

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**



PONTIFICIA  
**UNIVERSIDAD  
CATÓLICA**  
DEL PERÚ

**IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN QUE  
APOYE EL PROCESO DIARIO DE ELABORACIÓN DE  
CRONOGRAMAS DEL PERSONAL DE MEDICINA FÍSICA Y  
REHABILITACIÓN DE UN HOSPITAL**

Tesis para optar el Título de Ingeniero Informático, que presenta el bachiller:

**Jorge Manuel Márquez Redhead**

**ASESOR: Sofía Mercedes Dapello Sarango**

Lima, Julio del 2013

## Resumen

El presente proyecto consiste en el análisis, diseño e implementación de un Sistema de Información que apoye el proceso diario de elaboración de cronogramas del personal de medicina física y rehabilitación de un hospital, al mejorar la asignación de los pacientes a los tecnólogos, reduciendo los tiempos de espera para la primera cita de terapia, equilibrando a la vez la carga de trabajo de los tecnólogos, de tal forma que todos tengan más o menos el mismo número de sesiones programadas por mes, incluso cuando existan pacientes que desertan de sus tratamientos.

Para esto, es necesario realizar un cambio en el proceso actual del hospital estudiado, con un nuevo proceso que sea apoyado por la implementación de una herramienta de software.

Este cambio consiste básicamente en la eliminación de la necesidad de la negociación de un cronograma de atención entre el paciente y el tecnólogo que le ha sido asignado. En el nuevo proceso, la determinación de las citas se realiza mediante un algoritmo logrando que el paciente reciba sus terapias en tiempos y horarios pertinentes, así como también que los tecnólogos cuenten con horarios más equilibrados y no distraigan sus funciones asistenciales con labores relacionadas a la determinación de cronogramas de atención.

El presente trabajo ha sido estructurado en cinco capítulos, tal como se describe a continuación:

En el capítulo 1, se comienza con la definición de la problemática. Luego, se muestra los conceptos fundamentales que intervienen en el problema y ayudan a resolverlo. Además, se detallan los objetivos, los resultados esperados y la metodología a seguir. Después se define el alcance y las limitaciones. Por último, se explica el cronograma de las actividades del proyecto y los riesgos involucrados.

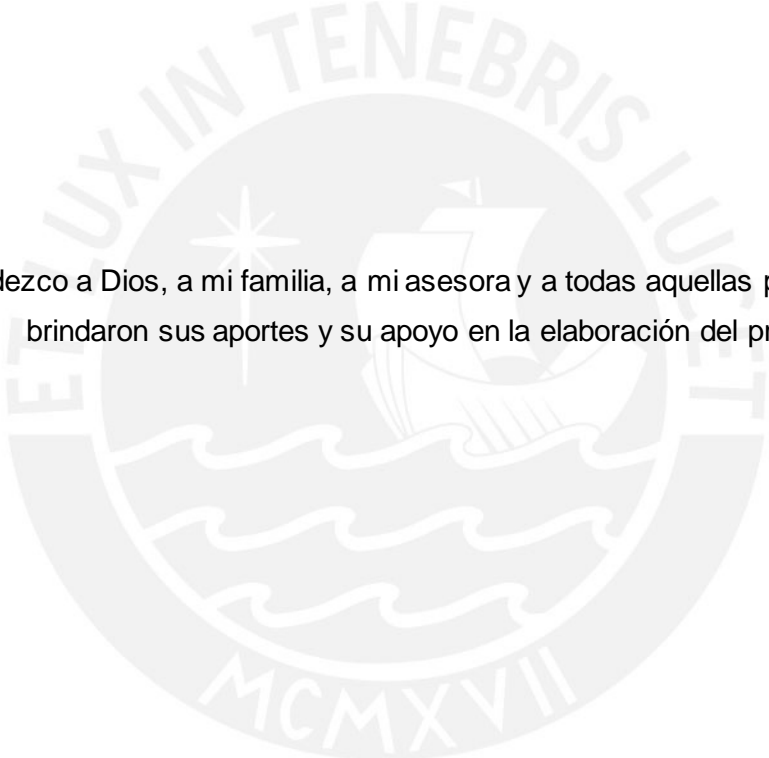
En el capítulo 2, se presenta el análisis de la solución a implementar, como los requerimientos que el sistema cumple.

En el capítulo 3, se desarrolla la definición de la arquitectura del sistema, la elección de las herramientas a utilizar, el diseño de la interfaz gráfica de usuario, así como el diseño del algoritmo.

En el capítulo 4, se explica la construcción del algoritmo de asignación de horarios y turnos óptimos; además, de parámetros necesarios para solucionar los diversas funcionalidades del sistema que puedan presentarse.

En el capítulo 5, se presentan las observaciones más importantes realizadas durante el trabajo, así como las conclusiones finales del presente proyecto, en los objetivos planteados.





Agradezco a Dios, a mi familia, a mi asesora y a todas aquellas personas que me brindaron sus aportes y su apoyo en la elaboración del presente proyecto.

## Tabla de Contenido

Introducción.....	1
1.Capítulo I: Generalidades .....	2
1.1 Problemática.....	2
1.2 Marco teórico .....	6
1.2.1 Los conceptos utilizados para la atención de pacientes en esta área de servicio de Medicina Física y Rehabilitación. ....	6
1.2.2 Conceptos relacionados a la propuesta de solución .....	12
1.3 Estado del arte.....	18
1.3.1 Formas aproximadas de resolver el problema.....	18
1.3.2 Productos comerciales disponibles para resolver el problema.....	19
1.3.3 Conclusiones sobre el estado del arte .....	22
1.4 Objetivo general .....	23
1.5 Objetivos específicos.....	23
1.6 Resultados esperados.....	24
1.7 Métodos, metodologías y procedimientos .....	24
1.7.1 Metodología para la gestión del proyecto .....	24
1.7.2 Metodología para la gestión del producto .....	26
1.8 Alcance .....	27
1.9 Limitaciones .....	28
1.9.1 Limitaciones del proyecto .....	28
1.9.2 Limitaciones de la solución .....	29
1.10 Viabilidad y justificación del proyecto .....	29
1.10.1 Justificación del proyecto.....	29
1.10.2 Viabilidad del proyecto.....	29
1.11 Plan de actividades .....	30
1.11.1 Estructura de descomposición del trabajo .....	30
1.11.2 Diagrama de Gantt.....	30
1.11.3 Plan de Riesgos .....	30
2. Capítulo II: Análisis.....	33
2.1 Metodología aplicada para el desarrollo de la solución.....	34

2.2	Identificación de requisitos .....	34
2.2.1	Requisitos funcionales .....	34
2.3	Identificación de actores principales .....	36
2.4	Casos de uso por paquete .....	37
2.4.1	Paquete de Administración.....	38
2.4.2	Paquete de Generador de cronograma.....	39
2.4.3	Paquete de Seguimiento .....	39
2.5	Características de los Usuarios .....	40
2.6	Suposiciones y dependencias.....	40
2.6.1	Suposiciones .....	40
2.6.2	Dependencias .....	40
3.	Capítulo III: Diseño.....	41
3.1	Definición de la arquitectura del sistema .....	41
3.2	Arquitectura de solución .....	42
3.3	Herramientas a utilizar .....	42
3.3.1	Python .....	43
3.3.2	MySQL (MySQL 2013).....	44
3.4	Modelamiento de la Base de Datos .....	45
3.5	Definición del diseño del sistema.....	46
3.6	Diseño de la interfaz gráfica.....	48
3.6.1	Registro de pacientes .....	48
3.6.2	Registro de personal.....	48
3.6.3	Registro de historia clínica.....	49
3.6.4	Maestro de medicamentos .....	49
3.6.5	Maestro de turnos .....	50
4.	Capítulo IV: Construcción y pruebas .....	51
4.1	Construcción.....	51
4.2	Implementación de estructuras de datos en memoria.....	52
4.3	Implementación del algoritmo .....	53
4.3.1	Pseudo-código .....	55
4.4	Prototipo del sistema.....	61
4.4.1	Menú de mantenimiento .....	61

4.4.2 Módulo de configuración de horarios .....	64
4.4.3 Módulo de generación de horarios .....	65
4.4.4 Generar archivo XML de inasistencia .....	65
4.4.5 Módulo de eliminación de citas .....	65
4.5 Pruebas.....	66
4.5.1 Pruebas unitarias .....	67
4.5.2 Pruebas de integración .....	69
4.5.3 Pruebas con el usuario final .....	69
5. Capítulo V: Observaciones, conclusiones y recomendaciones .....	74
5.1 Observaciones.....	74
5.2 Conclusiones .....	75
5.3 Recomendaciones.....	76
Referencias bibliográficas.....	78
ANEXO 1: Especificación de Requisitos de Software.	
ANEXO 2: Diagrama de Gantt del Proyecto.	
ANEXO 3: Descripción de las clases para el desarrollo del software.	
ANEXO 4: Descripción de las pantallas del software.	
ANEXO 5: Catálogo de pruebas.	

## Índice de Figuras

Figura 1.1 Organigrama estructural del hospital estudiado (MINSA MOF HSJC 2008) .....	3
Figura 1.2 Diagrama de flujo del personal del área de medicina física y rehabilitación. Elaboración propia.....	9
Figura 1.3 Diagrama de flujo de atención al paciente. Elaboración propia. ....	11
Figura 1.4 Interfaz gráfica inicial para asignar turnos (Ángel 2012). ....	19
Figura 1.5 Interfaces gráficas para asignar todos los turnos en un periodo de tiempo (Ángel 2012).....	20
Figura 1.6 Interfaces gráficas para asignar la cita de un paciente a un turno (Ángel 2012). ....	20
Figura 2.1 Diagrama de Actores del Sistema.....	37
Figura 2.2 Paquete de administración.....	38
Figura 2.3 Paquete de administración.....	39
Figura 2.4 Paquete de seguimiento.....	39
Figura 3.1 Diagrama Entidad Relación (básico).....	46
Figura 3.2 Diagrama Entidad Relación (desplegado). ....	46
Figura 3.3 Diagrama de los componentes descritos.....	47
Figura 3.4 Registro de pacientes.....	48
Figura 3.5 Registro de personal.....	49
Figura 3.6 Registro de historia clínica.....	49
Figura 3.7 Maestro de medicamentos.....	50
Figura 3.8 Maestro de turnos.....	50
Figura 4.1 Pantalla de mantenimiento de tablas.....	61
Figura 4.2 Pantalla de detalle de persona.....	62
Figura 4.3 Pantalla de lista de terapias.....	62
Figura 4.4 Pantalla de detalle de terapia.....	63
Figura 4.5 Pantalla de listado de turnos.....	63
Figura 4.6 Pantalla de detalle de turnos.....	64
Figura 4.7 Pantalla de configuración de horarios.....	64
Figura 4.8 Diseño de pantalla de generación de horarios.....	65
Figura 4.9 Pantalla de eliminación de citas.....	66
Figura 4.10 Pantalla de registro de personal.....	70
Figura 4.11 Pantalla del Workbench de MySQL mostrando el contenido de la Tabla Personal.....	71



Figura 4.12 Archivo de datos CIE10 en formato XML.....	72
Figura 4.13 Contenido de la Tabla CIE10 en MySql.....	72

### Índice de Tablas

Tabla 1.1 Tabla de horario de atención del paciente.Datos brindados por el servicio de Medicina Física y Rehabilitación del hospital en estudio. ....	12
Tabla 1.2 Comparación del software de generación de horarios para el personal hospitalario. ....	22
Tabla 1.3 Plan de Riesgos.....	32
Tabla 4.1 Relación de pruebas unitarias.....	69
Tabla 4.2 Tipos de prueba con el usuario final .....	70

### Índice de Scripts

Script 4.1 Pseudo-código para la generación de horarios.....	60
--------------------------------------------------------------	----

## Introducción

Los avances tecnológicos han brindado grandes beneficios en todos los campos de la actividad humana, y principalmente el uso de sistemas computarizados son cada vez más importantes en los entornos que tienen que ver con la medicina y el cuidado de la salud. Es por ello, existen múltiples aplicaciones que abarcan todos los aspectos tanto médico-asistenciales como administrativos y que son utilizados dentro de las instituciones de salud, sean estas públicas o privadas. Ejemplo de ellos, son las soluciones que existen en el mercado para la gestión de Historias Clínicas Electrónicas (HCE), Gestión y programación de recursos humanos, Gestión de camas hospitalarias (Bed management) y la gestión de hospedaje que se le proporciona al paciente.

También hay aplicaciones orientadas a mejorar la gestión de los recursos humanos, lo cual implica una mejora inmediata en la calidad de las prestaciones otorgadas a los pacientes. Generalmente, el personal asistencial reparte su actividad profesional entre varios centros asistenciales o inclusive con la práctica privada o independiente por lo que una programación adecuada y lo más exacta posible permitiría aprovechar al máximo estos recursos, de la misma manera se reducirían al mínimo los tiempos de espera antes de cada cita, así como entre citas, de los pacientes.

En el caso que nos ocupa, se trata del servicio de rehabilitación y medicina física de un hospital público. El problema consiste en la falta de una programación adecuada de cronogramas de atención de los tecnólogos que realizan las terapias. Esto ocasiona básicamente que la carga de trabajo de ellos no esté distribuida en forma uniforme y que el paciente tenga que esperar mucho tiempo para ser atendido; además, los tecnólogos no siempre realicen los tratamientos de su especialidad o experiencia y que cuando los pacientes abandonan el tratamiento, no se puedan reprogramar las actividades de los profesionales mencionados trayendo como consecuencia la generación de “tiempos muertos” para ellos y la consecuente pérdida económica para la institución.

De acuerdo a lo anteriormente mencionado, este proyecto de tesis aborda el tema específico de programación de cronogramas de los tecnólogos del servicio de rehabilitación y medicina física de un hospital público.

## 1. Capítulo I: Generalidades

Este capítulo muestra aspectos relacionados al contexto del problema. Comienza con la definición de la problemática para la planificación de citas diarias del personal del servicio de medicina física y rehabilitación de un hospital del Estado. Luego se muestran los conceptos fundamentales que intervienen en el problema y ayudan a resolverlo. Además, se detallan los objetivos, los resultados esperados y la metodología a seguir del proyecto. Después de explicar la esencia del proyecto se define el alcance y las limitaciones. Por último, se explica el cronograma de las actividades y los riesgos involucrados.

### 1.1 Problemática

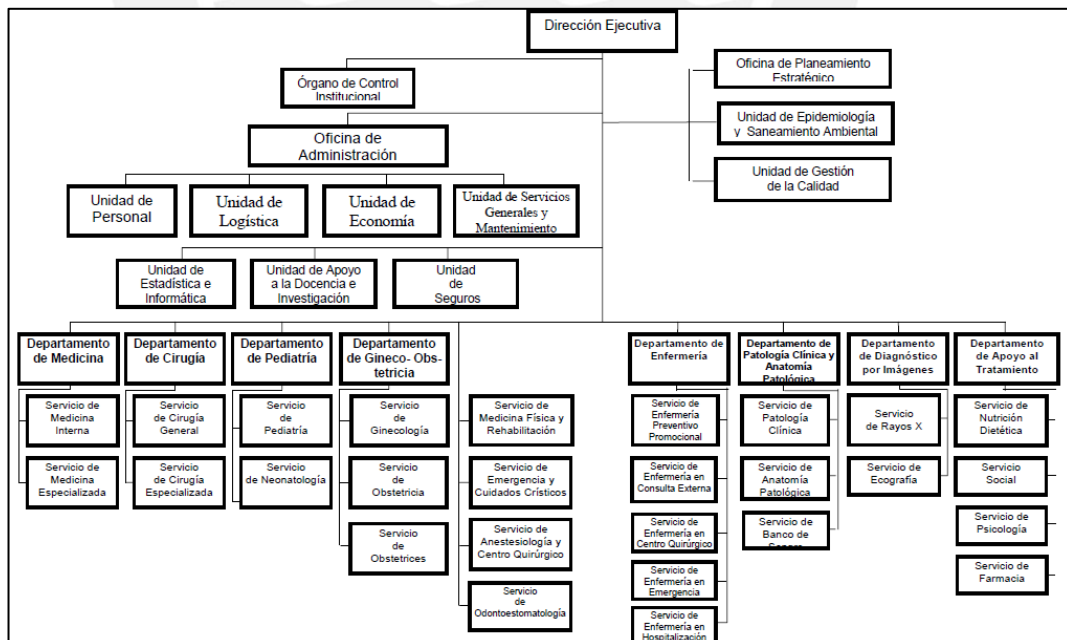
El contexto establecido como caso de estudio para desarrollar el presente proyecto es un hospital del Estado Peruano, el cual pertenece al Ministerio de Salud. El organigrama del hospital muestra que su Directorio se relaciona de forma directa

con ocho Departamentos y un servicio de Medicina Física y Rehabilitación (MINSA MOF HSJC 2008). Ver Figura 1.1.

Cualquier paciente externo, que no tiene historia clínica dentro del hospital público; o interno, accede a un servicio en el hospital por un monto de dinero, lo que también aplica al consultorio del área de medicina física y rehabilitación.

Al servicio de Medicina Física y Rehabilitación, asisten una población de pacientes niños y adultos con dolencias físicas, derivados de otros consultorios o de otros hospitales para atención médica y tratamiento de patología de lenguaje, terapia musculo esquelético, ocupacional y de secuelas neurológicas.

Antes de la consulta médica se pasa por admisión y caja del hospital, en la consulta se determina el diagnóstico y el cronograma del tratamiento a recibir que son generalmente 10 sesiones de terapias. Éstas quedan almacenadas en el área en forma escrita y un resumen va a la historia clínica. Cada vez que el paciente asiste a las sesiones debe presentar el cronograma de citas en caja y pagar un monto de dinero. Con el comprobante de pago y el cronograma de citas, se acerca al lugar de atención del tecnólogo para realizar su terapia.



**Figura 1.1** Organigrama estructural del hospital estudiado (MINSA MOF HSJC 2008)

Las sesiones de terapia indicadas por el médico para cada paciente son asignadas a un tecnólogo, tomando en cuenta su carga laboral de este último y las terapias que han indicado que prefieren. Se trata con esto que los tecnólogos tengan una carga de trabajo balanceada, y a pesar que todos ellos tienen la capacidad de realizar cualquier terapia, asignarles aquellas que les son preferidas, para contribuir de alguna forma, a un mejor ambiente laboral. Cada tecnólogo se encarga de elaborar el cronograma de tratamiento, utilizando la herramienta Excel, en el que se planifica los días y horas de las sesiones de terapia para un periodo máximo de quince días y según la disponibilidad del paciente sea para el turno de la mañana o de la tarde. Este proceso manual ocasiona demoras en la entrega del cronograma al paciente, y por la alta demanda, en la mayor parte de los casos esta elección del paciente en cuanto al turno no es tomada en cuenta, ya que el tiempo que debería esperar para acceder a sus terapias se prolongaría más allá de los quince días mencionados (MINSA MOF HSJC 2008).

