Cancelaciones -> Afiliado

Gestión de clínica-> Cancelaciones -> Profesional

Generar receta

Al registrar resultados de la atención médica se pregunta si quiere registrar recetas.

Listados estadísticos

Archivo -> Estadísticas