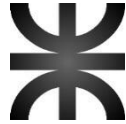


---

UNIVERSIDAD TECNOLÓGICA NACIONAL - FACULTAD REGIONAL CORDOBA  
INGENIERIA EN SISTEMAS DE INFORMACION



CATEDRA DE HABILITACION PROFESIONAL  
CURSO 4K4 - 2015

---

# FISIKS

## DOCUMENTACION DEL PRODUCTO

### ORGANIZACIÓN

CENTRO DE KINESIOLOGIA Y FISIOTERAPIA 'PIERRE DE COUBERTIN'

### METODOLOGIA DE DESARROLLO

METODOLOGIA AGIL SCRUM

### TEMA

SISTEMA DE GESTIÓN PARA CENTRO KINESIOLÓGICO

### AREA DE APLICACIÓN

KINESIOLOGÍA Y FISIOTERAPIA

---

### PROFESORES

ZOHIL, JULIO CESAR NELSON  
AQUINO, FRANCISCO ALEJANDRO

### DETALLE

ADJUNTO  
JTP

---

### ALUMNOS

BARBOSA, NATALIA  
CHIAVARINO, SANTIAGO  
JUAREZ, FACUNDO  
SANTORO, EXEQUIEL JUAN LUIS  
TRILLO PELLIZZARI, LUCIANO

### LEGAJO

44610  
55421  
55441  
47940  
50929

---

## INDICE

1.0 INTRODUCCION .....	3
1.0 EL PRODUCTO.....	4
1.1 Gestión de Configuración: Estructura del Repositorio.....	4
1.2 Gestión de Configuración: Contenido del producto .....	5
1.2.1 Fase de requerimientos.....	5
1.2.1.1 Información de relevamiento: .....	5
1.2.1.2 Información de planificación.....	5
1.2.2 Fase de análisis.....	6
1.2.2.1 Esquema de base de datos.....	6
Visualización focalizada de entidades/relaciones.....	7
1.2.2.2 Vista lógica - Diagrama de clases (Paradigma Orientado a Objetos).....	11
Visualización focalizada de clases .....	12
1.2.2 Fase de diseño .....	16
1.2.2.1 Vista de despliegue .....	16
1.2.2.1.1 Nodos .....	16
1.2.2.1.1 Hardware.....	16
1.2.2.2 Vista dinámica .....	17
1.2.3 Fase de implementación .....	19
1.2.3.1 Esquema de codificación del Sistema FISIKS.....	19
1.2.3.1.1 Capas del modelo.....	20
1.2.3.2 Módulos funcionales del Sistema FISIKS.....	21
Configuración .....	22
Turnos.....	22
Pacientes .....	22
Historia Clínica.....	22
Tratamientos .....	22
Cobros .....	22
Kinesiólogos.....	22
Usuarios.....	22
Estadísticas .....	22
1.2.4 Trazabilidad del Producto .....	23

## **1.0 INTRODUCCION**

El sistema **FISIKS** fue desarrollado en el marco de un proyecto ágil de desarrollo de software a través la metodología **SCRUM**.

La propuesta resultante se destinó al Centro de Kinesiología y Fisioterapia **Pierre de Coubertin**, con el fin cubrir la necesidad de disponer de un sistema informático que les permita registrar la gestión de sus procesos de negocio.

Este documento permitirá visualizar la descripción detallada del producto desarrollado que el equipo de trabajo involucrado propuso para cumplir con las necesidades planteadas, de modo que se conozcan los detalles de la composición del mismo.

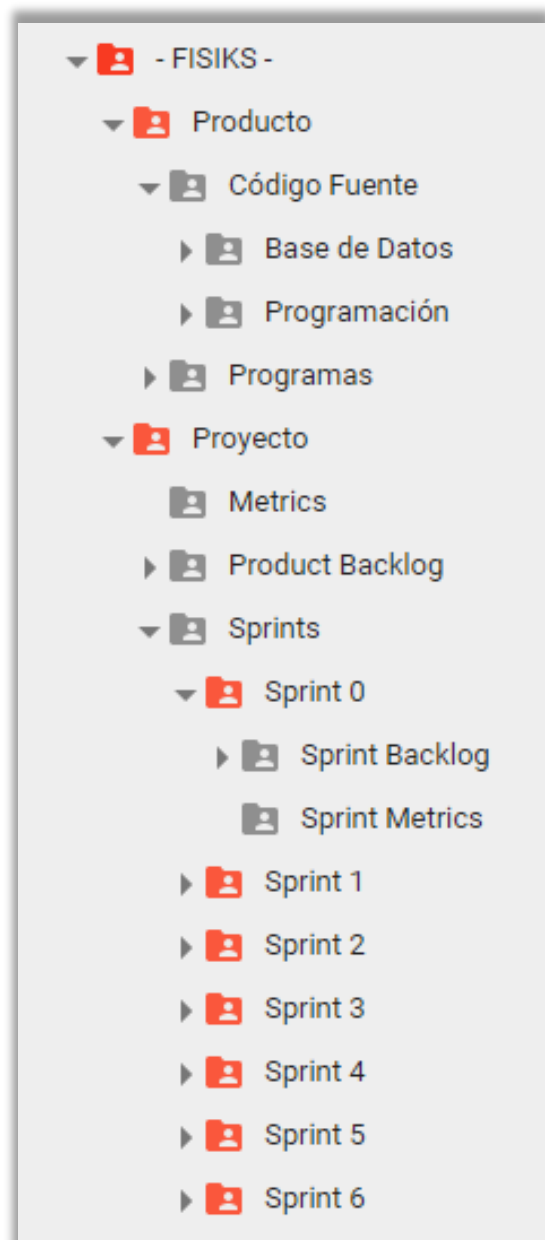
## 1.0 EL PRODUCTO

El sistema de información implementado se denominó “**FISIKS**” (Acrónimos de Fisioterapia / Sistema / Kinesiología).

De acuerdo a la metodología de trabajo implementada, el presente producto atravesó distintas etapas de análisis, diseño y desarrollo, a través de los diferentes Sprints trabajados (iteraciones).

Las distintas etapas, iteraciones y sus incrementos fueron documentadas en un repositorio de información a través de la correspondiente definición de la Gestión de Configuración del Proyecto.

### 1.1 Gestión de Configuración: Estructura del Repositorio



## **1.2 Gestión de Configuración: Contenido del producto**

### **1.2.1 Fase de requerimientos**

#### **1.2.1.1 Información de relevamiento:**

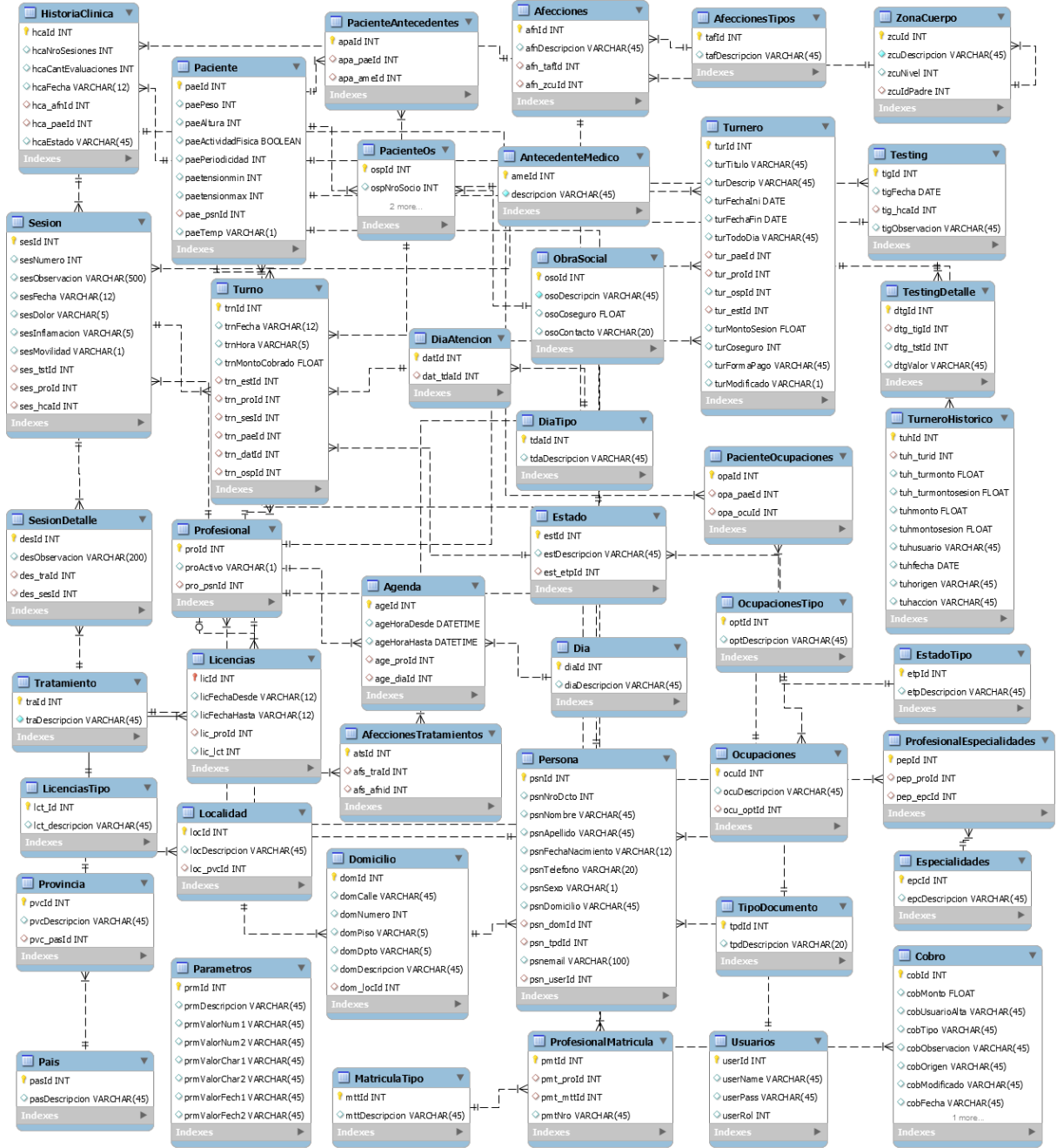
- Respuestas obtenidas sobre las preguntas (abiertas y específicas) realizadas al encargado del centro de rehabilitación.
- Conclusiones realizadas en base al diálogo con el encargado (Cliente).
- Comentarios anexos respecto del proyecto de desarrollo del producto.
  - Consideraciones de acuerdo a los procesos identificados (ámbitos) y consideraciones técnicas (herramientas de trabajo e implementación).
- Imágenes tomadas del centro de rehabilitación (lugar físico de trabajo).
- Imágenes tomadas de formularios utilizados en papel dentro del centro de rehabilitación.