Cada vez que el paciente asiste a una sesión de terapia, el tecnólogo anota en un formato impreso el día, la terapia que se le aplicó, el nombre del paciente, su documento de identidad y el código de la enfermedad (Clasificación Internacional de las Enfermedades CIE-10), que son parte del registro de información del Sistema Estadístico de Información del Ministerio de Salud.

Los trabajadores de este servicio (Medicina Física y Rehabilitación) han identificado los siguientes problemas:

Un problema es que se lleva un control manual deficiente que no permite un buen control de los pacientes que asisten o no asisten a sus terapias. Las inasistencias de pacientes dejan por tanto, horarios en los que los tecnólogos no tienen a quién atender, y se mantienen reservados cupos que podrían ser asignados a otros pacientes. La depuración se hace de forma manual y el tecnólogo nota el problema cuando el paciente viene después de 4 o más inasistencias para que se le haga otro cronograma, ya que cuando hay más de 3 inasistencias es necesario realizar un diagnóstico nuevo, y por lo tanto un cronograma de citas nuevo.

También ocurre que después de la depuración de citas existan tecnólogos que tienen poca carga laboral restante en comparación con otros, teniendo días libres. Cabe resaltar que cada tecnólogo recibe la misma remuneración mensual y pueden atender hasta una cantidad determinada de pacientes por día.

Otro problema es que el tecnólogo de triaje realiza la elección del tecnólogo en base a la observación de sus tablas de disponibilidad, pero no siempre se logra balancear la carga.

Además, si el tecnólogo con el que se va a organizar las citas no se encuentra en esos momentos, el paciente debe buscarlo en su horario de trabajo otorgado por el tecnólogo de triaje para que elabore su cita en el momento que se encuentre desocupado. Esto incomoda al paciente.

Por otra parte, si el paciente pierde su hoja de citas el tecnólogo se lo regenera con la ayuda del paciente, quien informa qué días ha asistido, y así puede buscarlo según las fechas indicadas en las fichas donde están las tablas de asistencia histórica ordenadas por fecha para determinar a qué citas acudió según su cronograma de horarios de Excel y su Tabla de asistencia en fichas de papel.

Se plantea la solución a los problemas planteados mediante un cambio en el proceso actual. El nuevo proceso estará soportado por un sistema de información que elabore el horario de citas de los pacientes según las sesiones que indique el médico y la disponibilidad mensual de cada tecnólogo según su rol de trabajo. Además, se debe considerar un balanceo en la carga de citas de cada tecnólogo en general. Cada paciente es asignado a uno o varios tecnólogos, tomando en cuenta sus cargas laborales y preferencias por determinado tipo de terapias. Cuando los pacientes alcancen un determinado número de inasistencias, las citas restantes deben ser eliminadas. El sistema de información debe brindar las alertas de estas circunstancias para que el personal administrativo se comunique telefónicamente con los pacientes que alcancen esta condición. Las citas no canceladas en la caja del hospital son las consideradas como no asistidas, por lo que el sistema de

información debe incluir la funcionalidad de carga de esta información desde el sistema de caja existente.

## **1.2 Marco teórico**

En esta parte se describe tanto los conceptos para entender el problema relacionado a los procesos en el área de la medicina física y rehabilitación, conceptos relacionados a la creación de citas, además de los conceptos relacionados a la optimización combinatoria y el problema de generación de horarios, conocido en la literatura como Timetabling.

### **1.2.1 Los conceptos utilizados para la atención de pacientes en esta área de servicio de Medicina Física y Rehabilitación.**

A continuación se van a definir los conceptos que son necesarios para llevar a cabo la atención de pacientes en esta área.

#### **1.2.1.1 Historia clínica**

Es el documento médico legal, en el que se registran los datos, de identificación y de los procesos relacionados con la atención del paciente, en forma ordenada, integrada, secuencial e inmediata de la atención que el médico u otros profesionales brindan al paciente (MINSA NTSGHC 2008).

#### **1.2.1.2 Codificación internacional de enfermedades CIE 10**

Codificación internacional estándar de enfermedades en su décima revisión (CIE 10 2012).

#### **1.2.1.3 HIS (acrónimo de Hospital Information System, sistema de información hospitalario)**

Sistema de información hospitalario, el cual tiene la finalidad de gestionar de manera eficiente y precisa la información que se genera en el ámbito hospitalario. El manejo de información médica automatizada puede mejorar significativamente la atención al paciente, reduciendo errores al acelerar el flujo de órdenes y resultados, y haciendo disponible una información más completa para la toma de decisiones (GARCÍA MUÑOZ 2010: I).

#### 1.2.1.4 Informe médico

El paciente solicita su informe a través de mesa de partes-previo pago-, documento que llega al servicio junto con la historia clínica. El informe médico consta de la siguiente información (MINSA MNP HSJC 2010):

- Nombre del paciente
- Edad
- Historia Clínica
- Diagnóstico
- Fecha de Informe
- Servicio de Medicina Física y Rehabilitación
- Breve Reseña de la Enfermedad
- Examen Clínico Preferencial
- Tratamiento
- Evolución
- Recomendaciones
- Firma y sello del médico.

#### 1.2.1.5 Normas del Médico Asistente

El médico asistente debe permanecer 4 horas diarias, durante las cuales se le programa atenciones de consulta. En estas consultas se realizan el diagnóstico y el tratamiento especializado farmacológico, terapia física, ocupacional, de lenguaje, tratamiento de secuelas neurológicas y procedimientos de rehabilitación, tales como infiltraciones, bloqueos paraespinales, desgatillamientos, manipulaciones, aplicación de láser, vendajes funcionales. Estas opciones de tratamiento pueden ser indicadas para ser realizadas en un conjunto de sesiones (MINSA MNP HSJC 2010).

#### 1.2.1.6 Normas del Tecnólogo Responsable

Las actividades del tecnólogo responsable son (MINSA MNP HSJC 2010):

- Coordinar la programación del horario de atención terapéutica de los pacientes junto con ellos.



- Realizar pruebas y aplicar métodos de tratamiento que use Tecnología Médica prescritos en el diagnóstico elaborado por el médico.
- Elaborar el informe de evolución en el tratamiento del paciente luego de cada serie de sesiones según indicación médica.
- Registrar el tratamiento recibido por el paciente en la ficha de tratamiento junto con la fecha del tratamiento, el primer nombre y apellido del paciente, además de su DNI.
- Realizar al final de cada terapia el llenado del HIS del Ministerio de Salud con la utilización de la codificación internacional de enfermedades CIE 10.

#### **1.2.1.7 Normas específicas dentro del servicio**

Se prohíbe la atención de pacientes en terapia que no hayan pasado consulta médica de rehabilitación previa. También, se encuentra prohibida la atención de pacientes sin recibo de atención expedido por el hospital. El horario de atención es desde las 8:00 am hasta las 7:30 pm de lunes a sábado (MINSA MNP HSJC 2010).

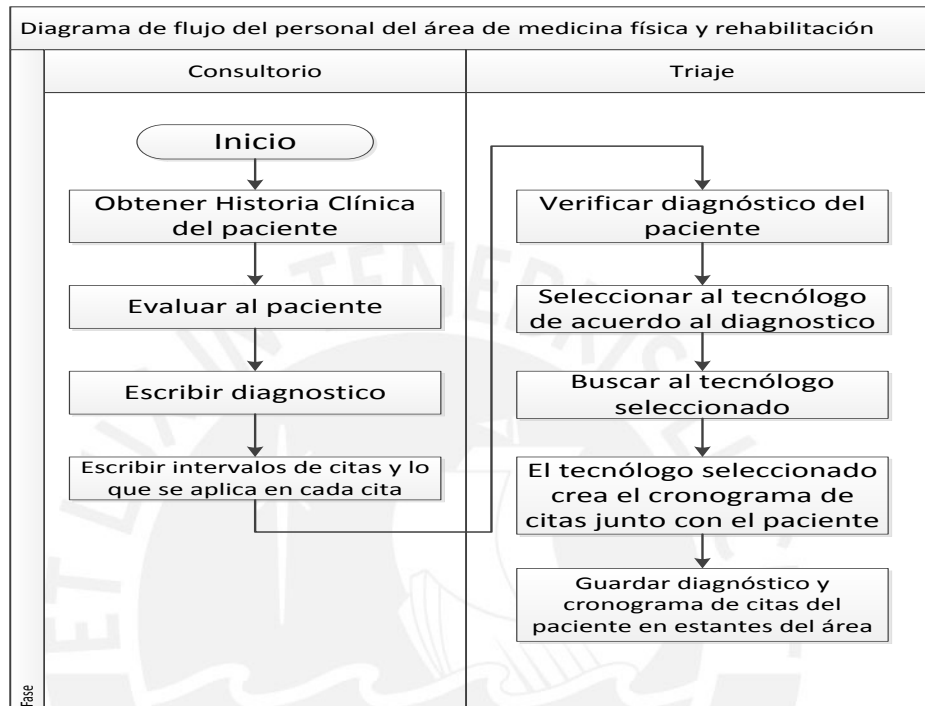
#### **1.2.1.8 Armado manual de horarios de sesiones de terapia**

Luego de elaborado el diagnóstico por el médico, el tecnólogo terapeuta de triaje, ubicado en el área de recepción de la misma área, decide de acuerdo al diagnóstico del médico cuántas sesiones son necesarias y busca en la tablas de disponibilidad mensual de horarios (archivo Excel), al tecnólogo con menor número de pacientes asignados para sesiones de terapia. En caso de encontrar dos o más tecnólogos con el mismo nivel de disponibilidad se elige al tecnólogo cuya preferencia tenga relación con el diagnóstico del paciente. Esta actividad podría ser realizada por el médico, ya que conoce las preferencias de los tecnólogos, pero desconoce los horarios disponibles. Una vez seleccionado, si el tecnólogo se encuentra en el hospital, se le solicita que se acerque a armar su cronograma de sesiones con el paciente. Si el tecnólogo se encuentra ocupado con una atención, el paciente debe esperar hasta que se desocupe. Si el tecnólogo no está en el hospital por no encontrarse en su turno de trabajo, se indica al paciente el horario en que puede encontrar al tecnólogo para que lo busque y elabore su horario junto con él.

Una vez organizadas las sesiones de terapia con el paciente y el tecnólogo, el tecnólogo le entrega un formato impreso al paciente, en donde se encuentran todos

los días de ese mes y del siguiente, en caso aplique, con la finalidad de marcar las fechas y las horas de las sesiones del paciente.

En la Figura 1.2 se muestra el proceso para elaborar el cronograma de horarios de terapia física y rehabilitación.



**Figura 1.2** Diagrama de flujo del personal del área de medicina física y rehabilitación. Elaboración propia.

Como se ha mencionado en la sección 1.1, Problemática, el objetivo de la solución propuesta es mejorar la asignación de los pacientes a los tecnólogos, reduciendo los tiempos de espera para la primera cita de terapia, equilibrando a la vez la carga de trabajo de los tecnólogos, de tal forma que todos tengan más o menos el mismo número de sesiones programadas por mes, incluso cuando existan pacientes que desertan de sus tratamientos y en general, tener un sistema más eficiente con una mejor utilización de los recursos. Para esto, es necesario realizar un cambio en el proceso actual, con un nuevo proceso que sea soportado en la implementación de la herramienta de software.

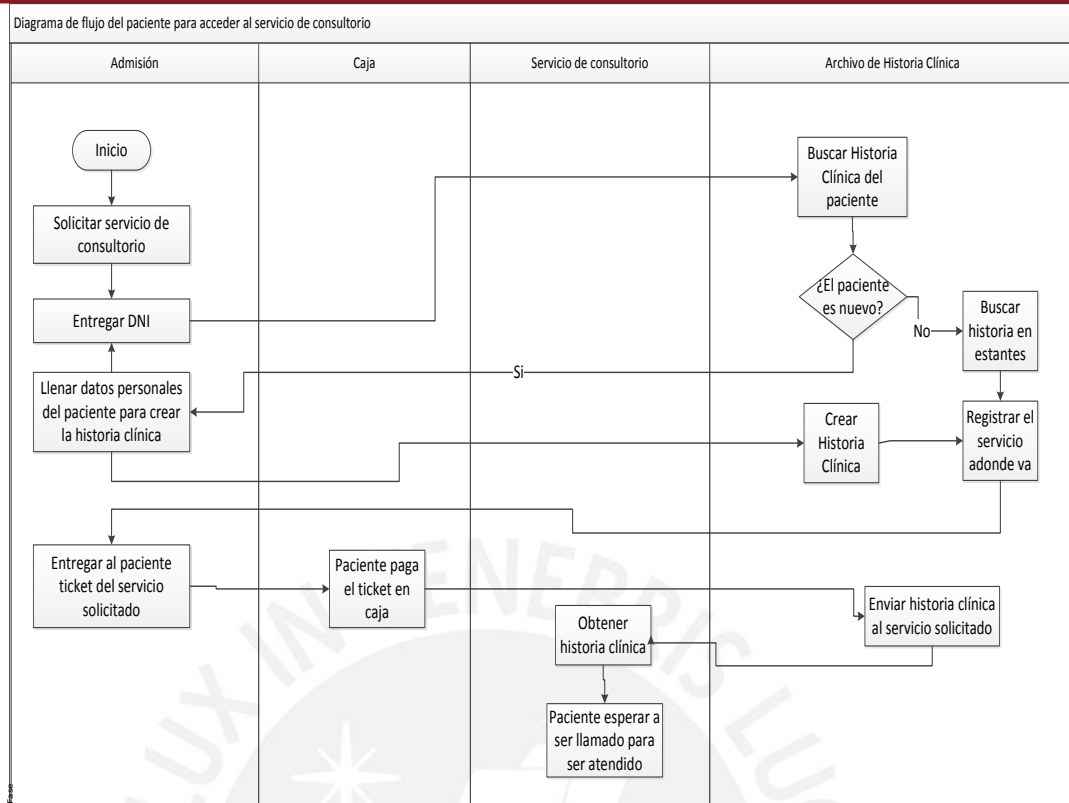
Este cambio consiste básicamente en la eliminación de la necesidad de la negociación de un cronograma de atención entre el paciente y el tecnólogo que le ha sido asignado. En el nuevo proceso, la determinación de las citas se realiza mediante un algoritmo que optimiza los aspectos contemplados en el párrafo anterior logrando que el paciente reciba sus terapias en tiempos y horarios pertinentes, así como también que los tecnólogos cuenten con horarios más equilibrados y no distraigan sus funciones asistenciales con labores relacionadas a la determinación de cronogramas de atención.

#### **1.2.1.9 Acceso del paciente a los servicios de terapia**

El sistema informático del hospital ha sido instalado este año (2013) y permite hacer el seguimiento de la historia clínica. La información contenida en estas historias clínicas no se encuentra digitalizada. Cuando se solicita cualquier servicio en el área de admisión del hospital, por el que es necesario pagar, las computadoras del área de historias clínicas reciben el estado del paciente. Si todavía no han pagado, al costado del código de su historia clínica junto con el nombre y el servicio al que va, aparece un indicador en color rojo. Tras un periodo, si el indicador continúa en rojo, el registro se retira de la lista. Si el paciente paga antes de la hora, el indicador pasa a color amarillo y el responsable busca las historias que están ordenadas por código y las retira. Como cada historia tiene asociado un código de barras, el encargado de trasladar la historia al servicio donde se requiere, pasa el código de barras a un lector para guardar automáticamente la hora de traslado. Una vez que el médico del servicio utilizó la historia como ayuda para hacer el diagnóstico y para agregar un informe del paciente de su propia evaluación, ésta regresa al área de historias clínicas donde se lee el código de barras para registrar que se encuentra nuevamente en el área de historias. En este momento, todos los indicadores de colores del sistema son borrados (MINSA MPP 2008).

#### **1.2.1.10 Diagrama de flujo del paciente**

En la Figura 1.3 se muestra el diagrama de flujo que sigue el paciente para acceder a cualquier servicio de consultorio, entre ellos, el de medicina física y rehabilitación.



**Figura 1.3** Diagrama de flujo de atención al paciente. Elaboración propia.

Este diagrama de flujo no se vería alterado como consecuencia de la mejora en el proceso que se ha implementado, pues el mencionado diagrama hace referencia a las etapas previas a la atención en el servicio de medicina física y rehabilitación.

### 1.2.1.11 Horario de atención al paciente

Actualmente (año 2013) se manejan los horarios que se muestran en la Tabla 1.1.

### 1.2.1.12 Disponibilidad de cubículos de terapia

La cantidad de cubículos de terapia física y rehabilitación son 4, cada cubículo es ocupado por un tecnólogo. Debido a que son máximo 7 tecnólogos por turno (mañana o tarde) hay conflictos para obtener cubículos, entre tecnólogos, mientras elaboran sus cronogramas de terapia junto con el paciente.

Cargo	Horario
Médico 1	De 8:00 am a 12 pm de lunes a sábado, excepto martes. Martes de 1:00 a 5:00 pm.
Medico 2	De 8:00 am a 12:00 pm, lunes, miércoles y viernes.
Tecnólogo 1	De 8:00 am a 2:00 pm de lunes a sábado.
Tecnólogo 2	De 1:30 pm a 7:30 pm de lunes a sábado.
Tecnólogo 3	De 7:00 am a 7:00 pm de lunes, miércoles y viernes.
Tecnólogo 4	De 1:30 pm a 7:30 pm de lunes a sábado.
Tecnólogo 5	De 7:00 am a 7:00 pm de lunes, miércoles y viernes.
Tecnólogo 6	De 7:00 am a 7:00 pm martes, jueves y sábado
Tecnólogo 7	De 1:00 a 7:00 pm de lunes a sábado
Tecnólogo 8	De 1:30 pm a 7:30 pm de lunes a sábado.

**Tabla 1.1** Tabla de horario de atención del paciente. Datos brindados por el servicio de Medicina Física y Rehabilitación del hospital en estudio.

## 1.2.2 Conceptos relacionados a la propuesta de solución

Se debe recordar que en el proceso propuesto el paciente ya no negocia su cronograma de atenciones con el tecnólogo que le ha sido asignado, sino que este será generado por el algoritmo de optimización que programa un determinado horario por una cierta cantidad de días, según lo indicado en su tratamiento, y con uno o varios tecnólogos.

### 1.2.2.1 Optimización combinatoria

Es una rama de las ciencias matemáticas y ciencias de la computación que estudia qué métodos se pueden realizar como operaciones, combinaciones y ordenamientos que son necesarios para generar una solución a un problema que utilice un conjunto de elementos. A veces las soluciones pueden tomar mucho tiempo computacional, por lo tanto, una forma de obtener una solución óptima o buena que satisfaga las restricciones dadas en el problema es mediante el uso de algoritmos heurísticos que utilizan diversas técnicas para conseguir los elementos que reducen el espacio de búsqueda que son evaluados para hallar la solución óptima. Además, de usar diversos métodos para resolver el problema en base a los elementos ya obtenidos (Díaz 1996).

### 1.2.2.2 Problema de asignación de horarios a empleados

El problema de asignación de horarios de los empleados (ETP) es la operación de asignación de tareas a los empleados que se realizan dentro de turnos durante un período de tiempo fijo. En un hospital la asignación de turnos de enfermeras, se ve como un conjunto de tareas que necesitan cumplir un conjunto de enfermeras, cada una con sus propias características, limitaciones y preferencias. Otras limitantes son las normas del hospital, como la reducción del costo total, o una división equitativa de trabajo entre los empleados. Algunos problemas similares incluyen la asignación de los trabajadores a las cajas registradoras en un supermercado, o la asignación de los operadores de telefonía a las estaciones de un servicio orientado al call-center. La tarea es realizada dentro de un cierto periodo de tiempo fijo llamado turno. Los horarios de empleados se refieren a un proceso de asignación de empleados a turnos donde cada turno involucra una o más tareas. En vez de asignar deberes a los empleados, cada día de la semana, se asigna tareas a los empleados por turnos, donde los turnos pueden existir en cualquier momento en el tiempo. También entran a tallar limitaciones debido a las regulaciones laborales. El problema de horarios del empleado consiste en la asignación (semanal) de los empleados a las tareas en turnos con hora de inicio y fin fijo (Schaerf).

### 1.2.2.3 Restricciones del problema de asignación de horarios a empleados

El presente proyecto plantea una modificación en el proceso de asignación de los tratamientos de los pacientes a los tecnólogos, que consiste en precisamente utilizar algoritmos de optimización que logren una programación óptima desde los puntos de vista tanto del paciente como del tecnólogo.

Se debe tomar en cuenta las restricciones a las que está sujeto un empleado que se le asigna una tarea en un turno (Schaerf). A continuación se explican los elementos fundamentales que se toman en cuenta para la elaboración del algoritmo y la generación de la solución final. Estas definiciones no son únicamente una construcción teórica sino que son sumamente importantes para la comprensión de la forma en que se generan los horarios óptimos.

**Requisitos:** cada turno  $S_j$  está compuesto de un número de tareas, algunas de ellas repetidas. Un empleado necesita ser asignado a cada tarea que pertenece al turno  $S_j$ . Esto es dado por una matriz no negativa  $R$ , llamada matriz de Requisitos,

de tal manera que  $R_{jk}$  dice el número de ocurrencias de la tarea  $T_k$  en el turno  $S_j$ . La suma de todas las tareas de un turno corresponde el número exacto de empleados que deben ser asignados en el turno  $S_j$ , ya que cada empleado hace una tarea.

**Capacidad:** cada empleado tiene habilidades que permitan cumplir con ciertos tipos de tareas, es decir, cada empleado  $E_i$  tiene un conjunto de tareas  $[T_{i1}, \dots, T_{in}]$  a las que puede ser asignado. La matriz de capacidad es una matriz binaria  $Q$ , tal que  $Q_{ik} = 1$  si el empleado  $E_i$  es capaz de hacer la tarea  $T_k$ , sino  $Q_{ik} = 0$ .

**Disponibilidad:** se refiere a que cada empleado sólo se le puede asignar ciertos turnos. Estas limitaciones están representadas por una matriz binaria de disponibilidad  $A$ , donde  $A_{ik} = 1$  si el empleado  $E_i$  está disponible para el turno  $S_k$ , sino  $A_{ik} = 0$ .

**Conflictos:** un empleado no puede ser asignado a más de una tarea en el mismo turno. Además, los empleados no pueden ser asignados a dos turnos que están en conflicto debido a que ocurren a la misma hora, en horas consecutivas, o porque no cumple con las normas del hospital. Los conflictos pueden variar para cada empleado, ya que cada uno tiene un contrato diferente, y se describen con una matriz tridimensional binaria de Conflictos de  $C_{nm}$ , de tal manera que  $C_{ijk} = 1$ , entonces, el empleado  $E_i$  no se puede asignar al turno  $S_i$  y  $S_j$ .

**Carga de Trabajo:** hay un límite superior e inferior en el número de tareas que a cada empleado se le puede asignar. Otro límite es asignar un número limitado total de tareas por horario y también un limitado número de asignaciones específicas que debe hacer el empleado. Por lo tanto, un conjunto de turnos  $G_1, \dots, G_s$ , donde cada conjunto especifica un conjunto de turnos. Luego se define dos matrices de enteros  $V_{ms}$  y  $W_{ms}$  tal que  $E_i$  empleado sea asignado por lo menos en los turnos  $V_{ik}$  y a lo máximo en los turnos  $W_{ik}$  del grupo  $G_k$ . El problema es encontrar cualquier asignación que satisfice todas las restricciones anteriores. Otras definiciones incluyen también restricciones de la función objetivo (a ser minimizada) y un justo balanceo de carga de trabajo para los empleados.

#### 1.2.2.4 Asignación dinámica de horarios

Los problemas de generación de horarios tienen la incertidumbre de qué citas van a aparecer. Para asignar solicitudes de citas de inmediato, antes de que otras solicitudes de asignación de horarios se conozcan, se propone una asignación en forma de plantilla dinámica, técnica que combina la optimización de asignar horarios antes que las solicitudes ocurran y hacer modificaciones al instante cuando no se llega a una solución óptima o cuando aparece una solicitud no esperada (Shoshana 2011).

En el caso de inasistencias de tecnólogos, se mantendrán las citas, pero se cambiarán los tecnólogos cumpliendo con las restricciones de carga de trabajo y capacidad. En el caso de eliminación de citas el turno y el tecnólogo quedan libres para ser reasignados cuando sea necesario.

Las restricciones presentadas en el problema de asignación de horarios a empleados y la forma como se plantea la asignación dinámica de horarios debido a que es similar a la asignación de horarios de citas de terapia física y rehabilitación de cada tecnólogo, se toma en cuenta para elaborar la plantilla de asignación de horarios de terapia física y rehabilitación debido a que tiene restricciones de capacidad que son las enfermedades en las que se especializa cada tecnólogo, de disponibilidad de horas asignadas a cada tecnólogo y que la carga de trabajo sea similar para cada tecnólogo. Además, de tener la incertidumbre del momento en que se generan nuevas citas.

#### 1.2.2.5 Búsqueda Local

Dado un caso  $p$  del problema de búsqueda  $P$  se le asocia un espacio de búsqueda  $S$  a él. Cada elemento  $s \in S$  corresponde a una solución potencial de  $p$ , y se llama un estado de  $p$ . Un elemento  $s \in S$  que corresponde a una solución que satisface todas las limitaciones de  $p$  es llamado estado factible de  $p$ .

Búsqueda local se basa en una función  $N$  la cual asigna a cada  $s \in S$  su vecindario  $N(s) \in S$ . Cada estado  $s' \in N(s)$  se llama un vecino de  $s$ . El vecindario depende de la estructura de  $P$  y se compone generalmente de los estados que se obtienen por algún cambio local, llamado movimiento, desde el estado actual. Se denota por



$s+m$  el vecino obtenido de la aplicación de  $s$  por un movimiento  $m$ . Una corrida de un algoritmo de búsqueda local se inicia a partir de un estado inicial  $s_0$ , obtenido con el algoritmo GRASP o generado al azar, y entra en un bucle que se desplaza por el espacio de búsqueda, paseando de un estado  $s_i$  a uno de sus vecinos  $s_{i+1}$ .

Las técnicas de búsqueda local se diferencian entre sí en las estrategias que utilizan para seleccionar el movimiento en cada estado y para detener la búsqueda. En todas las técnicas, la búsqueda es impulsada por una función de costo  $f$  que estima la calidad del estado. Para los problemas de búsqueda, una función de costo común cuenta el número de restricciones corrompidas, y el objetivo de la búsqueda es minimizar  $f$  a cero (para problemas de optimización  $f$  incluye también la función objetivo del problema) (Schaerf).

### 1.2.2.6 Hill Climbing

Entre técnicas de búsqueda local, se usa la técnica subir pendientes (*hill climbing*). Subir pendientes no es una técnica simple de búsqueda local, pero forma parte de la familia de técnicas basadas en la idea de la realización de movimientos únicos que mejoran o dejan sin cambios el valor del costo de la función  $f$ . La técnica de subida de pendientes se puede describir de la siguiente manera: en cada iteración  $i$ , "dibujar" el movimiento  $m_i$ , y si  $f(s_i + m_i) <= f(s)$  entonces dejar que  $s_{i+1} = s_i + m_i$  sino dejar  $s_{i+1} = s_i$ . Diferentes estrategias dibujan sus movimientos de manera diferente.

Como subir pendientes acepta movimientos laterales, la búsqueda no se detiene cuando alcanza un mínimo local. Con la finalidad de no tener un bucle infinito, se utiliza un criterio de parada basada en el número de iteraciones transcurridas desde la última mejora. Específicamente, dando al algoritmo el valor fijo  $n$  para que se detenga después de  $n$  iteraciones que no mejoran el valor de la función de costo, es decir, se detiene en  $j$  tal que la iteración  $f(s_j) = f(s_{j-1}) = \dots = f(s_{j-n})$ .

Para aplicar la subida de pendientes con ETP (Employee Timetabling problem) es necesario definir el espacio de búsqueda, la generación del vecindario, la función de costo, y la regla de selección para su solución inicial (Schaerf).

### 1.2.2.7 Espacio de búsqueda

Un adecuado procesamiento de la búsqueda local es una matriz rectangular de enteros  $M_{mn}$  con una fila para cada empleado. Cada columna representa un turno y

cada celda  $M_{ij}$  de la Tabla se le asigna un valor en el intervalo  $[0...t]$ . El valor  $M_{ij} = 0$  denota que el empleado  $E_i$  no trabaja en turno de  $S_j$ , y  $M_{ij} = k$  (con  $k > 0$ ) indica que el empleado  $E_i$  se le asigna la tarea  $T_k$  en el turno  $S_j$ .

Estas reglas de representación automáticamente descartan la posibilidad de que a un empleado se le asigne más de una tarea en el mismo turno. Además, se consideran sólo los estados (y movimientos) en los que los requisitos están siempre satisfechos, por lo tanto, los requisitos no necesitan ser revisados. Por el contrario, todos los demás tipos de restricciones pueden ser corrompidos y son incrustados en la función de costo (Schaerf).

#### 1.2.2.8 La generación del vecindario

Dado un movimiento **Reemplazar**: dado un turno  $S_h$ , y dos empleados  $E_i$  y  $E_j$ , de tal manera que  $M_{ih} = k$  (con  $k \neq 0$ ) y  $M_{jh} = 0$  en  $S$ , el estado que vino desde  $S$  obtenido después de aplicar el movimiento **Reemplazar**  $\langle h; i; j \rangle$  es igual a  $S$  con la diferencia de que  $M_{ih} = 0$  y  $M_{jh} = k$ .

Es decir, un empleado  $E_i$  hace una tarea  $T_k$  en un turno  $S_h$  es sustituido por otro empleado  $E_j$  que no trabaja en el turno de  $S_h$ . Un **Reemplazar** por lo tanto identificado con un  $\langle h; i; j \rangle$  que representa en el espacio de búsqueda el intercambio de la columna de valores  $M_{ih}$  y  $M_{jh}$  en  $h$ .

El **Reemplazar** no cambia el número de asignaciones para los pares turno-tarea, por lo tanto, cumple con las restricciones de requisitos (Schaerf).

#### 1.2.2.9 Técnicas de hill climbing

Dado un estado  $S$ , las reglas de selección que se aplican son las siguientes:

**RHC**: seleccionar al azar un **Reemplazar** $\langle h; i; j \rangle$  legal.

**RRSHC**: seleccionar un turno aleatorio  $S_h$  y un empleado  $E_i$  al azar, y encontrar  $E_j$  tal que  $f(S + \langle h; i; j \rangle)$  es un mínimo (sin hacer lazos aleatorios) entre todos los empleados.

**RSSH**: seleccionar un turno aleatorio  $S_h$  y encontrar  $E_i; E_j$  tal que  $f(S + \langle h; i; j \rangle)$  es mínimo (sin hacer lazos aleatorios) entre todos los pares de empleados.

**SHC**: encontrar  $m = \langle h; i; j \rangle$  tal que  $f(S + m)$  es mínimo (sin hacer lazos aleatorios) entre todos los movimientos **Reemplazar** legales. Se selecciona aquel que sus posibles soluciones tengan menos conflictos debido a experiencia previa en el problema que se va a resolver (Schaerf).

#### **1.2.2.10 Componentes de la función de costo**

La función de costo se define como el número total de restricciones incumplidas del estado actual. Todos los tipos de restricciones (capacidad, disponibilidad, carga de trabajo y conflictos) son, pues, implícitamente el mismo peso.

#### **1.2.2.11 Solución inicial**

La plantilla sobre la cual se asigna las citas de terapias. Esta plantilla contiene el horario de los tecnólogos, en el que cada turno puede estar libre, o asignado previamente a un paciente. Toma en cuenta las restricciones de asignación de citas a cada tecnólogo.

### **1.3 Estado del arte**

En la literatura revisada se ha encontrado información sobre algoritmos y toda la teoría involucrada que resuelve el problema de asignación de horarios que se ajustan al problema explicado. También, se ha buscado información de soluciones en el mercado para resolver el problema, aunque tenga ciertas limitaciones.

#### **1.3.1 Formas aproximadas de resolver el problema**

A continuación se presenta la información de los algoritmos útiles para encontrar soluciones para resolver el problema.

##### **1.3.1.1 Best- first searching**

Es un algoritmo de búsqueda local que parte de un elemento que es obvio que forma parte de la solución, evalúa los elementos que junto al elemento seleccionado cumplan con la función objetivo junto con las restricciones que conduzcan a obtener la solución y agrega al que da un mejor resultado para construirla. Se repite sucesivamente hasta formar la solución óptima que resuelva el problema planteado y que cumpla con todas las restricciones del problema (Dearden 2009).

##### **1.3.1.2 Branch and Bound Search**

Un algoritmo Branch and Bound busca en el completo espacio de soluciones para encontrar una solución exacta al problema dado. El uso de límites para la función a optimizar combinado con el valor de la mejor solución actual permite que el algoritmo busque las partes de la solución de forma implícita (CLAUSEN 1999).

### 1.3.2 Productos comerciales disponibles para resolver el problema

En esta sección se muestra los productos comerciales que intentan resolver el problema según la literatura revisada.

#### 1.3.2.1 Proyecto Ángel

Ángel es un software que implementa un Sistema Integral de Información Sanitaria. Este proyecto fue desarrollado por la empresa Argentina CONNMED, dedicada al desarrollo de diferentes soluciones médicas. Está desarrollado para ser una aplicación distribuida, en arquitectura cliente/servidor, en donde una sola base de datos almacena la información manejada por el sistema.

Para crear los turnos en que atiende un médico a cada paciente, primero se debe crear la agenda. Para reservar un turno se ingresa a la agenda, se selecciona en el calendario el día del turno, en la plantilla de turnos se selecciona el horario que se desea reservar, luego se selecciona al paciente a quien se quiere asignar el turno. Se puede incluir una nota aclaratoria si es necesario que se almacene junto con el turno. Sólo es posible asignar turnos en aquellos horarios, días, meses que configuró como válidos en las propiedades de la agenda (Ángel 2012).

En la Figura 1.4 se ve la Agenda antes de configurarla. Después de seleccionar cada uno de los ítems de la pestaña propiedad de la Figura 1.4 parecen las interfaces gráficas de la Figura 1.5.

Luego se organiza la agenda en pacientes, horarios, teléfonos en los intervalos que se establecieron en cada turno como se muestra en la Figura 1.6.

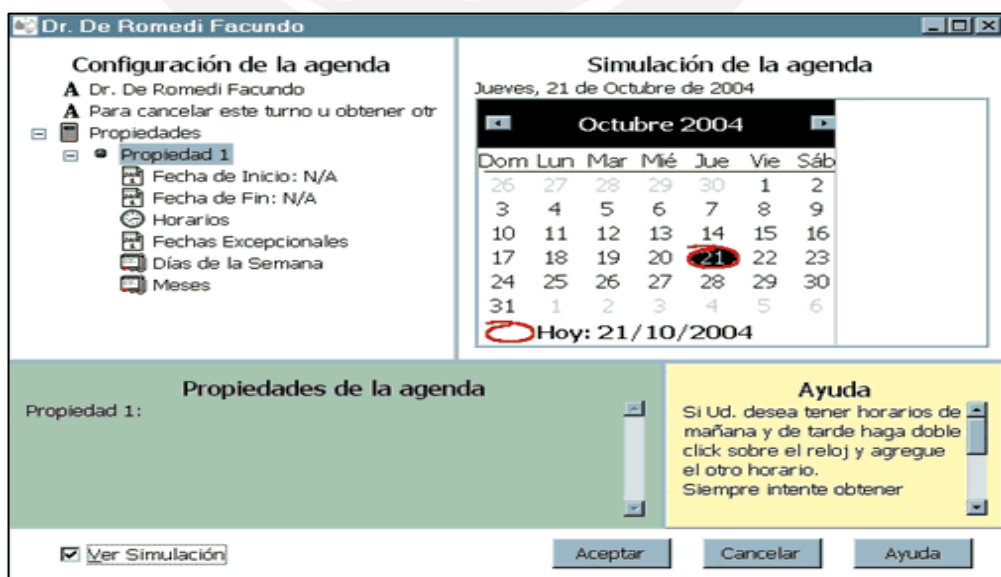
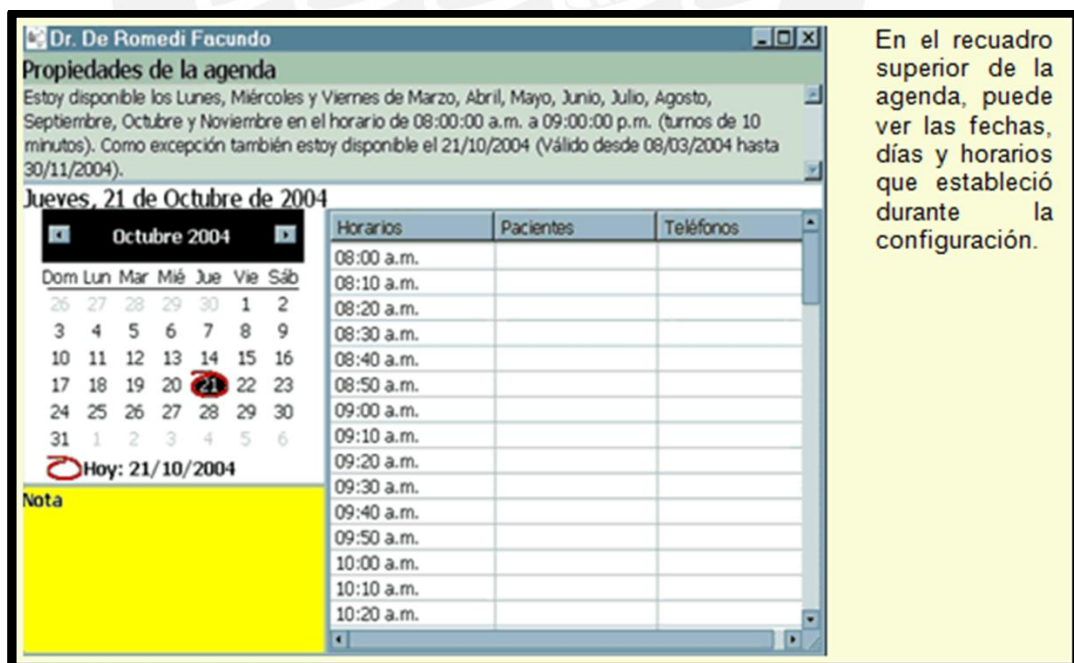


Figura 1.4 Interfaz gráfica inicial para asignar turnos (Ángel 2012).