#### **1.2.1.2 Información de planificación**

- Definición del Product Owner
- Definición del Equipo
  - Integrantes del equipo de trabajo
- Definición de Herramientas de Software a utilizar
  - Herramientas para gestión de proyecto
  - Herramientas de programación
  - Herramientas de Diseño y/o documentación
  - Bases de Datos
- Métricas del Proyecto
- Gestión de Configuración del Proyecto
  - Herramientas
  - Estructura del repositorio
  - Listado de Ítems de Configuración
- Glosario Técnico General
- Definición del Product Backlog Inicial (User Stories)

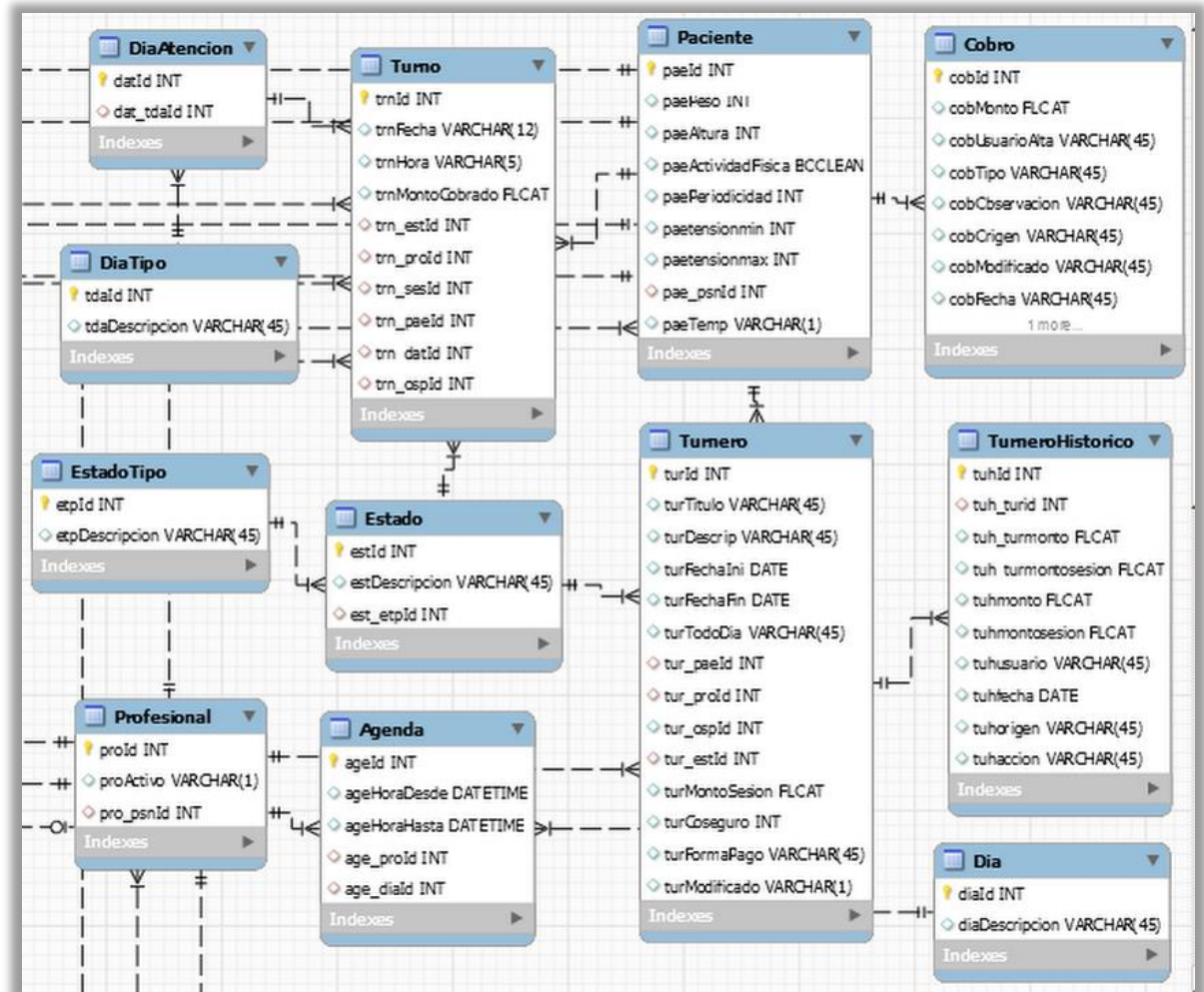
1.2.2 Fase de análisis

1.2.2.1 Esquema de base de datos

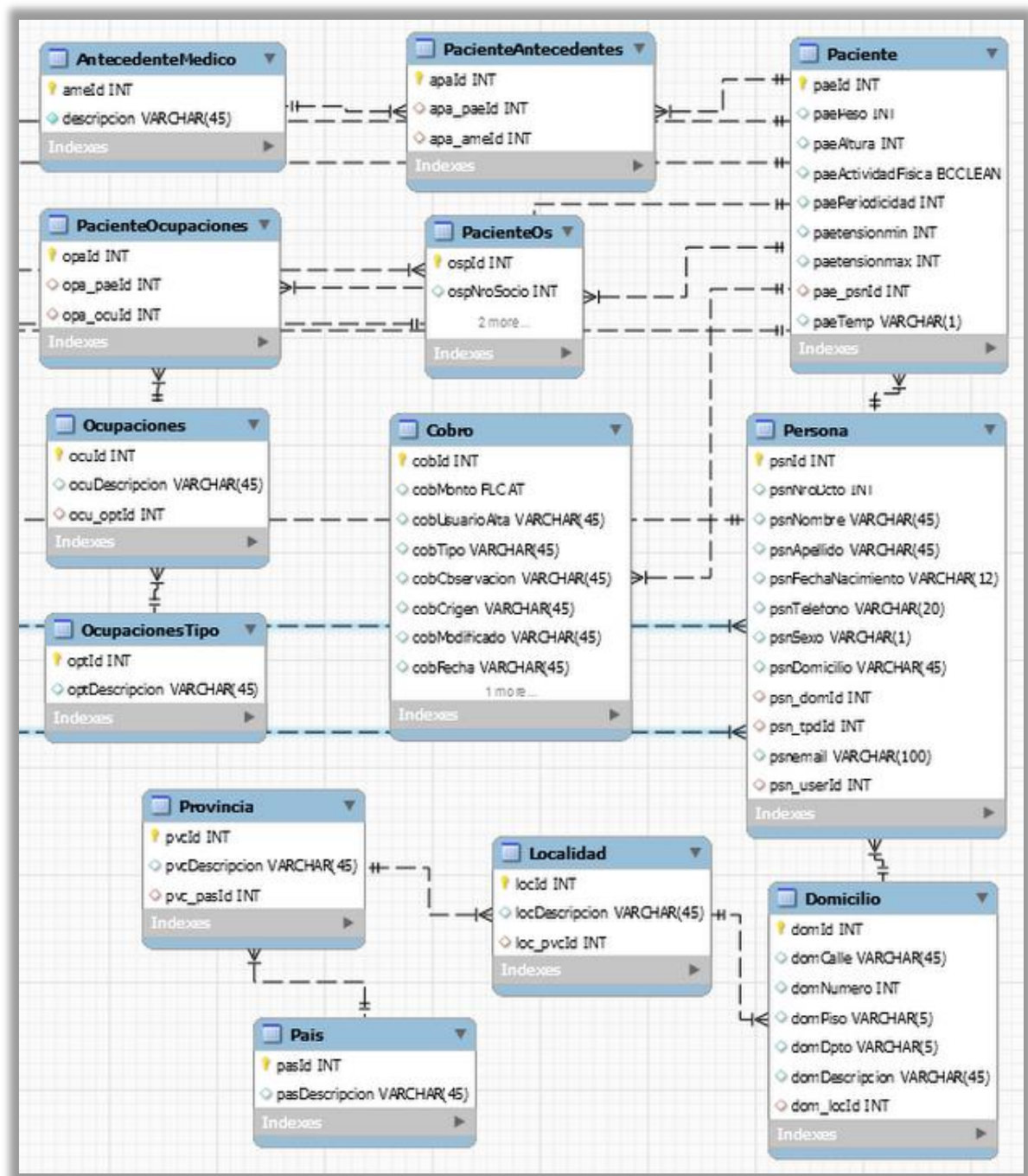


Visualización focalizada de entidades/relaciones

Gestión de Turnos:

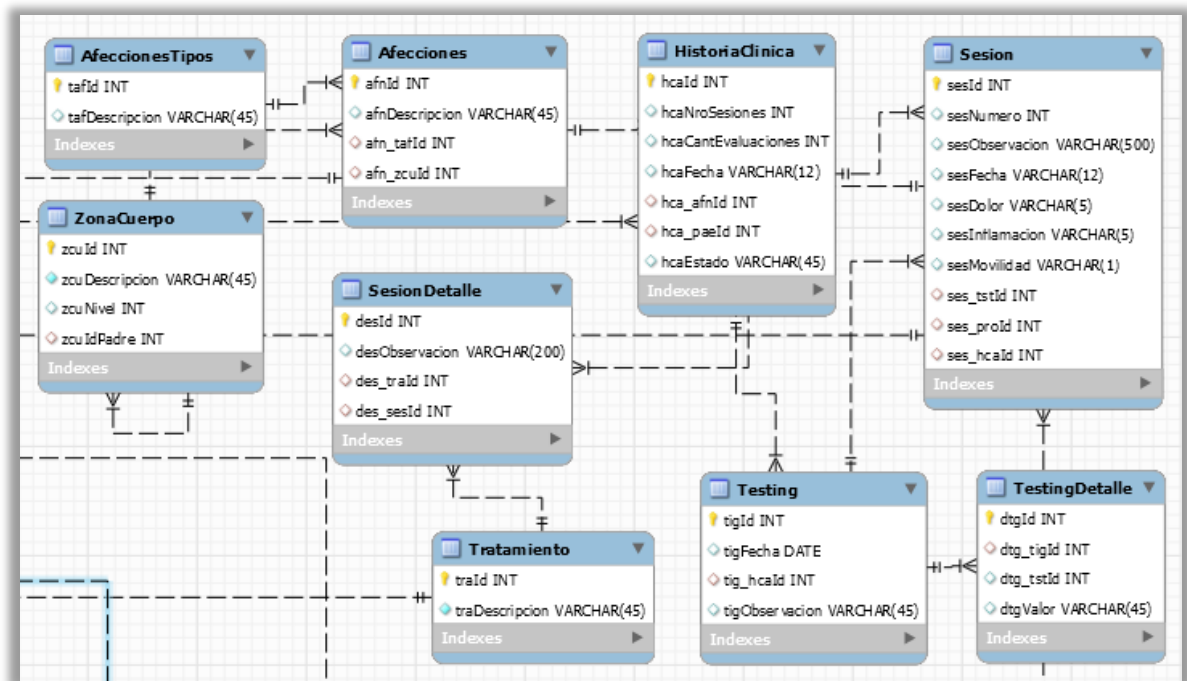


Gestión de Pacientes:

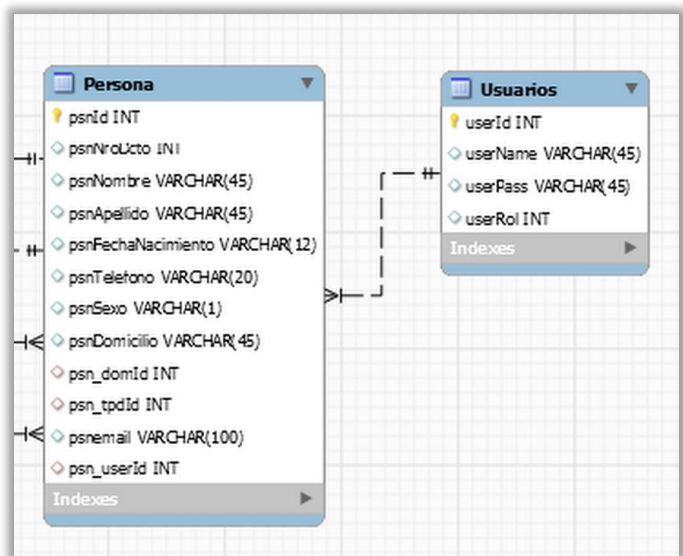




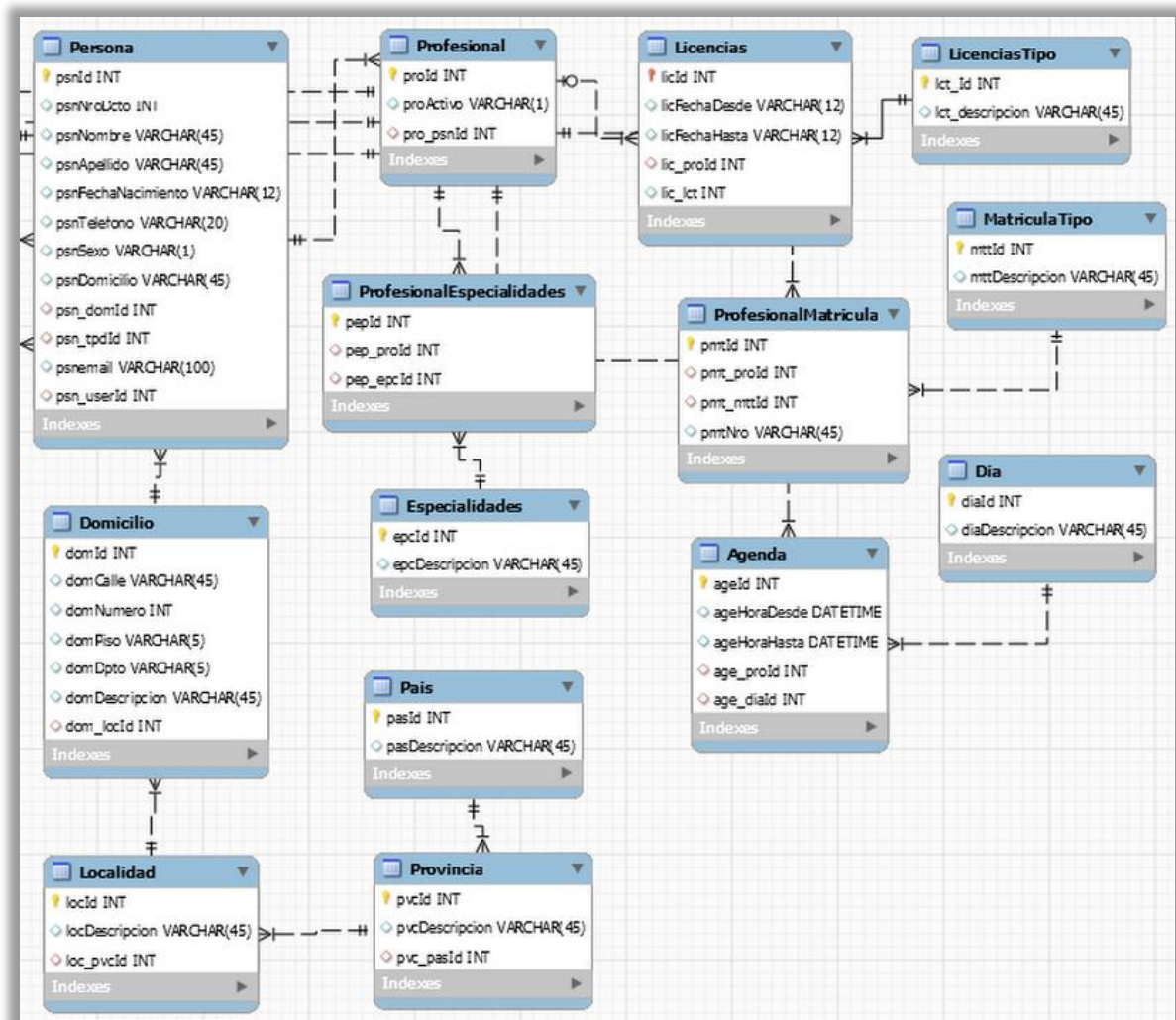
Gestión de Historia Clínica:



Gestión de Usuarios:

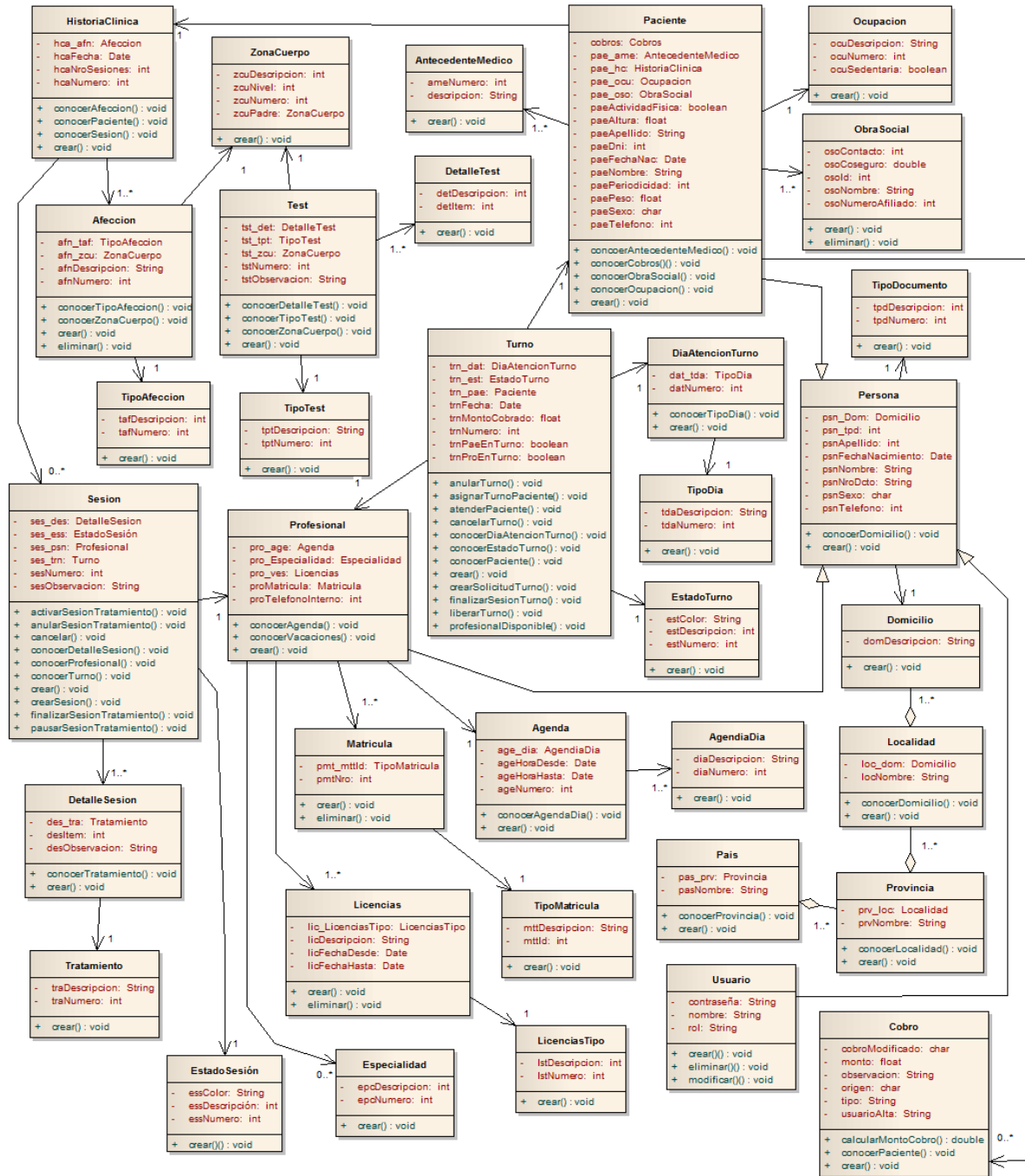


Gestión de Kinesiólogos:



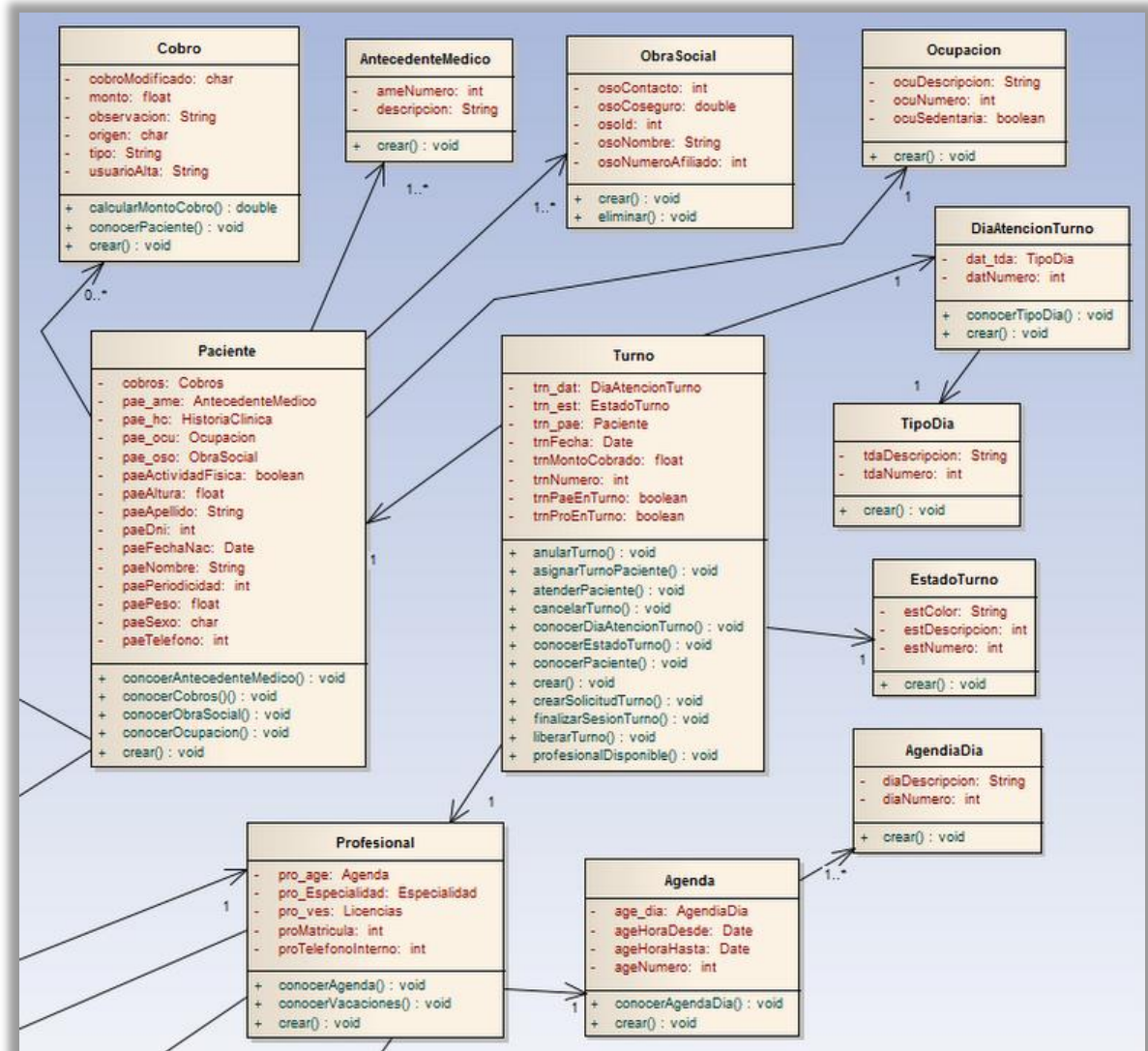
1.2.2.2 Vista l3gica - Diagrama de clases (Paradigma Orientado a Objetos)

Diagrama derivado del esquema de datos propuesto a trav3s del paradigma correspondiente.

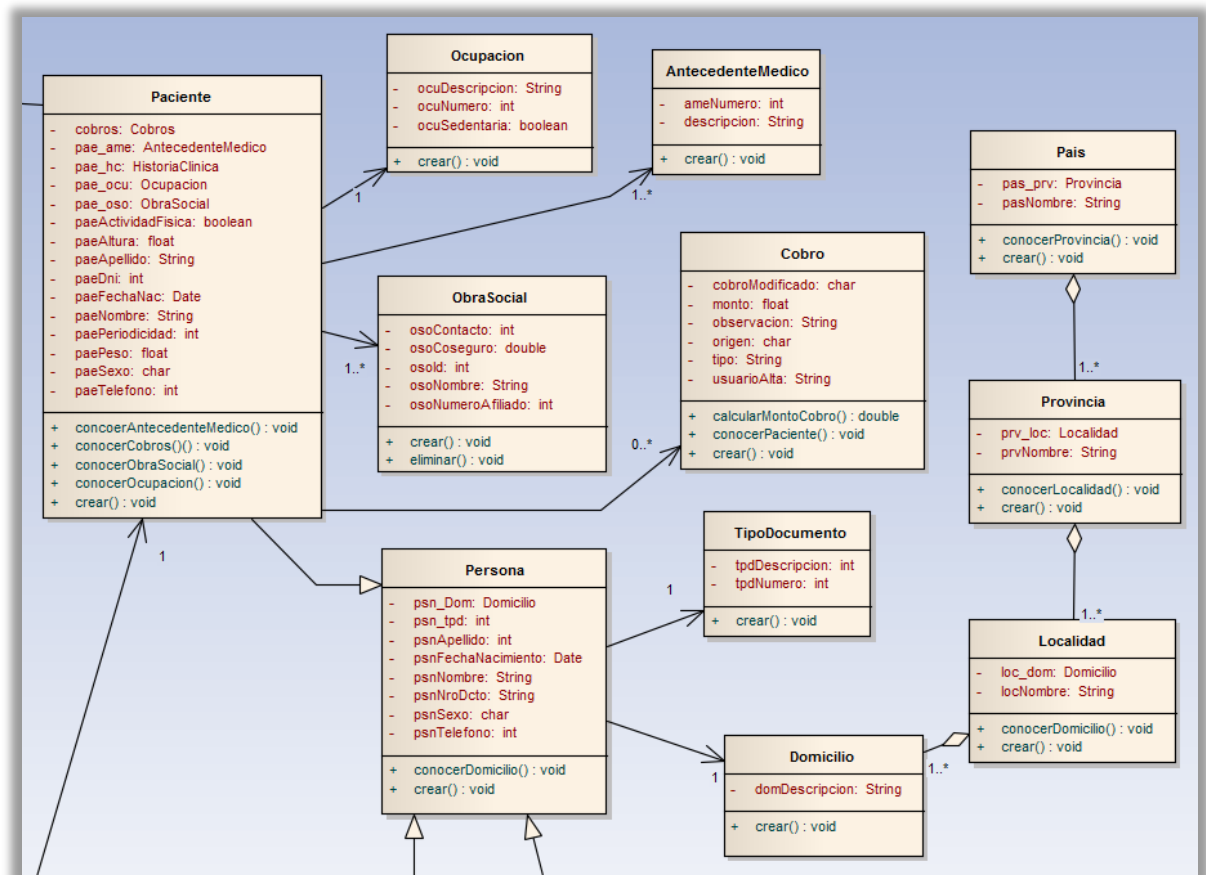


Visualización focalizada de clases

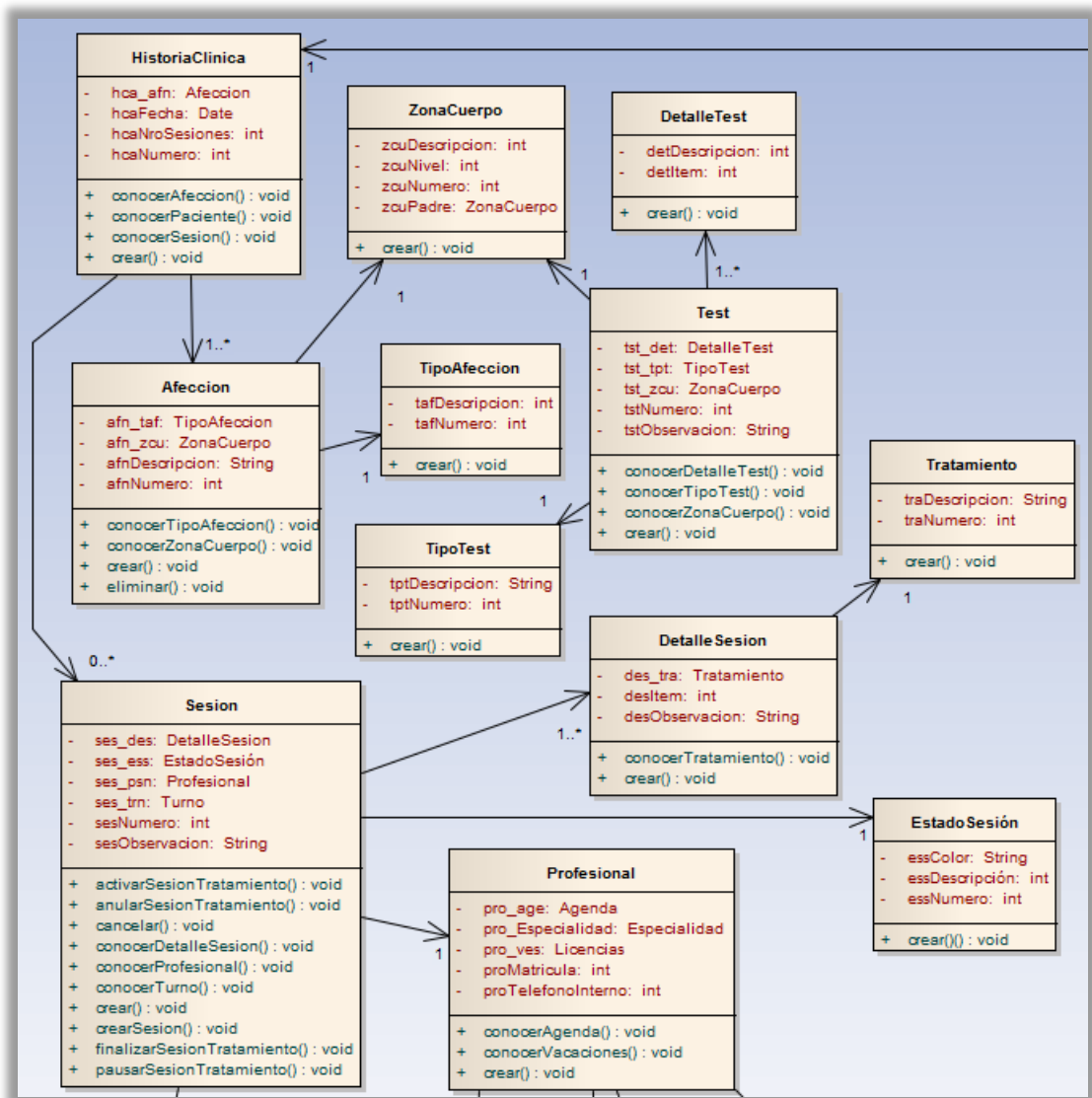
Gestión de Turnos:



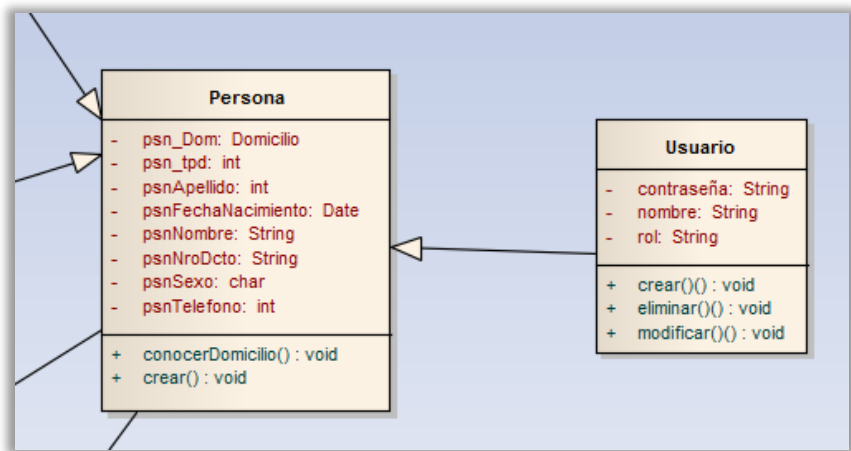
Gestión de Pacientes:



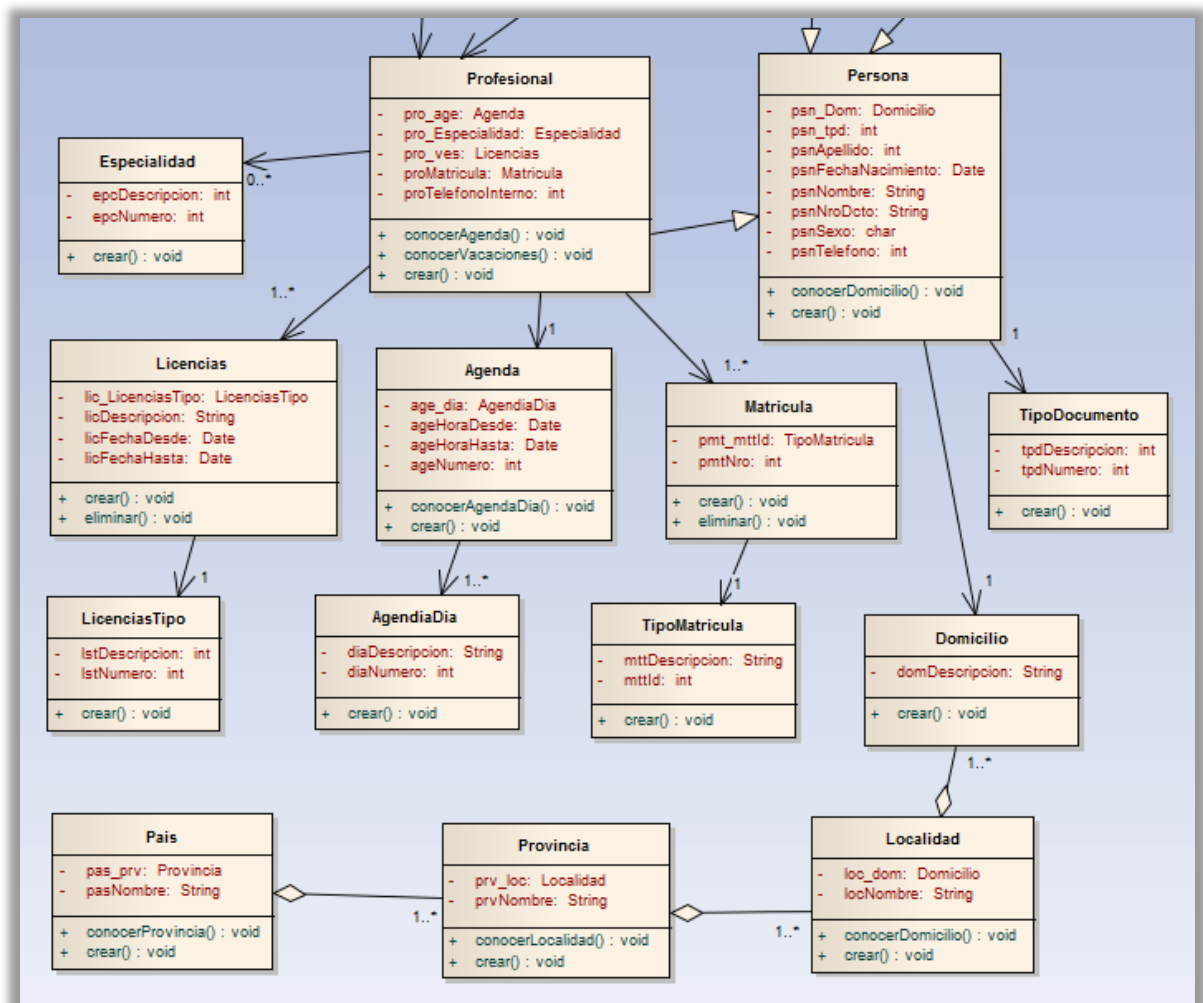
Gestión de Historia Clínica:



Gestión de Usuarios:



Gestión de Kinesiólogos:

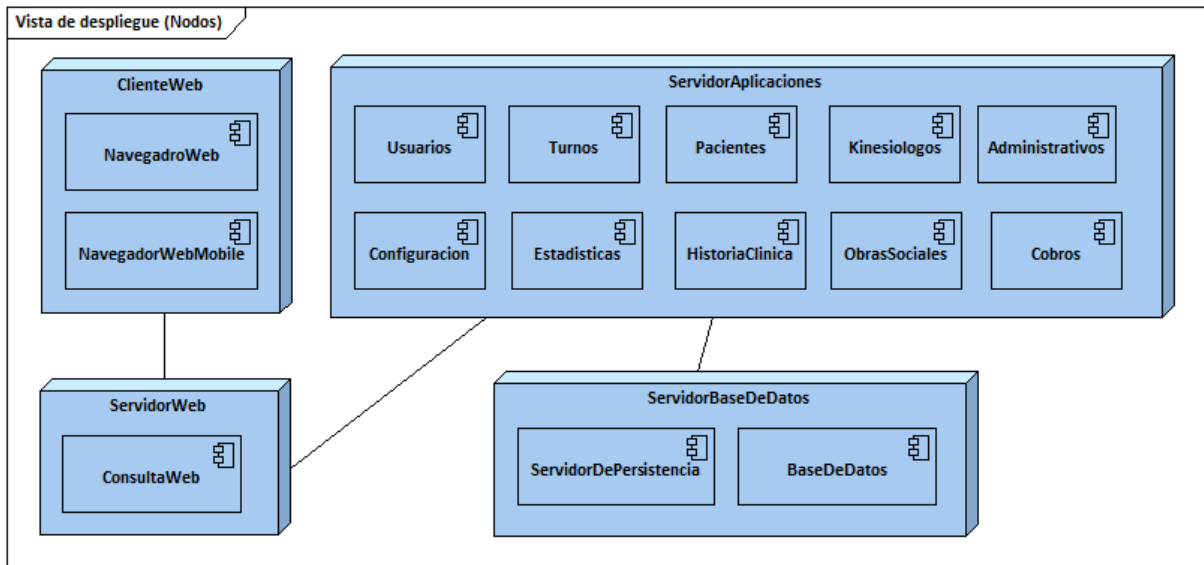


## 1.2.2 Fase de diseño

### 1.2.2.1 Vista de despliegue

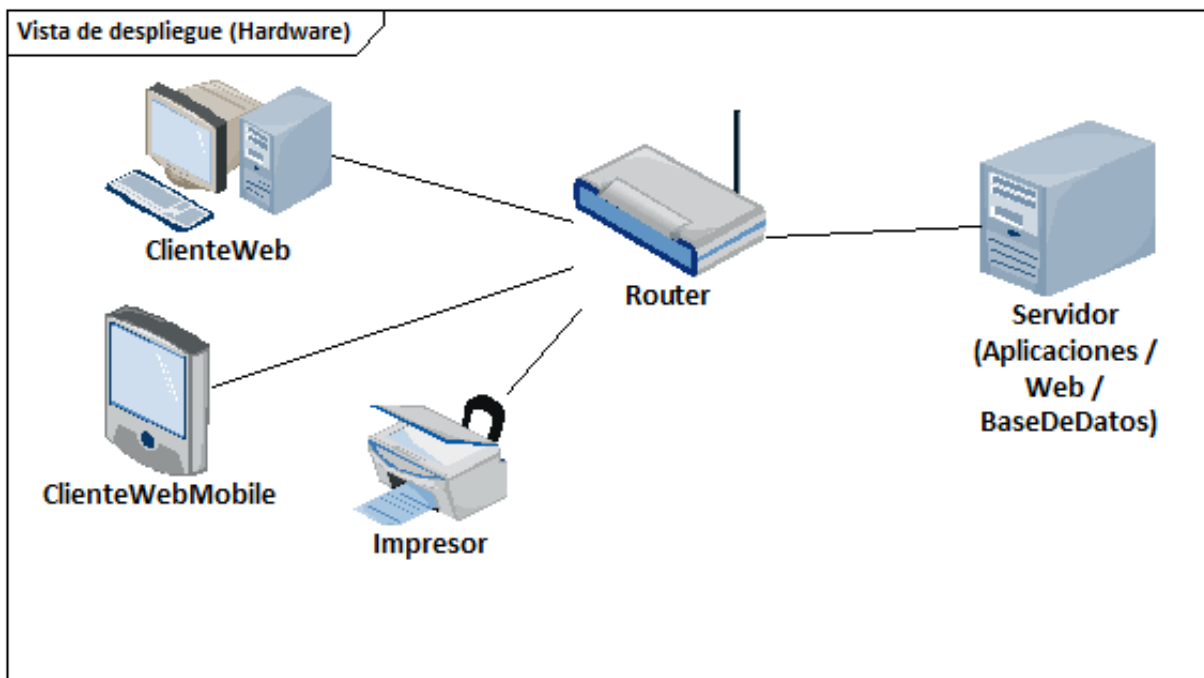
#### 1.2.2.1.1 Nodos

Vista que engloba la organización física de los diferentes componentes de software en nodos con los cuales estará conformado el sistema.



#### 1.2.2.1.1 Hardware

Disposición de la colección de Hardware necesaria para el despliegue del sistema.

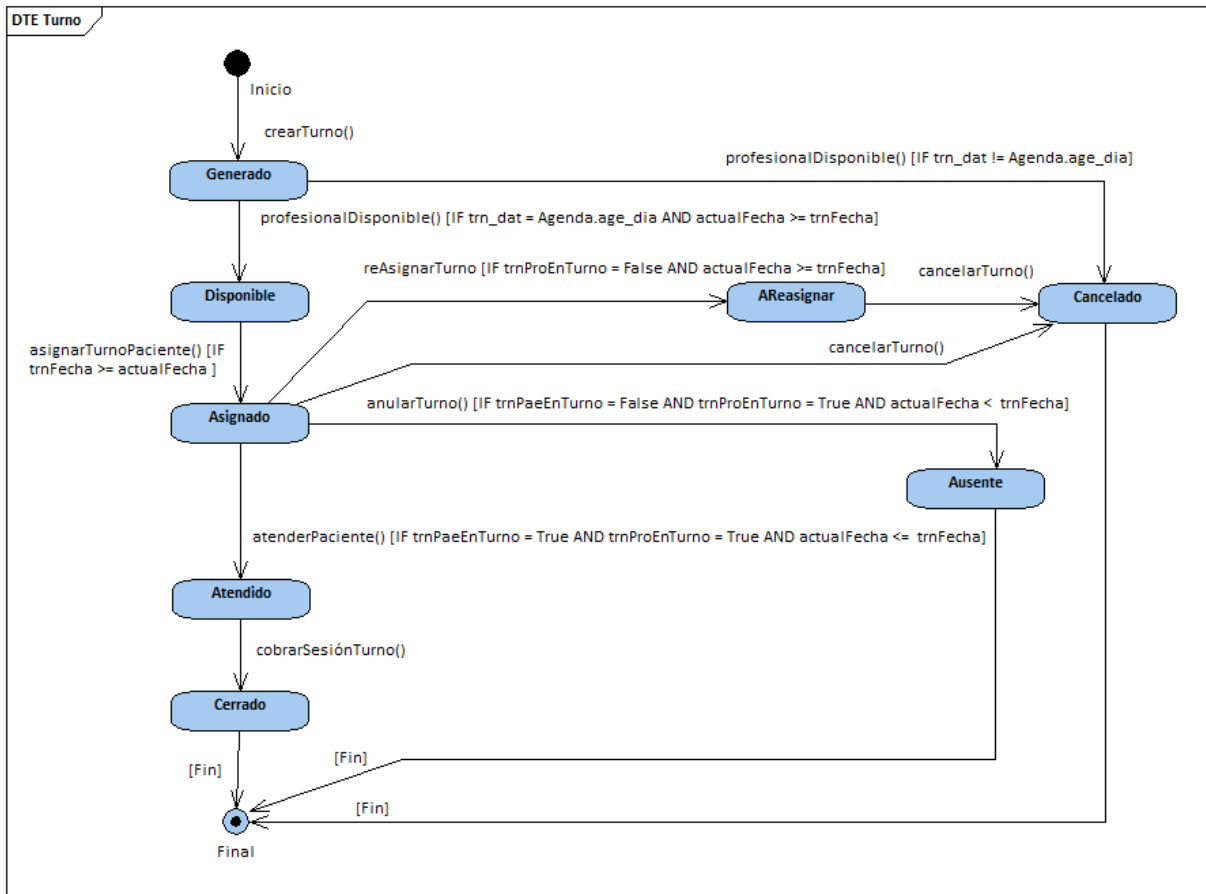




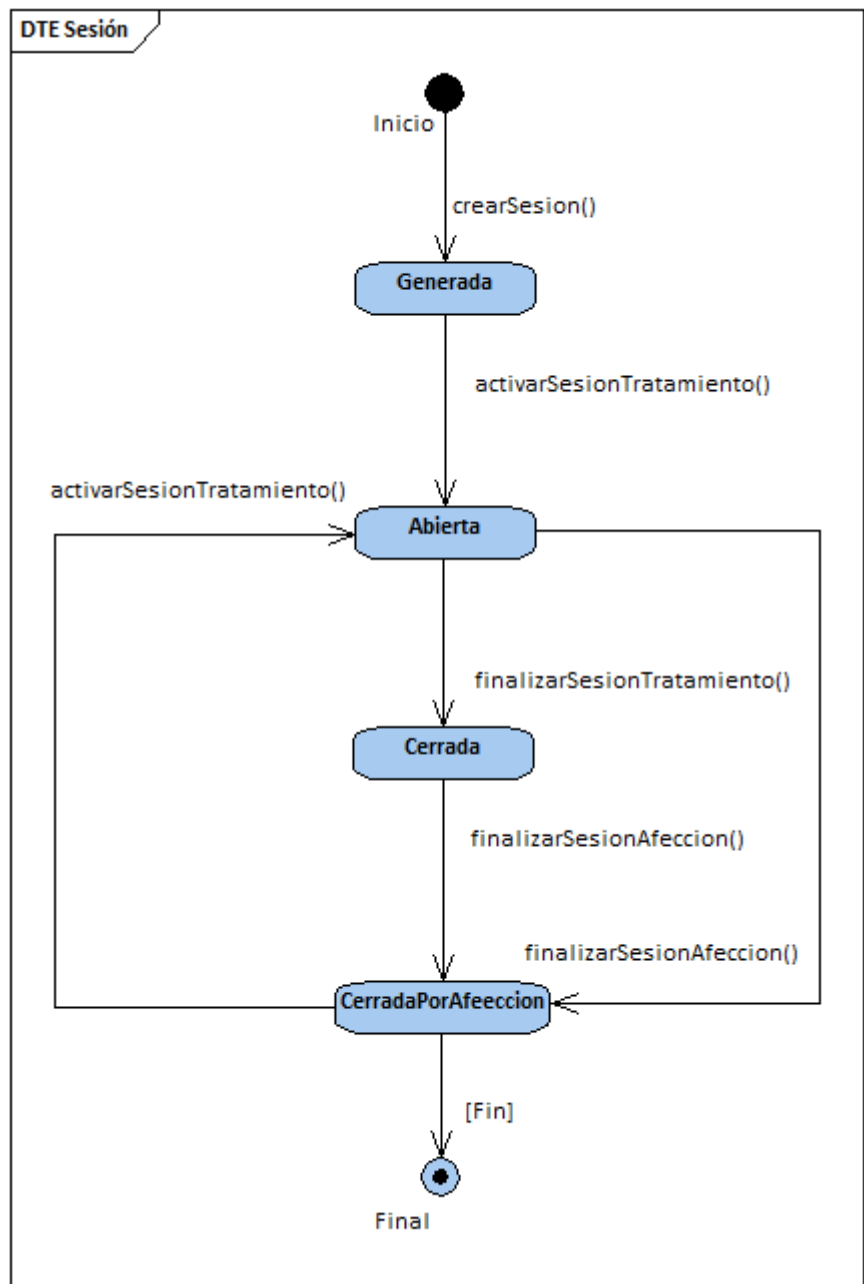
1.2.2.2 Vista dinámica

Diagramas de transición de estados (DTEs) elaborados acorde al reconocimiento de objetos identificados en los distintos estados en los que pueden establecerse:

Objeto identificado: 'Turno'.



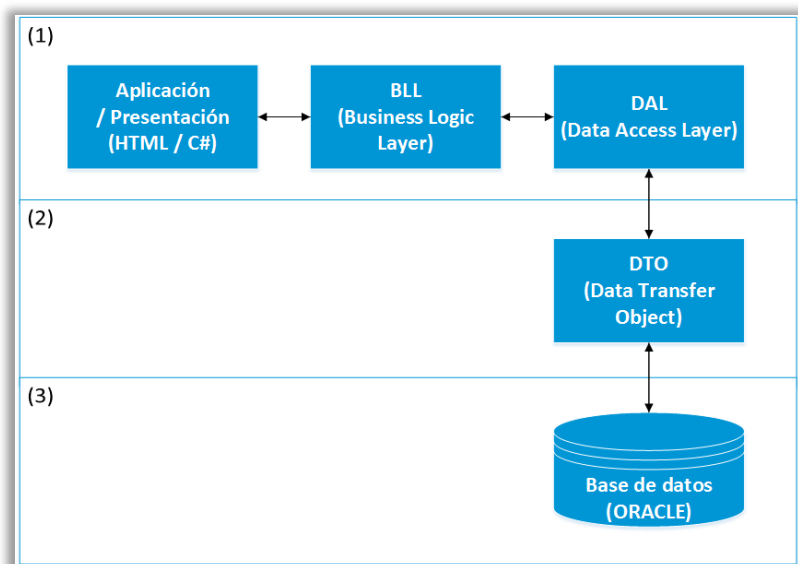
Objeto identificado: 'Sesión'



### 1.2.3 Fase de implementación

#### 1.2.3.1 Esquema de codificación del Sistema FISIKS

El código fuente de la aplicación se desarrolló en la herramienta 'Visual Studio Community 2013', mediante la programación por capas:



La programación por capas es una técnica de ingeniería de software propia de la programación por objetos, éstos se organizan principalmente en 3 capas: la capa de presentación o aplicación, la capa de lógica de negocio o control, y la capa de asociada datos.