Figura 1.5 Interfaces gráficas para asignar todos los turnos en un periodo de tiempo (Ángel 2012).



En el recuadro superior de la agenda, puede ver las fechas, días y horarios que estableció durante la configuración.

Figura 1.6 Interfaces gráficas para asignar la cita de un paciente a un turno (Ángel 2012).

Puesto que este producto requiere la interacción entre el tecnólogo y el paciente podría ser una solución adecuada a la situación actual y dado que además es un software ofrecido sin costo, podría ser una opción al proceso vigente, más no al proceso optimizado que se quiere lograr con este trabajo.

### 1.3.2.2 HospiSalus

Es un software para hospitales que tiene un módulo de citación que permite un control completo de la programación de consultas externas, así como la planificación de salas y demás recursos. Posee una interfaz equivalente a Outlook, es decir, se crean citas para registrar el tiempo que toma la tarea. Además se puede organizar una agenda a partir del esquema organizativo del hospital, disponiendo de una serie de códigos visuales mediante los cuales se distingue el estado del paciente en cada momento (llegada al centro, en consulta, salida). El sistema de agendas está concebido para agilizar al máximo el trabajo del personal de atención telefónica, ya que programa citas en base a una agenda. Adicionalmente, permite la creación y gestión de listas de espera en las citas (QSoft 2012).

Como en el caso del producto anterior, este software no resolvería la situación óptima que se busca en este proyecto.

### 1.3.2.3 OPTIHPER

Es un sistema para la asignación automática de horarios y tareas al personal de una empresa, teniendo en cuenta el conjunto de tareas a realizar, el personal disponible y su calificación, restricciones y preferencias existentes. Se aplica en los procesos de Asignación Optimizada de Personal en muy diversos entornos (OPTIHPER 2012):

- Empresas de distribución comercial y grandes superficies.
- Asignación de tripulaciones y personal de apoyo en entornos de transporte (aéreo, terrestre y marítimo).
- Asignaciones de Personal en hospitales, Centros Docentes, etc.
- Procesos de fabricación y producción, Logística, etc.

La Tabla 1.2 muestra la comparación de software de generación de horarios del personal hospitalario.

	Proyecto Ángel	Hospisalud	OPHITER
Utiliza agenda de disponibilidad de fechas y horas para cada personal en salud.	Si	Si	No
Utiliza un algoritmo para la generación de horarios en base a restricciones y objetivos.	No	No	Si
Da una solución especializada para la creación de horarios de acuerdo a un entorno de trabajo.	No	No	No
Es un producto comercial.	Si	Si	Si
Visualización de horarios en tablas.	Si	Si	Si
Utiliza rango de horas y minutos para crear horarios.	Si	Si	No
Utiliza rango de horas para crear horarios.	Si	Si	Si

**Tabla 1.2** Comparación del software de generación de horarios para el personal hospitalario.

### 1.3.3 Conclusiones sobre el estado del arte

La mayor parte del software presentado como estado del arte muestra que el proceso de citas se soluciona necesariamente con el personal que participa en las citas. No es de forma automática, por lo que el personal invierte mucho tiempo en la creación de agenda de citas. También, son genéricos y tienen una forma de trabajar no adaptable a los requisitos específicos del hospital en estudio, ya que

éste tendría que realizar muchos cambios en sus sistemas de información actuales, y probablemente también las políticas del servicio con que trabaja.

El presente proyecto de fin de carrera usa una plantilla de horarios vacía como solución inicial y el algoritmo hill climbing para la asignación óptima de horarios de pacientes.

#### 1.4 Objetivo general

Implementar un sistema de información que apoye en el proceso de elaboración de los cronogramas de terapias, permitiendo la planificación automática y eliminación de citas de pacientes que desertan a su tratamiento, para el área de rehabilitación y medicina física de un hospital.

#### 1.5 Objetivos específicos

- **Objetivo específico 1:** Definir las variables y estructuras que utiliza el algoritmo de búsqueda local, que permitan generar los horarios de cada tecnólogo.
- **Objetivo específico 2:** Establecer la función de mérito que tome en cuenta las restricciones del problema: capacidad, disponibilidad de los tecnólogos, carga de trabajo equilibrada y conflictos.
- **Objetivo específico 3:** Implementar un algoritmo Hill Climbing para resolver el problema de programación dinámica de horarios para citas de terapia física y rehabilitación
- **Objetivo específico 4:** Definir los formatos para la importación de datos para la modificación de horarios, respecto a las inasistencias de tecnólogos y de pacientes (proveniente del sistema de caja).
- **Objetivo específico 5:** Diseñar una interfaz gráfica que permita ingresar las restricciones del problema, y generar, así como, almacenar una o más planificaciones.



## 1.6 Resultados esperados

- **Objetivo específico 1:** Estructuras de datos que permitan manejar los datos de pacientes, citas, tecnólogos y horarios, para que puedan ser utilizados por el algoritmo Hill Climbing.
- **Objetivo específico 2:** Función de mérito utilizada por el algoritmo.
- **Objetivo específico 3:** Algoritmo de generación de horarios de terapia física y rehabilitación tomando como base el algoritmo Hill Climbing.
- **Objetivo específico 4:** Formato de archivo XML para la importación semanal de datos de pacientes con un límite de inasistencias y las inasistencias confirmadas por cada tecnólogo.
- **Objetivo específico 5:** Interfaz gráfica en la que el médico pueda ingresar las restricciones, además de aceptar o generar nuevamente los horarios.

## 1.7 Métodos, metodologías y procedimientos

En este punto se muestra las metodologías que se gestionan el proyecto de fin de carrera y para desarrollar el producto, los mecanismos para desarrollar los resultados esperados y por ende, alcanzar los objetivos específicos de proyecto.

### 1.7.1 Metodología para la gestión del proyecto

Para este proyecto personal, se requiere de metodologías que permitan la culminación a tiempo y generen entregables que benefician la elaboración del proyecto. Por eso, el proyecto toma en cuenta algunas áreas de conocimiento de las buenas prácticas del PMBOK y algunos puntos para que su aplicación se adecue a la gestión del presente proyecto de fin de carrera, como son:

### **Gestión del Alcance del Proyecto.**

La Gestión del Alcance del Proyecto define y controla los procesos para completar el proyecto satisfactoriamente. En la fase de planificación del Alcance se crea un plan de gestión del alcance del proyecto que refleje cómo se crea y define la Estructura de Desglose del Trabajo (EDT). En la Figura 1.7, se muestran los principales entregables y tareas del proyecto.

### **Gestión del Tiempo del Proyecto**

Se ha planteado que la duración del proyecto sea menos a un año. Según se observa en el mencionado EDT, esto sería posible siempre y cuando se verifique el cumplimiento a tiempo de las tareas definidas, en particular aquellas que se encuentran en la ruta crítica, por lo que se establece un seguimiento continuo a estas y la asignación de los recursos necesarios para su logro.

### **Gestión de los Costos del Proyecto**

Es deseable que se considere el efecto de los costos de uso, mantenimiento y soporte que surge como resultado de la implementación del proyecto. En este sentido, es necesario considerar los costos de cada actividad dentro de lo planeado, para no incrementar los costos relacionados a la implementación.

### **Gestión de la Calidad del Proyecto**

Los procesos de Gestión de la Calidad del Proyecto incluyen todas las actividades de la organización ejecutante para que el proyecto satisfaga las necesidades por las cuales fue ejecutado con alguna política de calidad. Aplica el sistema de gestión de calidad utilizando políticas, procedimientos y procesos de planificación de calidad, y como se controla la calidad, con actividades de mejora continua de los procesos que se realizan durante todo el proyecto, lo que trae como beneficio el cumplimiento de estándares de calidad que son necesarios para un correcto desempeño del proyecto y no haya deficiencias en la realización de actividades improductivas. En este proyecto se gestiona la calidad documentando todas las fases del proyecto como son la concepción para la toma de requisitos, el análisis de la concepción, el diseño, la construcción y pruebas, ya que se ofrece un producto

que cumple con los requerimientos dados, se podrá hacer mejora continua debido a que todos los procesos se encuentran documentados.

### **Gestión de las Comunicaciones del Proyecto**

Para el presente proyecto será de utilidad identificar a los usuarios, el cronograma de reuniones con ellos y definir los requisitos del proyecto de forma clara. Se trata de comunicar las mejoras esperadas con el cambio aplicado en el proceso y la utilización del sistema de información con el fin de vencer cualquier resistencia al cambio que se presente.

### **Gestión de los Riesgos del Proyecto**

Los riesgos que se pueden encontrar en este proyecto de tesis son variados siendo los más probables los de factor externo, ya que dependen del entorno donde se va a implementar.

Asimismo, se establecieron planes de contingencia en caso de fallas en el cronograma de actividades del proyecto (Institute Project Management Institute 2004).

No se utilizó el área de la Gestión de Recursos Humanos del Proyecto, debido a que es un proyecto personal, no fue necesario asignar tareas a diferentes personas, solo una persona estuvo encargada de desarrollar las tareas del proyecto.

Tampoco se tomó en cuenta el área de Gestión de la Integración del Proyecto, debido a que la gestión para cada área se hizo de manera secuencial, y no hubo muchas modificaciones a lo largo de la elaboración del proyecto.

#### **1.7.2 Metodología para la gestión del producto**

En este punto se explica, cómo algunos puntos de la metodología RUP (Rational Unified Process) servira para el desarrollo del presente proyecto.

Se consideró algunos puntos de RUP como metodología para la gestión del producto porque se enfoca en entender lo que el usuario necesita del sistema. En este proyecto, el usuario va a tener interacción con el sistema para proporcionar como entrada las restricciones con las que va a trabajar, cuando se genera el cronograma de citas de terapia física y rehabilitación, cuando reprograma el cronograma de citas, también cuando se muestra la información al paciente, al tecnólogo y al médico. Se utiliza el documento ERS, que es el documento de especificación de requisitos del software. Con el se define lo que debe hacer el software (Anexo 1) para entender lo que el usuario necesita del sistema, se hizo el modelado de los procesos a automatizar y del sistema. También sirve para definir cómo los usuarios utilizarán el sistema de manera clara. Esto es importante debido a que se tendrá un mejor alcance en cuanto al software que se va implementar y definir cómo va a funcionar.

Se utiliza el modelo de arquitectura del software para explicar los elementos básicos que necesitará el sistema (Kruchten. 2001), como son la vista de casos de uso del Capítulo 2 que explica mejor la funcionalidad del sistema, en el Capítulo 3 se explica la vista lógica para definir los elementos que va a utilizar cada componente del sistema, la vista de despliegue para indicar los elementos físicos que va a usar el software, y la vista de implementación, que muestra los elementos más importantes dentro del software sin operatividad con elementos externos.

Para hacer pruebas respecto a la funcionalidad del software se utiliza el documento de casos de prueba para comprobar que el sistema reciba y devuelva resultados correctos .

## 1.8 Alcance

El presente proyecto se relaciona con los procesos de atención del servicio de medicina física y rehabilitación de un hospital de atención pública, donde hay deficiencias en la asignación de citas a los pacientes, a cargo de los tecnólogos del área de terapia física y rehabilitación.

El proyecto consiste en la implementación de un sistema de información que permita optimizar el tiempo invertido en la generación de horarios de citas de terapia física. Permite también la eliminación de citas en caso de una cantidad de inasistencias por parte del paciente. También se reasignarán horarios en caso de que los tecnólogos falten en algún periodo de tiempo.

La información proveniente de otras fuentes, para eliminación y reasignación de horarios, serán proporcionados al sistema con archivos XML dados como entrada para el sistema, cuya carga se realiza mediante una interfaz sencilla con la que el personal no debería tener ninguna complicación técnica. Por el lado de la generación de los datos, también debe realizarse con un procedimiento totalmente transparente para el usuario final. Cabe resaltar que la propuesta intenta mejorar la situación descrita en la problemática a través de un sistema de información para el área de terapia de medicina física y rehabilitación, de tal manera que la percepción del usuario externo sea positiva, así como también el tiempo de espera sea menor y la productividad del personal se incremente.

Por lo expuesto este proyecto permitiría una mejor distribución de citas con respecto a la situación actual, tanto en términos del tiempo de espera del paciente antes de acceder a su primera cita, como de la utilización de los recursos de hospital, disponibilidad, horario y especialidad por cada uno de los tecnólogos que respondan a las necesidades del servicio.

## **1.9 Limitaciones**

Se tomaron en cuenta las siguientes limitaciones para el presente proyecto.

### **1.9.1 Limitaciones del proyecto**

Actualmente el proceso no se encuentra adecuadamente documentado pero aun así se lleva a cabo en forma diaria.

Por haberse utilizado software libre, no se consideraron limitaciones en cuanto al presupuesto para el desarrollo de esta solución. El resultado del proyecto va a ser publicado con una licencia de software libre (GLP) para que pueda ser utilizado por este hospital y otros que le encuentren utilidad.

### 1.9.2 Limitaciones de la solución

Al ser un proyecto orientado a un hospital del MINSA y no disponer de un presupuesto para la utilización de licencias de productos comerciales, se ha optado por la utilización de software libre en todos los niveles de la arquitectura. Se han elegido las herramientas con mayores aplicaciones en la actualidad y que por lo tanto también disponen de la mayor cantidad de fuentes de soporte disponible.

## 1.10 Viabilidad y justificación del proyecto

A continuación se analiza la viabilidad y justificación del presente proyecto.

### 1.10.1 Justificación del proyecto

Como se indicó anteriormente, el presente proyecto sirve para agilizar la generación de horarios de citas de terapia física y rehabilitación para un hospital de atención pública, así como, la eliminación de horarios de citas de pacientes con una determinada cantidad de inasistencias, que causan retrasos en la creación de horarios de citas para otros pacientes.

Los principales beneficiarios de la solución que plantea el proyecto serán los pacientes de un hospital de atención pública, ya que la elaboración de sus horarios de citas se harán de manera más rápida a lo que actualmente es, debido a que ya no estarán sujetos a la disponibilidad del tecnólogo, ya tendrán un horario de citas para sus respectivas terapias de acuerdo a las restricciones de tiempo entre cada terapia dado por el diagnóstico elaborado por el médico.

### 1.10.2 Viabilidad del proyecto

En este punto se detalla la viabilidad en los aspectos temporales y económicos.

#### Viabilidad Temporal

El desarrollo de la solución de software se inició en la quincena de marzo del 2013, con una finalización prevista para la última semana de junio del mismo año. El alcance se ha definido de acuerdo a las necesidades detectadas y al tipo de proyecto y los tiempos de ejecución han sido planteados según estas necesidades y las restricciones relativas a la disponibilidad de recursos de tiempo y económicas aplicables a un proyecto de fin de carrera.

## **Viabilidad Económica**

Las herramientas que serán necesarias para implementar el software son de distribución gratuita por internet, por lo tanto, no necesitará de inversión en software especial. En el caso del equipamiento necesario, se dispone de una computadora personal portátil y una computadora de escritorio con el software ya instalado que sirvieron tanto para el desarrollo del software como para las respectivas pruebas, por lo tanto no se requirió comprar equipos nuevos. Solamente se tuvieron gastos mínimos para temas de coordinación y operativos.

### **1.11 Plan de actividades**

En esta sección se presentan los elementos necesarios para la planificación del proyecto, siguiendo algunas buenas prácticas de la guía del PMBOK, consideradas relevantes para este trabajo.

#### **1.11.1 Estructura de descomposición del trabajo**

A continuación, en la Figura 1.7 se presenta una división detallada de los componentes del trabajo total que se realiza.

#### **1.11.2 Diagrama de Gantt**

En el Anexo 2 de este documento se encuentra el diagrama de Gantt correspondiente a las actividades a ejecutar.

#### **1.11.3 Plan de Riesgos**

En la Tabla 1.3 se muestran los riesgos identificados para el proyecto, así como los planes que se usa para hacer frente a ellos.

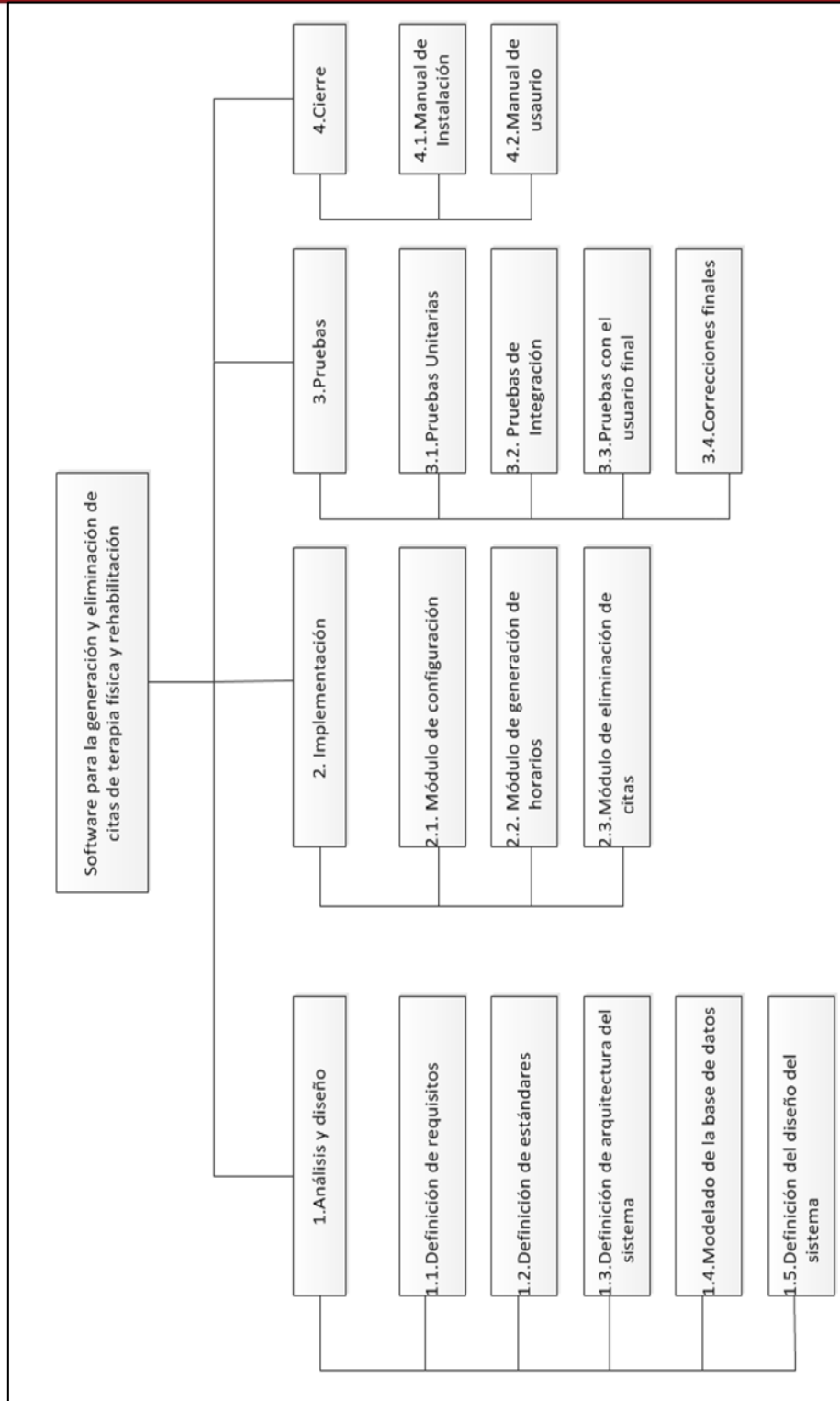


Figura 1.7 Descomposición del trabajo.