Siguiendo esta técnica, se asegura el avance en la programación del proyecto de una forma ordenada, lo cual beneficia en cuanto a reducción de costos por tiempo, al ser dividida la aplicación general en varios módulos y capas que pueden ser tratados de manera independiente y hasta en forma paralela.

Por otra parte, otra característica importante de recalcar es la facilidad para las actualizaciones de la aplicación. En este aspecto, la programación en capas juega un papel de suma importancia ya que sigue un estándar conocido en el ambiente de desarrollo de aplicaciones, lo cual da al programador una guía para hacer mejoras a la aplicación sin que esto sea una tarea tediosa y desgastante, siguiendo el estándar establecido para tal fin y dividiendo las tareas en partes específicas para cada capa del proyecto.

En la programación en capas los objetos se dividen según su funcionalidad. Destacan tres principales: la Capa de Interfaz o de Aplicación, se compone por los objetos encargados de interactuar con el usuario, como lo son los formularios e interfaces de la aplicación. Por otra parte se encuentra la Capa de Lógica de Negocio o Control, en donde se encuentran los objetos que realizan la mayor parte del trabajo interno del programa, en esta etapa destaca la lógica de la aplicación así como la funcionalidad de servir de enlace entre las otras capas; por último se encuentra la Capa de Datos, integrada por los objetos que envían y obtienen información al comunicarse con bases de datos u otros sistemas de información que colaboran con el programa.

### 1.2.3.1.1 Capas del modelo

#### **1.2.3.1.1.1 Capa de Presentación o Aplicación:**

La presentación del programa ante el usuario, se maneja en interfaces que cumplen con el objetivo principal de este componente, el cual es facilitar al usuario la interacción con la aplicación.

La interfaz debe ser amigable y fácil de utilizar, ya que el usuario final se va a encargar de utilizar el sistema y de dar retroalimentación al equipo de desarrollo en caso de que haya algo que mejorar.

Las interfaces son consistentes con la información que se requiere, no se utilizan más campos de los necesarios, así como la información requerida está especificada de manera clara y concisa, y por último, las interfaces deben satisfacer los requerimientos del usuario.

Dentro de la parte técnica, la capa de presentación contiene los objetos encargados de comunicar al usuario con el sistema mediante el intercambio de información, capturando y desplegando los datos necesarios para realizar alguna tarea. Además, se ubican aquí todos los recursos y componentes utilizados para añadir funcionalidad y contenido a esta capa, presentando un repositorio de hojas de estilos, imágenes, hojas de funcionalidad, etc. Esta capa se comunica únicamente con la capa de Reglas de Negocio o Control.

#### **1.2.3.1.1.2 Capa de Lógica de Negocio o Control:**

Es llamada capa de reglas de negocio porque en esta se definen todas las reglas que se deben cumplir para una correcta ejecución del programa. Es aquí donde se encuentra toda la lógica del programa, así como las estructuras de datos y objetos encargados para la manipulación de los datos existentes, así como el procesamiento de la información ingresada o solicitada por el usuario en la capa de presentación.

Representa el corazón de la aplicación ya que se comunica con todas las demás capas para poder llevar a cabo las tareas. La aplicación se comunica con la capa de datos para obtener información existente o ingresar nuevos datos. Recibe los datos que ingresó el usuario del sistema mediante la capa de presentación, luego los procesa y crea objetos según lo que se necesite hacer con estos datos; esta acción se denomina encapsulamiento.

Al encapsular los datos, el programa asegura mantener la consistencia de los mismos, así como obtener información precisa de las bases de datos e ingresar en las mismas, solamente la información necesaria, asegurando así no tener datos duplicados ni en las bases de datos, ni en los reportes solicitados por el usuario.

**1.2.3.1.1.3 Capa de Datos:**

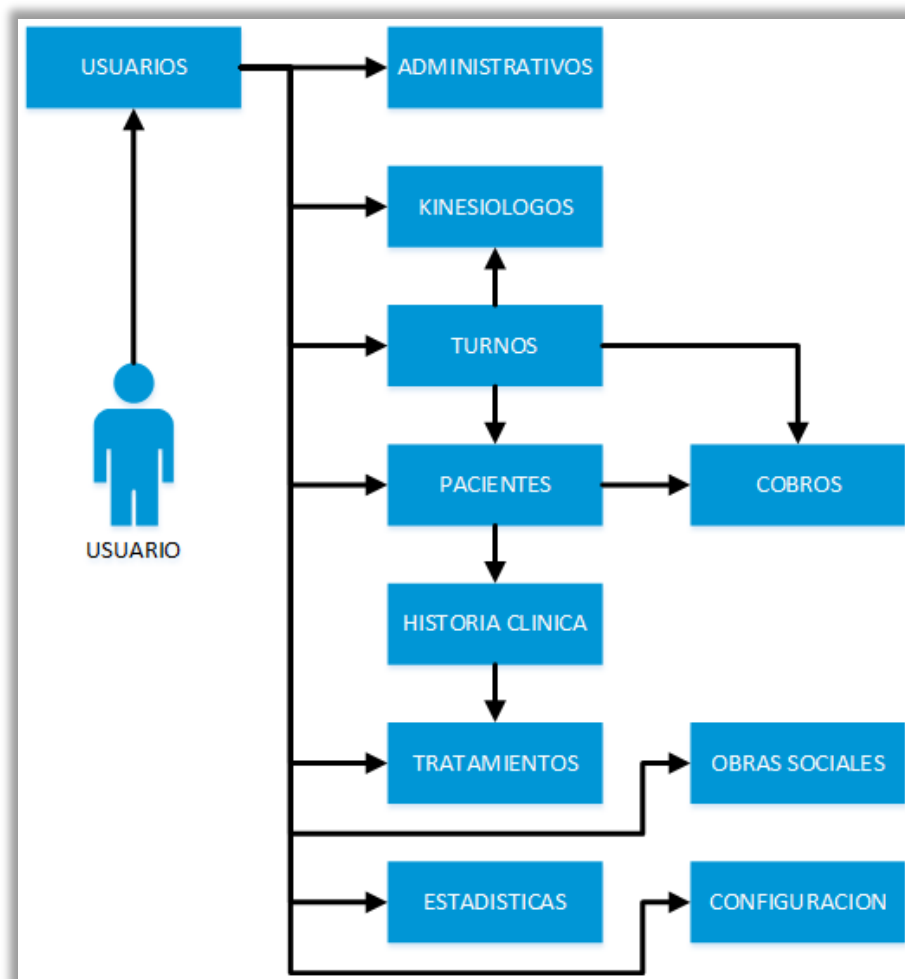
Es la encargada de realizar transacciones con la base de datos para almacenar y procesar información al propio sistema. El manejo de los datos se realiza de forma tal que haya consistencia en los mismos, con el fin de que los datos que se ingresan, así como los que se extraen de la base de datos, deban ser consistentes y precisos.

Es en esta capa donde se definieron las consultas a realizar en la base de datos, tanto las consultas simples como las consultas complejas para la obtención de datos más específicos y la creación de procedimientos que contemplan las funcionalidades, dejando así la lógica definida en la base de datos.

Esta capa envía la información directamente a la capa de reglas de negocio para que sea procesada e ingresada en objetos según se necesite. Esta acción se denomina encapsulamiento.

1.2.3.2 Módulos funcionales del Sistema FISIKS

La siguiente figura muestra la disposición general de los módulos del portal FISIKS y sus relaciones:



### **Configuración**

- Este módulo permitirá ingresar a las opciones de administración general y predeterminada de valores por defecto que definen el sistema. Algunos valores pueden ser: El valor monetario fijo de las sesiones a cobrar a pacientes; los horarios máximos y mínimos permitidos de agendas laborales; la tolerancia máxima de espera de un turno con ausencia de un paciente, etc.

### **Turnos**

- Módulo que reconocerá las opciones de asignación, modificación, cancelación y distintas formas de visualización de turnos de pacientes, de acuerdo a la administración de disponibilidad horaria y asignación de kinesiólogos. También se ofrecen las opciones de filtros de búsqueda de los mismos.

### **Pacientes**

- Módulo que permitirá las opciones de registración de pacientes, con sus datos personales, antecedentes médicos y su asociación con la información de Obras Sociales, como así también la búsqueda de los mismos.

### **Historia Clínica**

- Administración de historia clínica, registrando los detalles de la atención del paciente asociado en cada sesión, correspondiente a una afección en particular.
- Registro de sesiones de pacientes, con su respectiva evolución y asociación con la historia clínica, remarcando el tratamiento elegido para la afección correspondiente.

### **Tratamientos**

- Administrar búsquedas de los tratamientos asociados a diferentes afecciones con las que puede ingresar un paciente, mediante filtros de selección de áreas del cuerpo y tipo de afección.

### **Cobros**

- Administración de los ingresos por el cobro de las sesiones, registrando cada uno de ellos para control de los mismos, a fin de conocer los montos totales de recaudación por paciente. También se ofrecen las opciones de filtros de búsqueda de los mismos.

### **Kinesiólogos**

- Administración de profesionales, registrando datos personales, laborales y disponibilidad horaria, de modo que esta información brinde soporte a la gestión de turno. También se ofrecen las opciones de búsqueda de los mismos.

### **Usuarios**

- Control de acceso al sistema mediante la administración de cuentas usuarios, de acuerdo a los perfiles existentes en el contexto, con sus respectivos privilegios asociados.

### **Estadísticas**

- El sistema brindará opciones de emisión de estadísticas y reportes respecto a:
  - Cantidad de afecciones.
  - Afecciones por edad.
  - Afecciones por tratamientos.
  - Distribución de afecciones.
  - Comparación de tratamientos.