Riesgo	Impacto	Probabilidad	Severidad	Plan de Mitigación	Plan de Contingencia
Retraso en la presentación de entregables.	Alto	Media	Alta	Hacer el seguimiento del avance del proyecto verificando que esté acorde con lo planificado.	Reducir el tiempo dedicado a otras actividades para dedicar más tiempo al trabajo de fin de carrera.
Ausencia de usuarios adecuados disponibles para realizar pruebas.	Medio	Alta	Alta	Hablar con anticipación con los usuarios involucrados para que estén disponibles cuando se los requiera.	Obtener opiniones de otros usuarios de clínicas que hagan consultas médicas o exámenes médicos con cierta periodicidad de citas
Pérdida o daño del código fuente del producto.	Alto	Baja	Media	Tener un repositorio que guarde las distintas versiones del software.	Restaurar el código fuente del software a la última versión que funcionó de forma correcta.
Necesidad de inversión en recursos necesarios de manera imprevista.	Bajo	Baja	Baja	Tener una lista de todos los recursos necesarios que utiliza el software.	Obtener recursos de bajo costo para cubrir las necesidades del proyecto.

Tabla 1.3 Plan de Riesgos.

## 2. Capítulo II: Análisis

El servicio de Medicina Física y Rehabilitación de un hospital del Estado sin automatización de procesos no proporciona un tiempo adecuado de respuesta, ya que todos los registros se tienen documentos físicos y por lo tanto suelen llevar algún tiempo en obtener cierta información tanto para los pacientes como para los médicos y tecnólogos.

Actualmente el proceso no funciona de manera automatizada, ya que la información que debe manejarse sigue creciendo por el número de pacientes y número de consultas, es por ello que la implementación de un sistema de información permite una reducción en el tiempo de los procesos relacionados con la asignación y control de citas del servicio de medicina física y rehabilitación. Esta propuesta se considera práctica ya que se busca mejorar el sistema utilizando tecnologías de información.

En este capítulo se presenta el análisis de la solución a implementar. En la primera sección se presenta los requerimientos que el sistema cumple y la segunda sección detalla el análisis realizado para la solución.

## **2.1 Metodología aplicada para el desarrollo de la solución**

Se ha automatizado el proceso de asignación de citas a pacientes y la gestión de su asistencia a las mismas, en el Servicio de Medicina Física y Rehabilitación de un hospital, considerando las restricciones de personal y otros recursos con los que cuenta el mencionado servicio. La metodología que seguirá esta parte de análisis será en cascada, ya que se presentará todo el análisis para el desarrollo del software.

## **2.2 Identificación de requisitos**

Este trabajo consistió en realizar visitas de observación a los diferentes ambientes del área de Medicina Física y Rehabilitación del hospital; así como, entrevistas a los diferentes empleados que desempeñan funciones en los mismos. Además, se entrevistó a profesionales del área administrativa que trabajan de forma muy estrecha con el personal del área en mención; tales como, personal de Admisión, Archivo y Caja.

En cuanto a la obtención de información que no se refiere directamente a la labor médico-hospitalaria, fue obtenida mediante reuniones con los encargados de estadística e informática.

### **2.2.1 Requisitos funcionales**

En este punto se describen los requisitos funcionales del software separado en 3 módulos.

#### **2.2.1.1 Módulo de configuración inicial**

- 1) El sistema permite cargar información básica de los pacientes, tecnólogos y médicos tales como DNI, apellido paterno, apellido materno, nombre,

dirección, número telefónico, fecha de nacimiento; y sólo para los pacientes la fecha de su última atención en consultorio; además, de la última sesión de terapia.

- 2) El sistema permite registrar información del diagnóstico dado al paciente como son el tratamiento, indicaciones del intervalo de tiempo entre cada terapia y lo que se debe aplicar en cada una.
- 3) El sistema permite cargar información de los tecnólogos como disponibilidad de horas de trabajo mensual y preferencias de cada uno.
- 4) El sistema permite definir varios usuarios y los permisos de visualización del sistema de cada uno.

#### **2.2.1.2 Módulo de generación de horarios**

- 1) El sistema generará un cronograma de terapias que cumpla con asignar para cada sesión de terapia a un tecnólogo disponible y el tipo de preferencia por enfermedad. Asimismo, se debe lograr un balance de carga de trabajo equitativo entre todos los tecnólogos.
- 2) Al asignar las horas de disponibilidad de cada tecnólogo se tomará en cuenta las horas de refrigerio de 12:00 am hasta las 1:30pm, por trabajador.
- 3) Las sesiones de terapia no serán necesariamente asignadas todas al mismo tecnólogo.
- 4) Se considerará como prioridad principal la disponibilidad, seguida por las preferencias o capacidades de los tecnólogos.

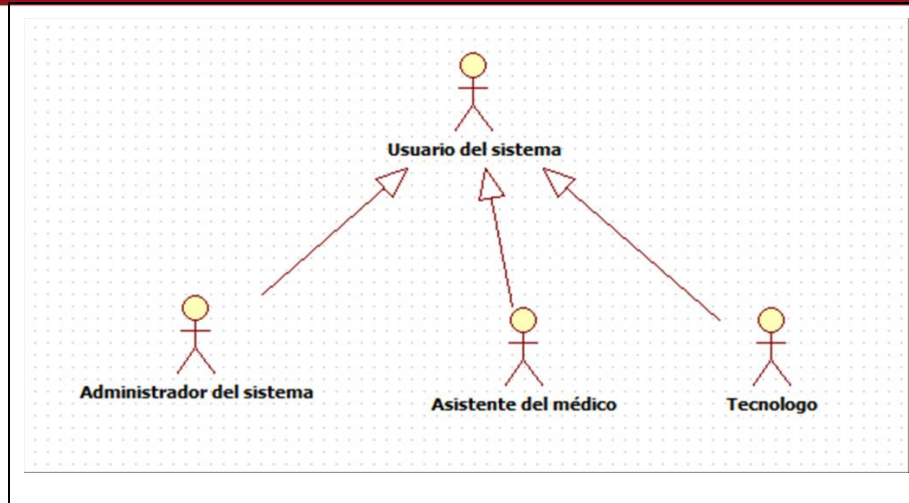
- 5) Si un tecnólogo no puede asistir a alguna de las citas asignadas, deberá ser reemplazado por otro.
- 6) La duración de cada sesión de terapia es de 30 minutos y será tomada en cuenta para planificar la disponibilidad de cada tecnólogo.
- 7) El sistema contará con una interfaz para modificar el cronograma de terapias generado, en caso de que estos requieran dichos cambios.

### **2.2.1.3 Módulo de eliminación de citas**

- 1) El sistema podrá actualizar la información de la disponibilidad de cada tecnólogo.
- 2) El sistema actualizará las citas programadas para tecnólogos que no puedan asistir por algún motivo (faltas, enfermedad, etc.) y las asignará a otros tecnólogos que si van a estar presentes. Esto se hace con el fin de no dejar al paciente sin atención y de que no se entere de que su terapia no se va a poder realizar en el momento que llegue para recibirla.
- 3) Después de 3 inasistencias del paciente, se eliminarán las citas restantes del mismo. La eliminación de citas y actualización de la disponibilidad de los tecnólogos se realizarán cada vez que se detecte la cantidad definida de inasistencias.
- 4) El sistema eliminará las citas restantes de los pacientes con 3 o más inasistencias en total con respecto a las que se le programaron, por lo general son 10 sesiones de terapia.

## **2.3 Identificación de actores principales**

Los actores principales que tienen el sistema desarrollado son los que se muestran en la Figura 2.1:



**Figura 2.1** Diagrama de Actores del Sistema.

**Asistente del médico:** generalmente este rol es realizado por una técnica de enfermería. Este actor es el responsable de registrar, modificar y dar de baja la información correspondiente al diagnóstico, indica la información técnica en cada terapia y el intervalo de tiempo de cada terapia.

**Tecnólogo:** es quién realiza las terapias a los pacientes. Tendrá acceso a la información de cada paciente que va a atender, para revisar las indicaciones del médico y poder actualizar las observaciones o resultados de cada sesión de terapia. Para esto requiere tener acceso al cronograma de terapia de cada paciente. Verifica la asistencia del paciente a cada terapia. No requiere tener acceso a otras tablas generales del sistema ni a su mantenimiento.

**Administrador del sistema:** posee permisos de acceso a todas las pantallas y funciones del sistema y se encarga del registro de usuarios y la asignación de roles a los mismos de acuerdo al tipo de labor que realizarán. Además, es el encargado de la administración de seguridad del sistema.

## 2.4 Casos de uso por paquete

A continuación se explican brevemente los casos de uso de cada uno de los paquetes identificados. Un detalle mayor podrán encontrarse en el Anexo 1 de este documento.

### 2.4.1 Paquete de Administración

Este paquete está constituido por el conjunto de actividades que realiza el actor Administrador del sistema que se muestra en la Figura 2.2 tiene:

- **Administrar pacientes:** alta, modificación y baja de registro básico de los pacientes
- **Administrar médicos:** alta, modificación y baja de la información básica de los médicos encargados de atender pacientes y derivarlos al servicio de Medicina Física y Rehabilitación.
- **Administrar tecnólogos:** alta, modificación y baja de la información de los tecnólogos, que incluye la gestión de sus preferencias de tratamientos, disponibilidad horario y programación de citas con pacientes.
- **Administrar pagos:** recepción de los pagos realizados en Caja del hospital, lo que sirve como dato fundamental para definir la asistencia o inasistencia de los pacientes a sus citas programadas.

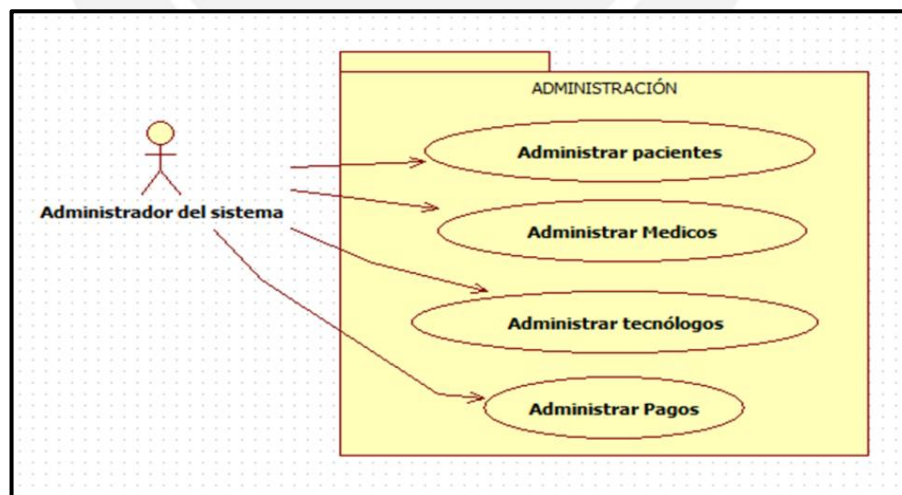


Figura 2.2 Paquete de administración.

### 2.4.2 Paquete de Generador de cronograma

El paquete generador de cronogramas que se grafica en la Figura 2.3 es gestionado por el asistente del médico y permite administrar los diagnósticos y los cronogramas de atención a los pacientes.

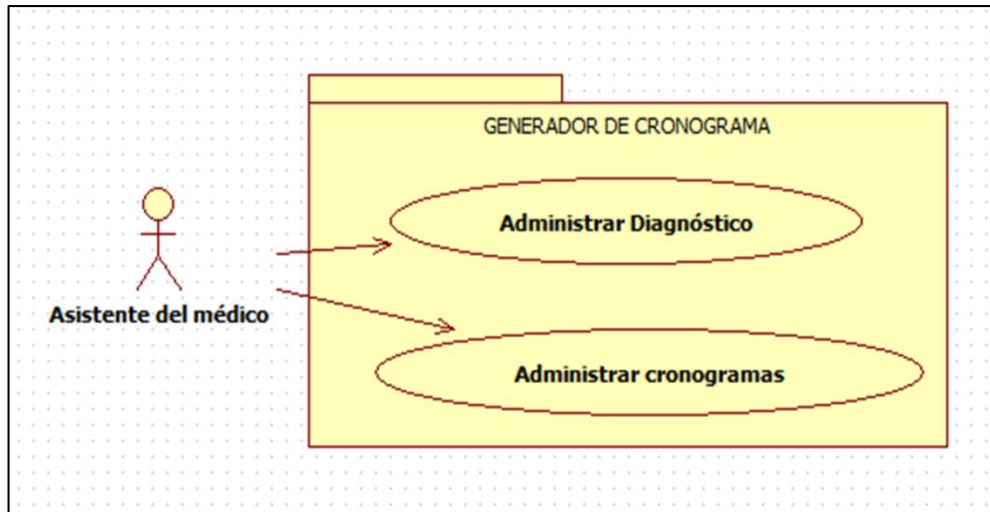


Figura 2.3 Paquete de administración.

### 2.4.3 Paquete de Seguimiento

Este paquete que se muestra en la Figura 2.4 es utilizado por el actor Tecnólogo para realizar el seguimiento de la evolución del paciente de terapia y su asistencia o inasistencia a las mismas.

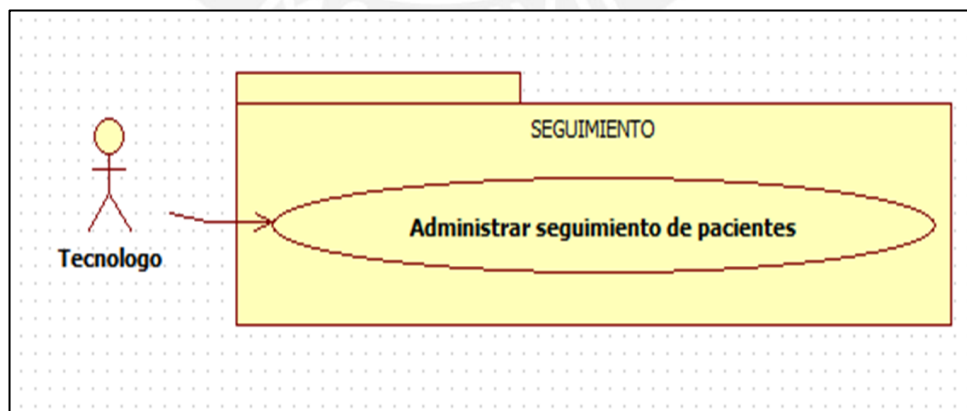


Figura 2.4 Paquete de seguimiento.



## 2.5 Características de los Usuarios

El sistema a desarrollar está orientado al personal de un hospital. Dicho personal está familiarizado con los procesos principales del negocio, pero su conocimiento en la utilización de computadores es básico y limitado, en la mayor parte de los casos. Es por ello que uno de los requisitos indispensables del software es que la interfaz de usuario sea muy sencilla de utilizar.

## 2.6 Suposiciones y dependencias

En este punto explicaremos las suposiciones y las dependencias sobre las que se basa el desarrollo del software.

### 2.6.1 Suposiciones

Se asume que las exigencias plasmadas en el documento se mantienen incluso después de ser aprobadas por los usuarios finales. Además se asume que el sistema será utilizado sobre la plataforma Linux y contará con los recursos necesarios de memoria y almacenamiento para asegurar un funcionamiento adecuado. El sistema operativo a utilizar es irrelevante desde el punto de vista del usuario puesto que se trata de una aplicación web que puede utilizar cualquiera de los exploradores más reconocidos del mercado.

### 2.6.2 Dependencias

La información de pacientes se encontrará en un archivo XML el cual será obtenido de la información histórica que actualmente se encuentre almacenada en el hospital y será procesado en el sistema mediante una interface. La consulta y modificación de los registros en la base de datos estarán sujetos a la conexión entre el equipo servidor y las terminales, las cuales se encontrarán en ambientes dentro del área de medicina física.

### 3. Capítulo III: Diseño

En el presente capítulo se desarrolla la definición de la arquitectura del sistema, la elección de las herramientas a utilizar, el diseño de la interfaz gráfica de usuario; así como, el diseño del algoritmo Hill Climbing explicado en el Capítulo 1 de este proyecto.

#### 3.1 Definición de la arquitectura del sistema

La Arquitectura del sistema es el diseño de más alto nivel de la estructura de un sistema y consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco necesario para su desarrollo.

Como se mencionó anteriormente, la arquitectura para este Proyecto será seleccionada y diseñada considerando sus objetivos y restricciones. En tal sentido no sólo se considerarán los aspectos funcionales sino también las restricciones de tipo económico a las que se puede enfrentar el hospital.

La arquitectura de software define, de manera abstracta, los componentes que llevan a cabo alguna tarea de computación, sus interfaces y la comunicación entre

ellos. Toda arquitectura debe ser implementable en una arquitectura física, que consiste simplemente en determinar qué computadora tendrá asignada cada tarea.

### 3.2 Arquitectura de solución

Dados los requerimientos de la solución, así como las restricciones en cuanto a la adquisición y modernización de las tecnologías existentes, se ha determinado que la mejor arquitectura a utilizar para el desarrollo exitoso de este proyecto es una Arquitectura de tres niveles con el siguiente detalle por capa:

- **Capa de Presentación.** Que comprende todos los formularios necesarios para la interacción directa con el usuario. Un requisito fundamental es la integración lógica de los componentes de esta interfaz. Es indispensable, asimismo, que la velocidad de acceso a estos formularios, y a la obtención de los resultados de los algoritmos programados sea la menor posible, y eso dependerá en gran medida de las herramientas utilizadas, así como de un diseño óptimo de la aplicación. El no cumplir con estos requisitos ocasionaría que los usuarios rechacen la utilización del software, o que en el mejor de los casos, deba vencerse una gran resistencia al cambio.
- **Capa de Lógica de negocio.** En esta capa se ubicará toda la lógica de los algoritmos de asignación óptima de los horarios, así como los relativos a la modificación de horarios, turnos, tecnólogos, eliminación de citas de pacientes que no asisten oportunamente, y en consecuencia, también de la reasignación de turnos. En este caso el factor crítico de éxito vuelve a ser la velocidad de respuesta, pues el usuario va a evaluar la utilidad de la aplicación en la medida que esta capa genera con precisión y rapidez el resultado final del procesamiento efectuado en esta capa.
- **Capa de Persistencia o almacenamiento.** Nuevamente esta es una capa con la que no tendrá contacto directo el usuario y sólo evaluará los resultados que obtenga en su interfaz. La base de datos debe garantizar la integridad de la información contenida en ella, su mantenibilidad y auditabilidad, entre otros aspectos importantes.