### 1.2.4 Trazabilidad del Producto

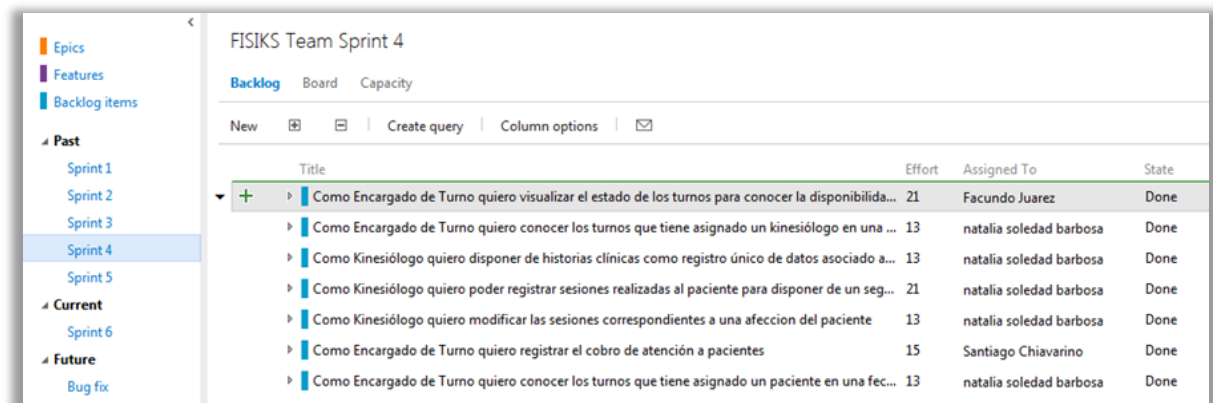
El concepto de trazabilidad dentro de nuestra gestión de proyecto representa el conocimiento continuo de la vida de un requerimiento, desde su concepción hasta la creación de diseños para el desarrollo, la verificación de la implementación y el cambio. En forma general, la definimos como la capacidad de rastrear el estado de los requerimientos en cualquier momento durante un proyecto.

La herramienta utilizada 'Visual Studio Online' permite generar y registrar relaciones de la historia de usuario con otros elementos de trabajo, tareas asociadas, prototipado de interfaces, casos de prueba, y versiones del código fuente, a fin de disponer de una trazabilidad integral de todos los elementos relacionados. Además, permite crear consultas para mostrar cualquier tipo de informe de trazabilidad. Las consultas también se pueden exportar o abrir en Excel, enviar a través de correo electrónico o abrir en Microsoft Project.

Desde las historias de usuario, pertenecientes al backlog, se pueden crear vínculos entre los elementos de trabajo, utilizando una de las pestañas de control de vínculos existente.

Cada pestaña está diseñada para permitir tipos de vínculos específicos y restringe los tipos de relaciones de vínculos creadas.

Navegando hacia una historia de usuario particular, podemos ingresar a la misma para visualizar en detalle su contenido.



Title	Effort	Assigned To	State
▶ + ▶ Como Encargado de Turno quiero visualizar el estado de los turnos para conocer la disponibilidad...	21	Facundo Juarez	Done
▶ Como Encargado de Turno quiero conocer los turnos que tiene asignado un kinesiólogo en una ...	13	natalia soledad barbosa	Done
▶ Como Kinesiólogo quiero disponer de historias clínicas como registro único de datos asociado a...	13	natalia soledad barbosa	Done
▶ Como Kinesiólogo quiero poder registrar sesiones realizadas al paciente para disponer de un seg...	21	natalia soledad barbosa	Done
▶ Como Kinesiólogo quiero modificar las sesiones correspondientes a una afección del paciente	13	natalia soledad barbosa	Done
▶ Como Encargado de Turno quiero registrar el cobro de atención a pacientes	15	Santiago Chiavarino	Done
▶ Como Encargado de Turno quiero conocer los turnos que tiene asignado un paciente en una fec...	13	natalia soledad barbosa	Done

A modo general, desde la pestaña de 'Detalles', se visualiza la descripción de la historia de usuario, sus criterios de aceptación, estado y detalles particulares.

Details (10)

Area: FISIKS Iteration: FISIKS\Release 1\Sprint 4

Add Tag

### Description

Reason: Work finished

Debe estar habilitada la opción de administración de turnos desde el modulo de turnos en cada una de las instancias que éste presente en calidad de las instancias de fragmentos de tiempos que se proponen: En cada día de un mes, en cada día de una semana y/o en cada horario habilitado en un día. Se debe implementar la estructura del turnero, disponiendo de datos de horarios, disponibilidad de kinesiólogos y acceso al alta de turnos.

ANEXO:

- [FISIKS - User Stories - Descripciones Generales](#)
- [FISIKS - Diseño](#)

### Acceptance Criteria

Para visualizar un turno y sus propiedades se debe permitir la disponibilidad de los siguientes datos:

KINESIÓLOGO:

- Apellido(s) (Alfa).
- Nombre(s) (Alfa).

\* El kinesiólogo a seleccionar debe estar previamente cargado en el sistema, habiendo completado historia de usuario #11 "Como Encargado del centro quiero registrar kinesiólogos".



Desde la pestaña ‘Historial de cambios’, se presenta el registro de todas las modificaciones que sufrió la historia de usuario, presentando un detalle de la transacción, la fecha de modificación y el usuario que la realizó.

Details ↻ ⌵ (10) 🔍

Area  
FISIKS
Iteration  
FISIKS\Release 1\Sprint 4

Add Tag

### State Graph

```

graph LR
    Start((New backlog item)) --> New[New]
    New -- "Approved by the Product Owner" --> Approved[Approved]
    Approved -- "Work finished" --> Done[Done]
    
```

### History

- ▲ LP Luciano Trillo Pellizzari *made field changes* (21/11/2015 1:20)
  - ▲ Fields
 

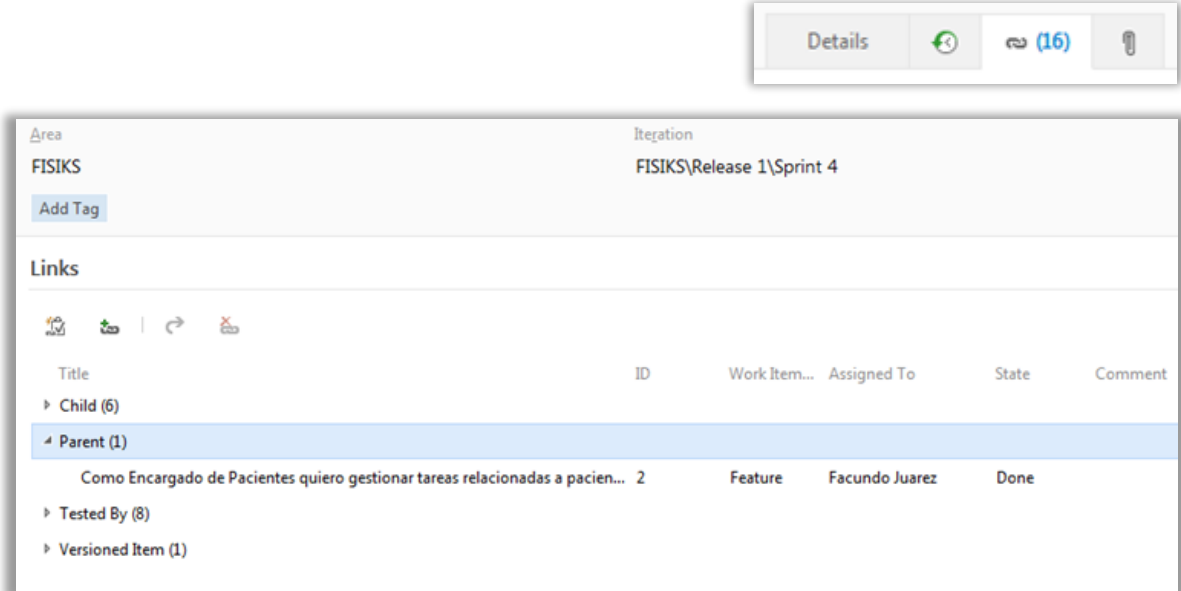
Field	New value
Rev	45

Debe estar habilitada la opción de administración de turnos desde el modulo de turnos en cada una de las instancias que éste presente en calidad de las instancias de fragmentos de tiempos que se proponen: En cada día de un mes, en cada día de una semana y/o en cada horario habilitado en un día. Se debe implementar la estructura del turnero, disponiendo de datos de horarios, disponibilidad de kinesiólogos y acceso al alta de turnos.

ANEXO:

- [FISIKS - User Stories - Descripciones Generales](#)
- [FISIKS - Diseño](#)
- [FISIKS - DTE: TURNO](#)

Desde la pestaña de 'Enlaces', podemos generar y registrar relaciones de la historia de usuario con otros elementos de trabajo, tareas asociadas, prototipado de interfaces, casos de prueba, y versiones del código fuente, a fin de disponer de una trazabilidad integral de todos los elementos relacionados.



The screenshot shows a software interface with a 'Links' section. At the top, there is a navigation bar with 'Details', a refresh icon, a link icon with '(16)', and a pin icon. Below this, the 'Area' is 'FISIKS' and the 'Iteration' is 'FISIKS\Release 1\Sprint 4'. There is an 'Add Tag' button. The 'Links' section contains a table with columns: Title, ID, Work Item..., Assigned To, State, and Comment. The table has a 'Parent (1)' row selected, which is expanded to show a single entry: 'Como Encargado de Pacientes quiero gestionar tareas relacionadas a pacien...' with ID '2', 'Feature' type, assigned to 'Facundo Juarez', and state 'Done'. There are also expandable sections for 'Child (6)', 'Tested By (8)', and 'Versioned Item (1)'.

Title	ID	Work Item...	Assigned To	State	Comment
▶ Child (6)					
◀ Parent (1)					
Como Encargado de Pacientes quiero gestionar tareas relacionadas a pacien...	2	Feature	Facundo Juarez	Done	
▶ Tested By (8)					
▶ Versioned Item (1)					

Con todo lo descrito anteriormente, podemos disponer de la trazabilidad de todas las historias de usuario existentes en el backlog.