### 3.3 Herramientas a utilizar

Para la programación de toda la aplicación se ha decidido utilizar el lenguaje de programación Python, en su versión 2.7 y utilizando el framework Django 1.5.1,

utilizando el IDE Eclipse 4.2.2. Como base de datos para la data del Proyecto, se ha decidido utilizar MySql.

Se ha decidido por dos herramientas Open Source, muy potentes y difundidas, que minimizarían la inversión que debe realizar el hospital para su utilización. Con respecto al lado del soporte, cada vez se va generando una base mayor de usuarios de ambos software, obviamente aún menor a la que existe para soluciones comerciales tales como Microsoft .Net y Microsoft SQL Server u Oracle, pero cuyos costos los hacen inviables dadas las circunstancias actuales. Existen una amplia base de soportes internacionales a través de la Web, que hacen sumamente sencillo acceder a la solución de problemas que pudieran presentarse en el día, siendo entre muchos la comunidad de Stackoverflow (STACKOVERFLOW 2013) una de las más serias y completas. Obviamente, se espera que en el futuro se incremente la disponibilidad de soporte local, sobre todo, gracias a iniciativas precisamente del Estado a través la Oficina Nacional de Gobierno Electrónico e Informática (ONGEI) de la Presidencia el Consejo de Ministros. Una referencia a esta política es la siguiente (ONGEI 2008) :

*Software Libre. El Estado preferirá emplear esta tecnología cuando existan soluciones y componentes en software libre, preferirá asimismo la libre disponibilidad del código fuente -entre otras- por razones de seguridad, las que deberán contar con el soporte técnico sólido correspondiente y con la capacidad comprobada de soportar eficientemente los servicios electrónicos, de acuerdo con la Resolución Jefatural N°199-2003-INEI.*

### 3.3.1 Python

Sobre el lenguaje de programación Python se puede decir lo siguiente (KNOWLTON 2009):

*“...se trata de un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones”.*

*Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.*

*Una característica importante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado enlace dinámico de métodos).*

### 3.3.2 MySQL (MySQL 2013)

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones, perteneciente a Oracle Corporation, que posee un esquema de licenciamiento dual: versiones empresariales licenciadas y una versión comunitaria, ofrecida bajo la GNU GPL. Está desarrollado en su mayor parte en ANSI C.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

Algunas de sus características más importantes son:

- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por Tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL

### 3.4 Modelamiento de la Base de Datos

El modelo de datos contempla las siguientes tablas. En la Figura 3.1 se muestra el Diagrama Entidad Relación estos elementos. Las tablas más importantes son brevemente explicadas a continuación:

- **Pacientes.** Contiene los datos relevantes del paciente tales como DNI, apellido paterno, apellido materno, nombre, dirección, número telefónico, fecha de nacimiento, fecha de última atención en consultorio, fecha de última sesión de terapia. Esta, como todas las otras tablas, deberán poseer también los campos necesarios para la auditoría de las modificaciones que se realicen.
- **Historia clínica de Medicina física y rehabilitación.** Los campos de esta Tabla son DNI del paciente, DNI del médico, número de acto médico, edad, fecha, turno, consultorio, código CIE10 del diagnóstico, reseña de la enfermedad, examen clínico, tratamiento, evolución, recomendaciones, cantidad de sesiones necesarias, cantidad de sesiones efectuadas.
- **Medicamentos por acto médico.** Cada acto médico podría requerir uno o más medicamentos para el tratamiento de la dolencia del paciente: DNI, acto médico, código de medicamento, unidad de medida de la dosis, frecuencia de dosis, total de dosis indicadas.
- **Personal asistencial.** Los datos más importantes son Tipo (médico, tecnólogo, enfermera), DNI, CMP (en el caso de médicos), RNE (en el caso de médicos especialistas), apellido paterno, apellido materno, nombres.
- **Preferencias,** conteniendo DNI, tratamiento preferido 1, tratamiento preferido 2, tratamiento preferido 3.
- **Disponibilidad.** Los campos de esta Tabla serían: DNI del trabajador asistencial, código de turno, flag de disponibilidad en el turno (0 o 1)
- **Maestro de medicamentos.** Código de medicamento, lote, nombre genérico, nombre comercial, unidad de compra, unidad de despacho, stock actual (unidades de despacho), fecha de vencimiento, referencia de utilización, dosis recomendada, etc.

Existen otras tablas que no se han mencionado en esta sección para no abundar en detalles técnicos. En el Anexo 3 “Descripción de las clases para el desarrollo del software” se podrá encontrar el detalle de las tablas utilizadas y sus clases Django asociadas.

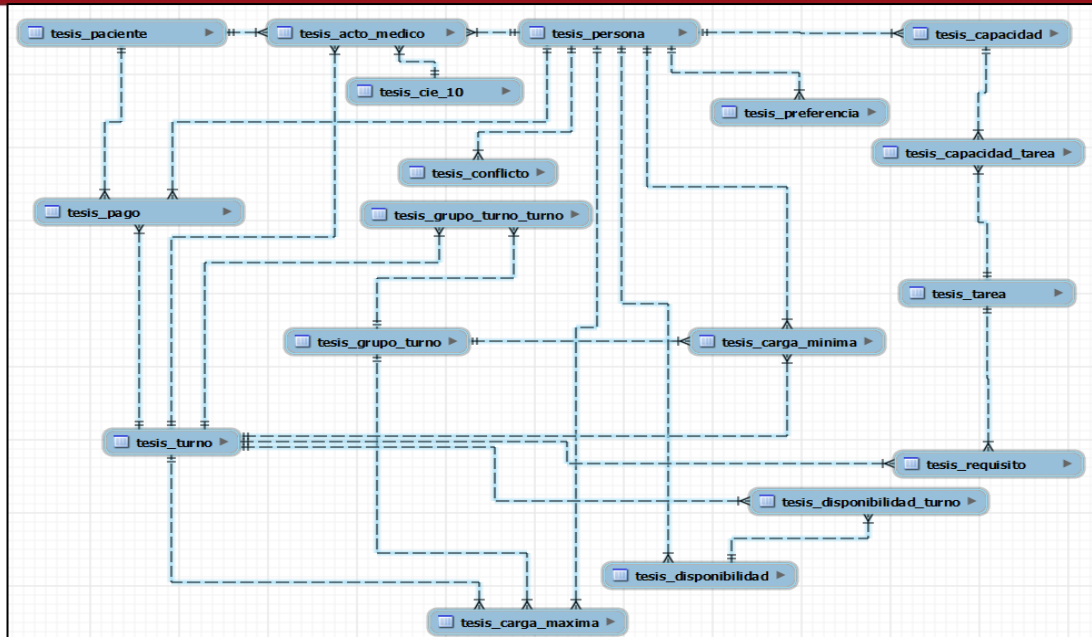


Figura 3.1 Diagrama Entidad Relación (básico).

En la Figura 3.2 se muestra el diagrama de entidad relación desplegado.

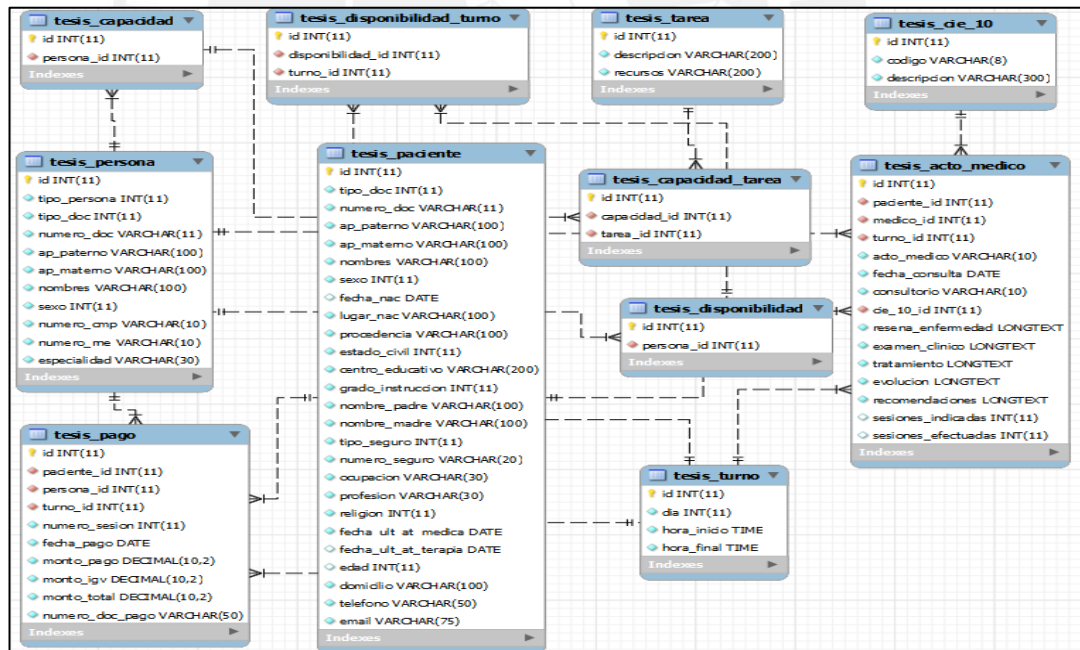


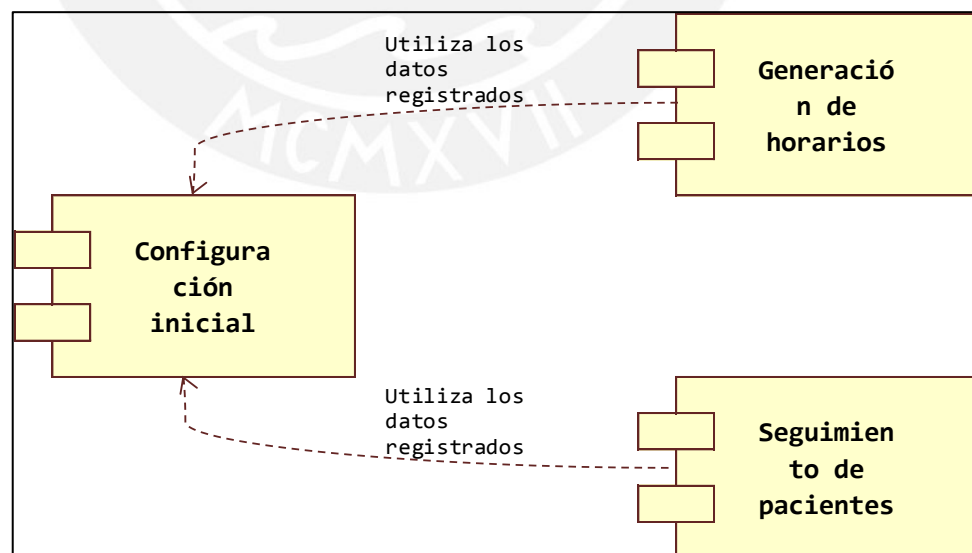
Figura 3.2 Diagrama Entidad Relación (desplegado).

### 3.5 Definición del diseño del sistema

En este apartado se describen los módulos principales así como se presenta el prototipo de diseño de la interfaz de usuario (formularios).

- Módulo de configuración inicial. Se podrá administrar toda la información como DNI, apellido paterno, apellido materno, nombre, dirección, número telefónico, fecha de nacimiento, fecha de última atención en consultorio y fecha de última sesión de terapia de los pacientes. Para los médicos y tecnólogos se obviarán las dos últimas fechas. Es decir, se podrá registrar en forma masiva la información dada en el modelamiento de datos en la sección 3.2. Además, se podrá hacer mantenimiento de la base de datos que se encuentre relacionado a médicos, tecnólogos, pacientes.
- Módulo de generación de horarios. Procesa cada vez que es requerido el algoritmo de asignación de horarios para cada paciente y da a elegir al paciente entre varias alternativas según su conveniencia. El usuario interactúa con los resultados de este módulo, sin necesidad de conocer el detalle del algoritmo y cómo este genera las alternativas. También incluye todas las acciones de mantenimiento de las tablas de restricciones del sistema: requisitos, capacidad, disponibilidad, conflictos y carga de trabajo por modificaciones debidas a inasistencias de los tecnólogos o los pacientes.
- Módulo de seguimiento de pacientes. En este módulo, tanto el médico en el momento de consulta como el tecnólogo durante las sesiones de terapia, actualizan la información relativa a cada acto médico, según aparezcan las ocurrencias.

En la Figura 3.3 se muestra el diagrama de los componentes mencionados.



**Figura 3.3** Diagrama de los componentes descritos.



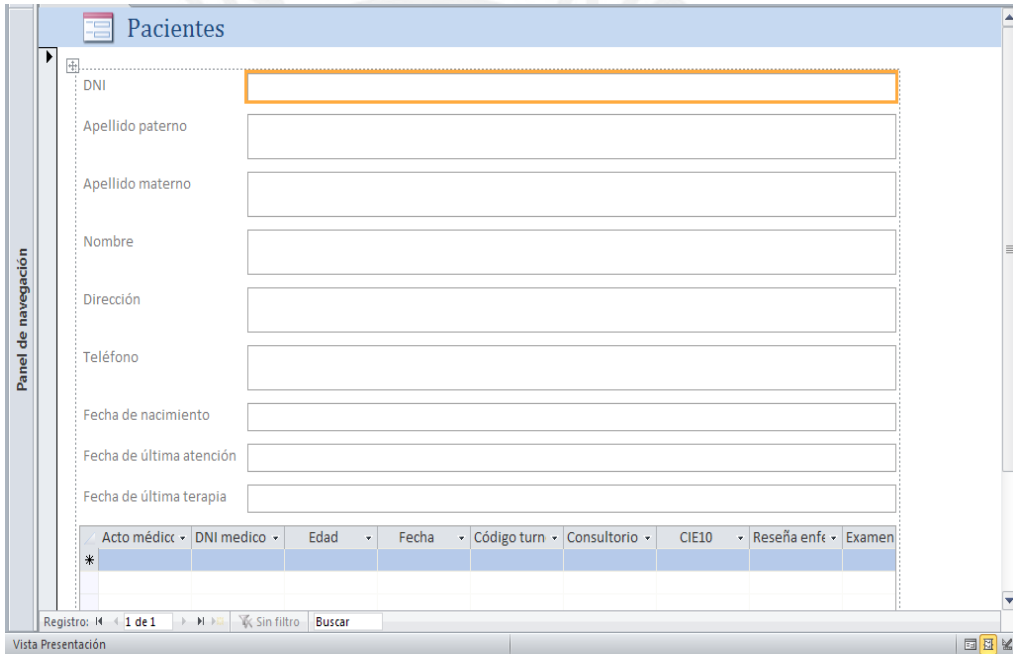
### 3.6 Diseño de la interfaz gráfica

En esta sección se presenta el diseño preliminar de las pantallas del sistema. Al ser una aplicación Web, algunas de las características detalladas en estos esquemas podrían variar ligeramente en su presentación, tal como se verá en los prototipos a presentar en el siguiente capítulo.

#### 3.6.1 Registro de pacientes

En esta pantalla se registra la información básica de cada paciente, sirviendo como guía para el diseño la información que actualmente figura en la ficha que es llenada manualmente.

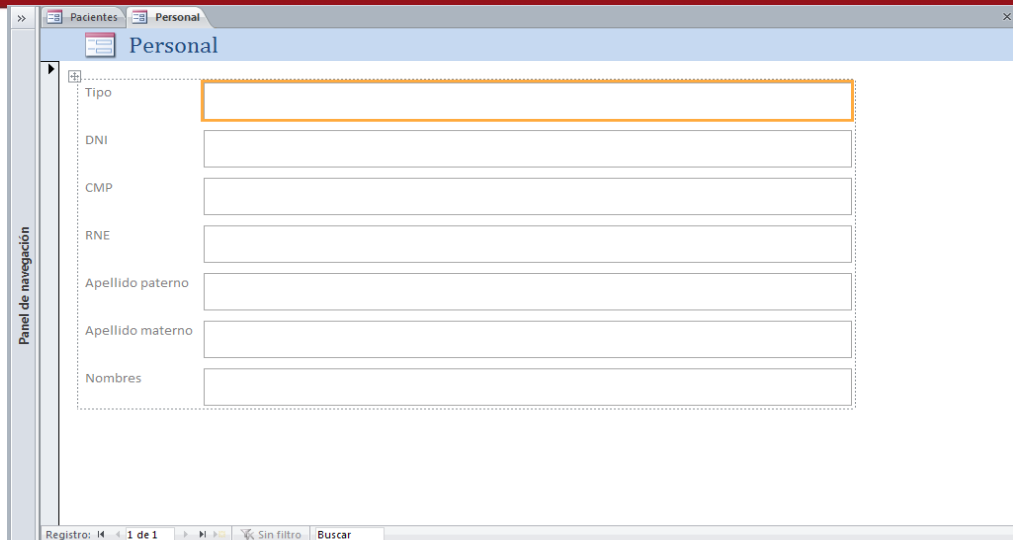
Esta pantalla (pacientes) que se muestra en la Figura 3.4 debe también mostrar todas las atenciones o actos médicos, relacionados con el individuo determinado.



**Figura 3.4** Registro de pacientes.

#### 3.6.2 Registro de personal

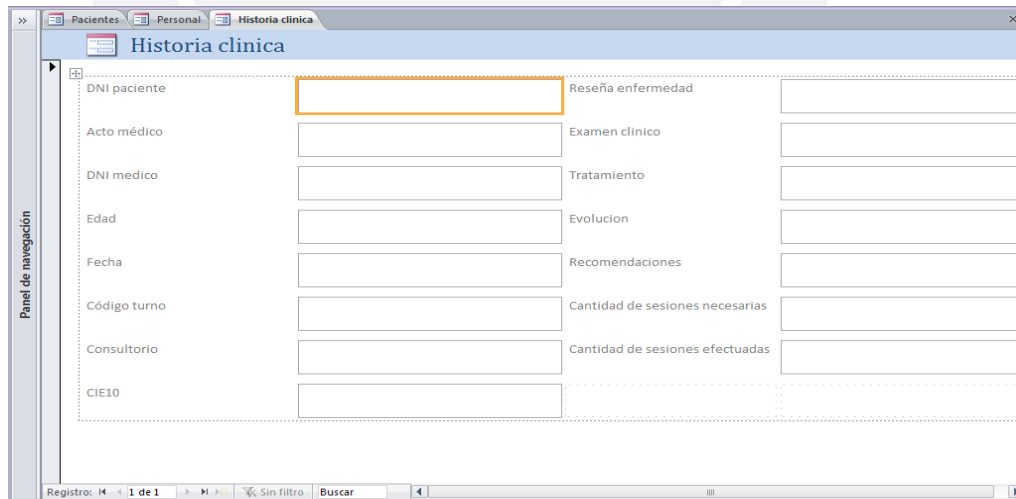
En la pantalla mostrada en la Figura 3.5, se debe registrar los datos del personal asistencial que labora en el Hospital San José. Se ha decidido mantener en una misma tabla los datos de los médicos, tecnólogos, enfermeras, etc., distinguiendo su grupo ocupacional al seleccionar el “Tipo de personal” en la pantalla de ingreso de datos.



**Figura 3.5** Registro de personal.

### 3.6.3 Registro de historia clínica

La historia clínica contiene el registro de todos los actos médicos practicados a un determinado paciente. En ella se graban los detalles de la enfermedad, el relato del examen clínico practicado, así como el tratamiento indicado, la evolución de la enfermedad y las recomendaciones brindadas. El diseño básico se presenta en la Figura 3.6:

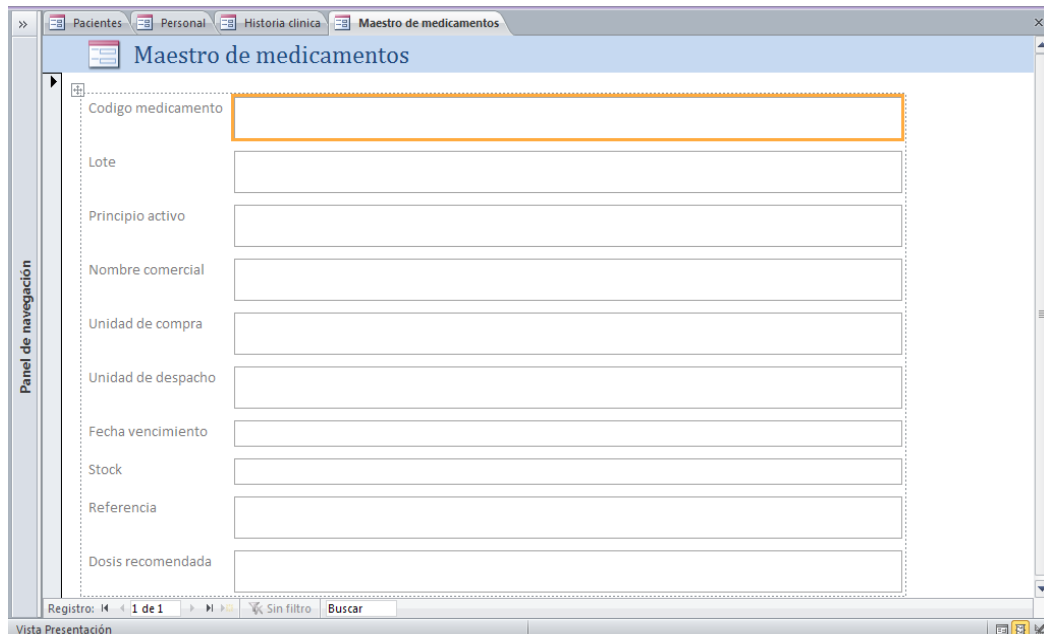


**Figura 3.6** Registro de historia clínica.

### 3.6.4 Maestro de medicamentos

Esta pantalla (Figura 3.7) se utiliza para el registro de los medicamentos que contiene el maestro correspondiente y los datos más importantes son el código del

medicamento, el número de lote, la fecha de vencimiento, el nombre genérico y el nombre comercial, etc.

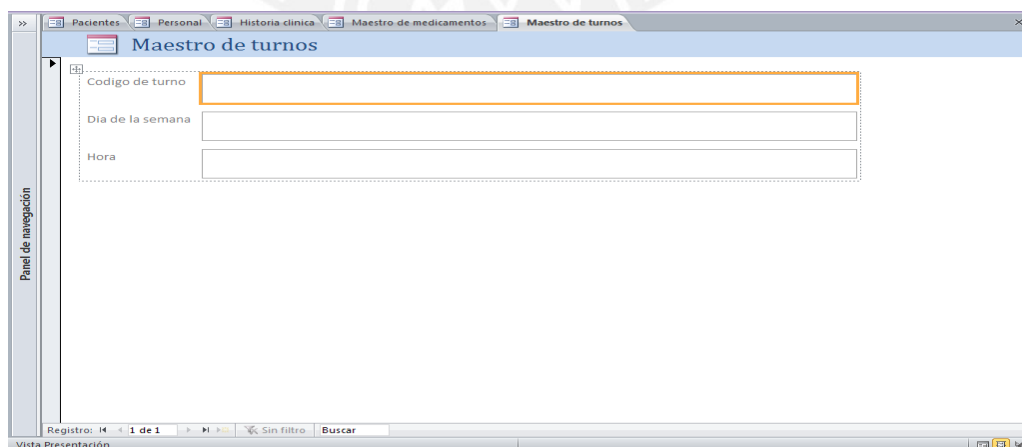


**Figura 3.7** Maestro de medicamentos.

### 3.6.5 Maestro de turnos

En esta sección se registran los turnos existentes durante la semana, indicando el día de la semana y la hora en la que se inicia el turno, tal como se muestra en la Figura 3.8. Cada turno tiene una duración de una hora.

A partir de estos datos se genera la disponibilidad de cada tecnólogo para la generación de los horarios óptimos.



**Figura 3.8** Maestro de turnos.

## 4. Capítulo IV: Construcción y pruebas

En este capítulo se explica la construcción del algoritmo de asignación de óptimos horarios y turnos, según las definiciones explicadas en los capítulos precedentes. Asimismo, se detallan los algoritmos y parámetros necesarios para solucionar los diversos casos de uso que puedan presentarse para la solución del problema.

### 4.1 Construcción

Según lo mencionado anteriormente, se utilizó el lenguaje Python 2.7 para realizar la programación de la aplicación. El framework a utilizar para la implementación fue el Django 1.5.1, con el IDE Eclipse SDK versión 4.2.2. Como sistema operativo se está utilizando Windows 7 aunque podría utilizarse en forma transparente casi cualquier distribución de Linux, ya que el conjunto de herramientas tiene versiones nativas para ese sistema operativo.

Se construyó la aplicación en tres ciclos: el primero tuvo como producto el modelo de datos del sistema, en el que se almacenan los datos de entrada y los horarios óptimos generados por la aplicación. El segundo ciclo se dedicó a la codificación del algoritmo de optimización propiamente dicho, y que considera la carga máxima de cada tecnólogo por turno de una hora y los recursos disponibles, llámense

cubículos, camillas, sillas de ruedas y equipamiento especializado en general. Finalmente, el tercer ciclo se dedicó a la construcción de las interfaces necesarias para todos los módulos y la carga de los datos. Se incluyó también en este ciclo, la construcción de las herramientas necesarias para la carga inicial de los datos en formato XML a MySQL. Por las características técnicas de la herramienta utilizada, en esta etapa se realizó también el diseño final y la codificación de la interfaz gráfica de usuario, basada en Web, incluyendo soporte para dispositivos móviles.

## 4.2 Implementación de estructuras de datos en memoria

El framework utilizado, Django 1.5.1, permite realizar el modelamiento de las estructuras de datos en su propio entorno sin tener que utilizar necesariamente, comandos SQL o la aplicación MySQL Workbench para crear las tablas, índices y relaciones de la base de datos que estamos diseñando para la aplicación.

Para esto, cuando se crean los proyectos de aplicaciones, Django genera un módulo en blanco con el nombre “models.py” que se utiliza para el diseño de las mencionadas estructuras de datos.

Para que el modelamiento de las estructuras de datos sea posible, es necesario importar el objeto de Django apropiado:

```
from django.db imports models
```

A partir de este punto se puede proceder a la creación de las tablas de la base de datos, a continuación se presenta el código necesario para crear la Clase Paciente, que precisamente genera la Tabla de pacientes. Es importante destacar que una de las grandes ventajas que ofrece Django es que el modelo se puede modificar dinámicamente; ya que si se requiere agregar, eliminar o modificar algún campo o índice, simplemente se debe cambiar el modelo en Django y luego ejecutar el comando de línea `python manage.py syncdb` para tener todo sincronizado y actualizado en MySQL.

Las demás clases involucradas se muestran en el Anexo 3 “Descripción de las clases para el desarrollo del software”.

### 4.3 Implementación del algoritmo

En esta sección se describe el algoritmo utilizado en diferentes pasos para obtener la asignación óptima de pacientes a los diversos horarios y tecnólogos, según las restricciones presentadas en el sistema.

1. Se definió una función de costo,  $f$ , que mide el costo en función de los siguientes aspectos: el tiempo de espera en días antes de que el paciente pueda acceder a su primera atención, la violación de restricciones, las preferencias de los tecnólogos y la limitación de espacios, equipos y otros recursos para la prestación de las terapias. Debe tenerse en cuenta que por cada hora, los tecnólogos pueden citar hasta cuatro pacientes, dependiendo del tipo de terapia que estos van a recibir y los recursos que se requieren para brindársela (cubículos disponibles, camillas, sillas de ruedas, equipos de terapia de calor, frío, laser, etc.).
2. Antes del inicio de la búsqueda de los horarios disponibles, debe actualizarse la disponibilidad en función de los pacientes que no han acudido a sus citas programadas en más de tres oportunidades (desertores). Esto permite liberar horarios que habían sido previamente ocupados y que pueden ser asignados a nuevos pacientes. Para realizar esto se requiere tener el registro de las inasistencias por paciente programados, hasta la hora en que se inicia la búsqueda.
3. Uno de los primeros problemas que debe resolver el algoritmo es la determinación de un horario de inicio para la búsqueda. Algunos parámetros deben ser configurados en la aplicación para hacer la programación. Entre los parámetros a considerar está el horizonte de planeación, que puede ser, considerando que cada terapia dura unos diez días interdiarios, de tres semanas (21 días). Otro de los parámetros importantes es la determinación del número de iteraciones que se va a realizar antes de parar la ejecución del algoritmo, a fin de no entrar en un bucle infinito que detenga la ejecución de la aplicación.

4. Para determinar el primer horario asignado al paciente se ha visto por conveniente que se realice una generación aleatoria, tanto de la fecha (número de días a partir de la fecha actual), dentro de cada día la asignación aleatoria de alguno de los tecnólogos disponibles y finalmente, la selección de un horario en el cual el tecnólogo tiene programados en una fecha determinada.
5. El siguiente paso, para todas las iteraciones es verificar si es una asignación factible; es decir, si el horario asignado ya se encuentra ocupado por algún paciente o se encuentra libre. También debe verificarse si los turnos interdiarios siguientes de ese tecnólogo están disponibles o ya han sido tomados por otro paciente. De estar libres, se calcula el costo de la asignación, que como ya mencionamos líneas arriba, tiene que ver con los días que el paciente tiene que esperar antes de su primera atención. Entonces, diremos que todas las restricciones tienen una misma ponderación y que su violación agrega una unidad al costo total. También se calcula la carga de trabajo del tecnólogo como la suma de citas que tiene asignadas para un día determinado.
6. Si el horario de inicio seleccionado no ha resultado factible se regresa al paso 4. En caso contrario se continúa al paso 7.
7. Se genera una vecindad. La mejor forma de hacerlo es generar nuevamente un día al azar dentro del horizonte de planeación, con un tecnólogo también seleccionado aleatoriamente y un horario al azar de los que tiene disponible.
8. Se verifica su factibilidad, si no es factible, se regresa al paso 4. Si es factible, se incrementa el número de iteraciones realizadas.
9. Se calcula el costo total de esta vecindad y si este resulta menor al que se calculó anteriormente, se asigna como nueva programación la recién calculada. En caso de que el costo resultase mayor al anterior o que el número de iteraciones exceda a la cantidad de iteraciones máximas, se

graba el último horario como el horario de atención que será entregado al paciente y se detiene el algoritmo.

10. Con respecto al costo de la nueva programación, se asigna un nuevo turno al paciente, y se continúa buscando una nueva vecindad según lo descrito en el paso 4.
11. Si al llegar al número de iteraciones máximas aún no se ha encontrado una programación para el paciente, se amplía el horizonte de búsqueda. Si al terminar este nuevo proceso, aún no se encuentra una programación adecuada se debe seguir por un nuevo y último periodo. En caso de concluir la búsqueda sin encontrar una programación adecuada, la herramienta presenta un mensaje que será comunicado al paciente, al que se le presenta como solución la asignación de atenciones como “paciente adicional”, en coordinación con los tecnólogos que se encuentren presentes en ese momento.

#### 4.3.1 Pseudo-código

El pseudo código para la implementación del algoritmo se presenta en el Script 4.1:

```
def GenerarHorario(Paciente)
    ""
    Algoritmo de generación de horarios en el servicio de
    Medicina Física y Rehabilitación del Hospital estudiado
    ""
    # Parámetros iniciales:
    max_iteraciones=50
    max_periodo_búsqueda=63
    iteraciones=1
    periodo_planeacion=21
    horizonte_actual=0
    horario_factible=False
```



```
dias_espera=0

tecnologo_asignado=0

f_previo=1000          #Costo total

f_nuevo=0

horario_optimo={}

horario_tentativo={}

numero_sesiones_a_programar=

    select sesiones_indicadas

    from acto_medico

    where paciente_id=paciente_a_programar

        and acto_medico.id=acto_medico_a_programar

#Actualiza la disponibilidad de horarios
ActualizaHorarios()

#Cuenta la cantidad de tecnólogos en la BD
max_tecnologos=CuentaTecnologos()

#Horario óptimo

while not horario_factible or horizonte_actual>max_periodo_búsqueda:

    horizonte_actual=horizonte_actual+periodo_planeacion

for iteraciones<max_iteraciones:

    dias_espera=GeneraDiasEspera()

    tecnologo=GeneraTecnologo()

    fecha_tentativa=hoy()+dias_espera

#Elige un horario al azar del tecnólogo seleccionado
```

```
horario_tentativo=SeleccionaHorario(fecha_tentativa, tecnologo)

#verificación de factibilidad

horario_factible=VerificaFactibilidad(fecha_tentativa,
    horario_tentativo, tecnologo, numero_sesiones_a_programar)
if horario_factible:
    f_nuevo=dias_espera
    #Calcula otros costos
    f_nuevo=f_nuevo+CalculaCostos(paciente,fecha_tentativa,
        horario_tentativo, tecnologo,
        numero_sesiones_a_programar)
    if f_nuevo<f_previo:
        f_previo=f_nuevo
        horario_optimo=DeterminaProgramacion(fecha_tentativa,
            horario_tentativo, tecnologo,
            numero_sesiones_a_programar)
        iteraciones=iteraciones+1

if not horario_factible:
    Mensaje('No se ha podido determinar un
        horario adecuado para el paciente')
    return

Actualizar_programación_de_horarios(paciente, horario_optimo, tecnologo)

def Actualizar_programación_de_horarios(paciente, horario_optimo, tecnologo)
    #en la tabla programación_ horarios:
    flag_ocupado=false
    id_paciente0=paciente
```

```
horario=horario_optimo

id_personal=tecnólogo

grabar()

#Libera los horarios asignados a pacientes desertores

def ActualizaHorarios():

    #En la tabla programación horarios:

    if conteo_inasistencias>3 y fecha_hora<=hoy():

        flag_ocupado=false

        grabar()

#Selecciona un horario al azar para el día y tecnólogo pasados

def SeleccionaHorario(fecha_tentativa, tecnologo):

    horarios_tecnologo=select horarios

    where (id_tecnologo=tecnólogo and

        fecha=fecha_tentativa from programación_horarios)

    return random(horarios_tecnologo)

#Verifica la factibilidad de un horario tentativo

def VerificaFactibilidad(fecha_tentativa, hora_tentativa, tecnólogo,

    numero_sesiones_a_programar):

    programación_factible=True

    dias_programados=select dias_programados from turnos

    where (id_personal=tecnólogo)

    #Por ejemplo {martes, jueves, sábado}

    #El horario del primer día ya se vio que sea factible

    fecha_verificada=fecha_tentativa
```

```
for i=0 to numero_sesiones_a_programar-1:

    fecha_ok=False

    while not fecha_ok

        fecha_verificada=fecha_verificada+1

        if fecha_verificada en dias_programados:

            fecha_ok=True

    if Ocupada(fecha_verificada, hora_tentativa, tecnólogo):

        programación_factible=False

        exit

    return programación_factible

#Calcular los costos asociados a una programación
def CalculaCostos(paciente, fecha_tentativa, horario_tentativo,
    tecnólogo, numero_sesiones_a_programar)

    costo=0

    for i=0 to numero_sesiones_a_programar:

        if tarea no está en la preferencia del tecnólogo:

            costo=costo+1

        if no hay disponibilidad de recursos

            para la fecha y horario tentativos:

                costo=costo+1

        if la cantidad de pacientes programados del tecnólogo

            excede su capacidad:

                costo=costo+1

        if la capacidad del servicio se excede:

            costo=costo+1

    return costo

#Crea la tuple o lista con las fechas/horas a bloquear para el paciente
```

```

def DeterminaProgramacion(fecha_a_programar, hora_a_programar,
    tecnólogo, numero_sesiones_a_programar):
    horario_a_programar={fecha_a_programar}
    fecha_verificada=fecha_a_programar
    for i=0 to numero_sesiones_a_programar-1:
        fecha_ok=False
        while not fecha_ok
            fecha_verificada=fecha_verificada+ 1
            if fecha_verificada en dias_programados:
                fecha_ok=True
            horario_a_programar=horario_a_programar+{fecha_verificada,
                hora_a_programar}
        return horario_a_programar

#Determina la cantidad de tecnólogos disponibles
def CuentaTecnologos():
    x = contar los tecnólogos en la tabla Personal
    return(x)

#Genera los días de espera
def GeneraDiasEspera()
    return random(0, horizonte_planeacion)

#Genera el tecnólogo que va a atender al paciente
def GeneraTecnologo()
    return random(0, max_tecnologos)
  
```

**Script 4.1** Pseudo-código para la generación de horarios.

## 4.4 Prototipo del sistema

En esta sección se presentan pantallas comprendidas en el prototipo del sistema.

### 4.4.1 Menú de mantenimiento

En esta pantalla de la Figura 4.1 se presenta el acceso a los formularios de consulta y mantenimiento de las tablas del sistema que han sido explicadas anteriormente. Como se aprecia, desde esta pantalla se puede acceder a la consulta de los valores de cada tabla, haciendo clic en el nombre de la tabla correspondiente, o según se requiera, hacer clic en el botón “Añadir” si se desea añadir un registro o en “Modificar” si se desea editar sus valores. Para acceder a la opción de “Eliminar” se debe entrar al detalle de la Tabla específica.



Figura 4.1 Pantalla de mantenimiento de tablas.

### ***Pantalla de Personal***

En esta pantalla de la Figura 4.2 se puede dar mantenimiento a la Tabla de Personal; editando, creando nuevos registros, o eliminando los registros actuales.

### ***Pantalla de Terapia***

En la pantalla de Terapia de la Figura 4.3 se presenta en primer lugar una lista de las terapias que se deben realizar en el servicio.

Al hacer clic en alguna de las terapias de la Figura 4.3, se muestra la pantalla de detalle de las Terapias de la Figura 4.4 que permite la creación, mantenimiento o eliminación de registros.

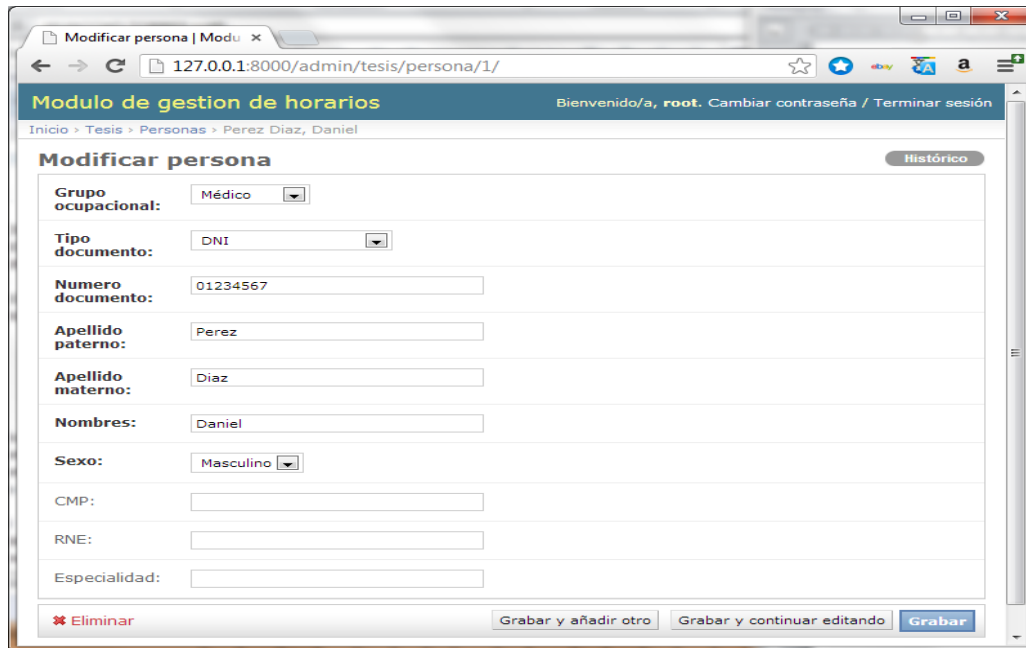


Figura 4.2 Pantalla de detalle de persona.

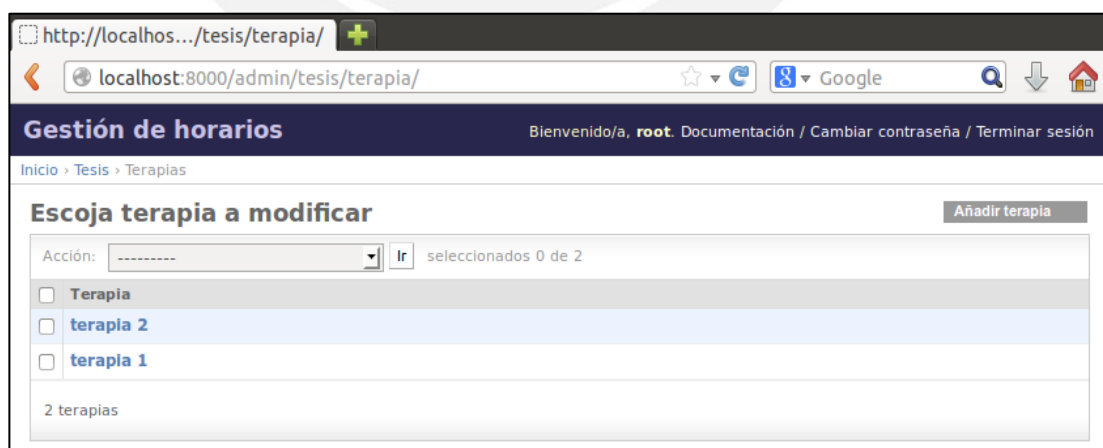


Figura 4.3 Pantalla de lista de terapias.

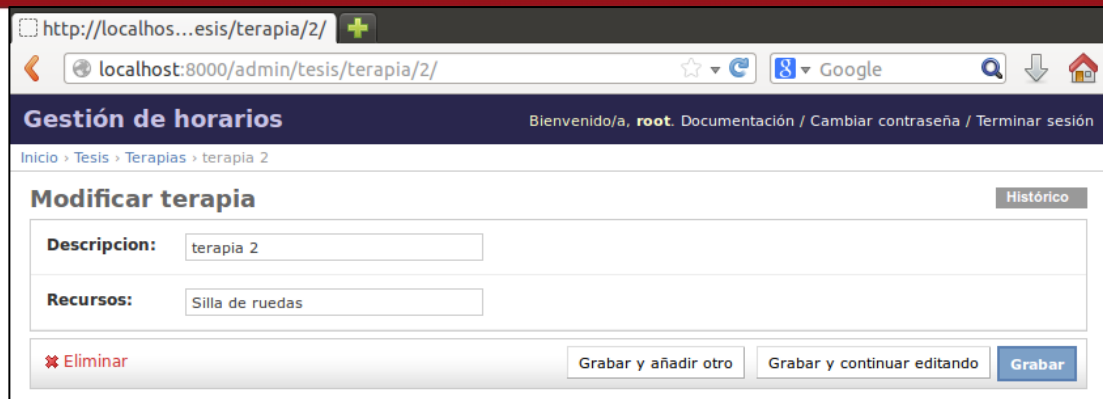


Figura 4.4 Pantalla de detalle de terapia.

### ***Pantalla de Turno***

En la pantalla de Turno que se muestra en la Figura 4.5 se crean todos los turnos posibles. En primer lugar se presenta el listado de turnos existentes.

Haciendo clic en alguna de las filas del listado de la figura 4.5, se accede al detalle correspondiente de la Figura 4.6.

En el Anexo 4 se muestra las demás pantallas que son del menú mantenimiento.

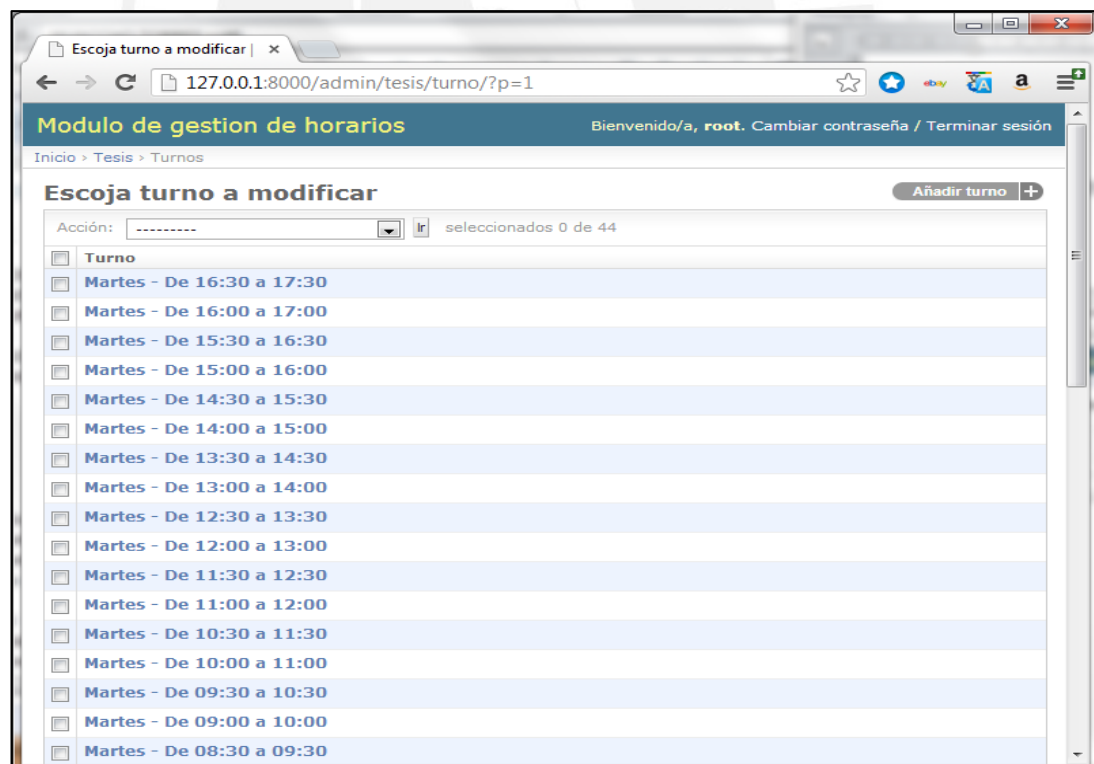


Figura 4.5 Pantalla de listado de turnos.





Figura 4.6 Pantalla de detalle de turnos.

#### 4.4.2 Módulo de configuración de horarios

En esta pantalla de la Figura 4.7 se configuran los horarios disponibles para cada profesional, en el caso que nos ocupa es especialmente importante la programación de los tecnólogos. Esta programación es utilizada como parámetro fundamental en la determinación de la asignación óptima.

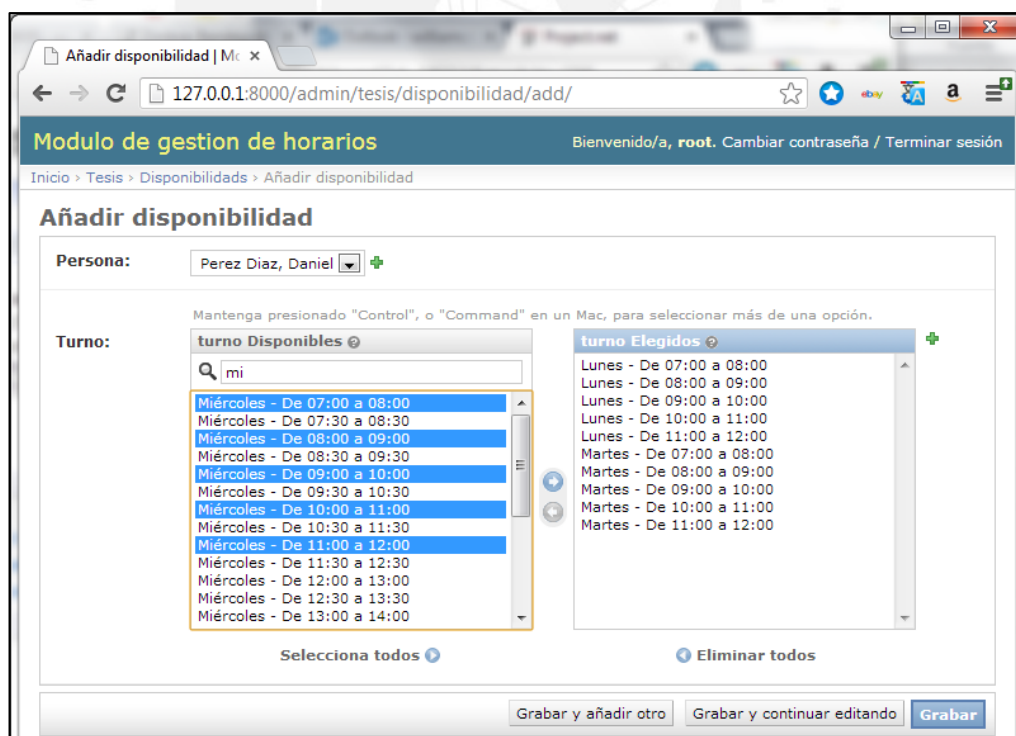
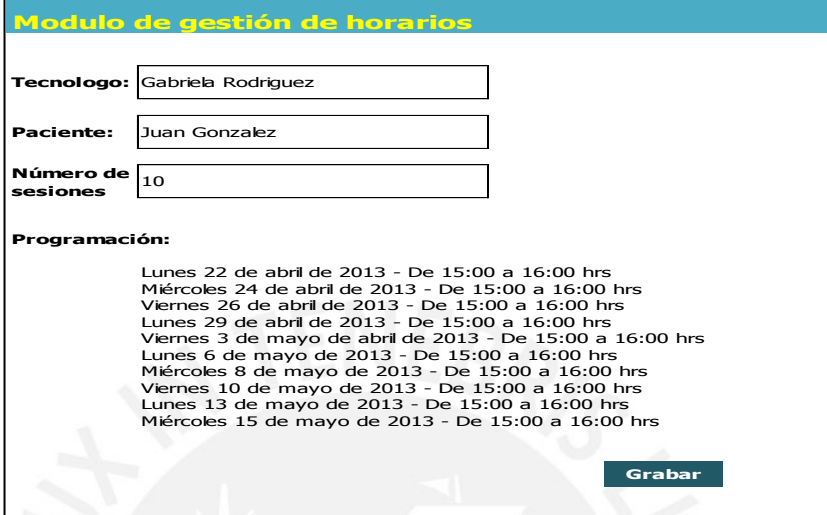


Figura 4.7 Pantalla de configuración de horarios.

#### 4.4.3 Módulo de generación de horarios

La generación de horarios se realiza no de forma interactiva con los usuarios sino en forma automática cada vez que se asigne un tratamiento a un paciente. El diseño de la pantalla correspondiente se muestra a continuación en la Figura 4.8:



**Modulo de gestión de horarios**

**Tecnologo:** Gabriela Rodriguez

**Paciente:** Juan Gonzalez

**Número de sesiones:** 10

**Programación:**

Lunes 22 de abril de 2013 - De 15:00 a 16:00 hrs  
 Miércoles 24 de abril de 2013 - De 15:00 a 16:00 hrs  
 Viernes 26 de abril de 2013 - De 15:00 a 16:00 hrs  
 Lunes 29 de abril de 2013 - De 15:00 a 16:00 hrs  
 Viernes 3 de mayo de abril de 2013 - De 15:00 a 16:00 hrs  
 Lunes 6 de mayo de 2013 - De 15:00 a 16:00 hrs  
 Miércoles 8 de mayo de 2013 - De 15:00 a 16:00 hrs  
 Viernes 10 de mayo de 2013 - De 15:00 a 16:00 hrs  
 Lunes 13 de mayo de 2013 - De 15:00 a 16:00 hrs  
 Miércoles 15 de mayo de 2013 - De 15:00 a 16:00 hrs

**Grabar**

Figura 4.8 Diseño de pantalla de generación de horarios.

#### 4.4.4 Generar archivo XML de inasistencia

Para generar archivos XML, tanto de inasistencias como de cualquier otro tipo, Django dispone de dos comandos que hacen el trabajo de forma automática:

`python manage.py loaddata nombre_de_archivo.xml`. Este comando carga los datos con formato XML que se ubican por defecto en la carpeta fixtures de la aplicación.

#### 4.4.5 Módulo de eliminación de citas

Para eliminar citas, se debe indicar a que paciente pertenecen las citas y seleccionar cuáles de ellas se desea eliminar. Debe tenerse en cuenta que la eliminación de los pacientes desertores se realiza automáticamente cuando se generan nuevos horarios, como se muestra en la Figura 4.9.

### Modulo de gestión de horarios

**Eliminación de citas:**

**Paciente:**

**Seleccione las citas a eliminar**

- Lunes 22 de abril de 2013 - De 15:00 a 16:00 hrs
- Miércoles 24 de abril de 2013 - De 15:00 a 16:00 hrs
- Viernes 26 de abril de 2013 - De 15:00 a 16:00 hrs
- Lunes 29 de abril de 2013 - De 15:00 a 16:00 hrs
- Viernes 3 de mayo de abril de 2013 - De 15:00 a 16:00 hrs
- Lunes 6 de mayo de 2013 - De 15:00 a 16:00 hrs
- Miércoles 8 de mayo de 2013 - De 15:00 a 16:00 hrs
- Viernes 10 de mayo de 2013 - De 15:00 a 16:00 hrs
- Lunes 13 de mayo de 2013 - De 15:00 a 16:00 hrs
- Miércoles 15 de mayo de 2013 - De 15:00 a 16:00 hrs

**Eliminar**

**Figura 4.9** Pantalla de eliminación de citas.

#### 4.5 Pruebas

La estrategia de pruebas se basa en comprobar las funcionalidades de la aplicación en cada una de las iteraciones que los usuarios pueden realizar con esta.

Se realizó las pruebas unitarias de las pantallas del sistema, de tal forma que los resultados obtenidos sean iguales a los esperados. Ver los resultados más adelante.

A continuación, se realizaron pruebas de integración que permitan evaluar el funcionamiento de los módulos en su conjunto, de tal forma verificar que el correcto funcionamiento observado en la etapa anterior siga observándose cuando los módulos son puestos a prueba en conjunto.

Las pruebas con el usuario final se realizaron en aquellos componentes de la aplicación con los que el usuario interactúa y aquí se verificó por ejemplo que los datos registrados en las pantallas sean grabados adecuadamente en la base de datos y que los datos que sean cargados directamente desde archivos XML también sean adecuadamente registrados en la base de datos.

Por último, de los resultados de todas las pruebas realizadas se obtiene la relación de las correcciones que son necesarias realizar, tanto a nivel del algoritmo principal de generación de horarios, como de los demás módulos. Puesto que el framework

ayuda de gran forma en la construcción de la interfaz gráfica con el usuario, se prevé que este tipo de correcciones debería ser mínimo. De hecho, la mayor carga en cantidad de correcciones a realizar estaría por el lado del algoritmo de generación de horarios y que podrían llevar inclusive a una revisión del diseño de algunas tablas del modelo o en casos necesarios, la eliminación de algunas de ellas o la creación de nuevas tablas auxiliares o intermedias.

#### 4.5.1 Pruebas unitarias

Para esta etapa se ha previsto el siguiente catálogo de pruebas para los módulos del algoritmo principal. A continuación se muestra en la Tabla 4.1 la prueba para la Pantalla de detalle de Turno. En el Anexo 5 se muestra el catálogo de pruebas completo.

***Pantalla de detalle de Turno***

Condición de entrada	Clases válidas	Clases no válidas
Día	1. Selección de Lunes 2. Selección de Martes 3. Selección de Miércoles 4. Selección de Jueves 5. Selección de Viernes 6. Selección de Sábado 7. Selección de Domingo	8. Vacío 9. Otro valor
Hora inicio	10. Caracteres numéricos y dos puntos (:) de 5 caracteres.	11. Vacío 12. Caracteres del abecedario
Hora final	13. Caracteres numéricos y dos puntos (:) de 5 caracteres.	14. Vacío 15. Caracteres del abecedario

### Pruebas unitarias

Prueba 1	
<b>Objetivo Prueba:</b>	Verificar que se ingrese los datos correctos de Turno.
<b>Precondición:</b>	Se ha accedido al sistema como usuario
<b>Descripción de la prueba:</b>	<p>En la interfaz de entrada introducir los siguientes datos:</p> <ul style="list-style-type: none"> <li>▪ Día: Selección de Lunes</li> <li>▪ Hora inicio: Caracteres numéricos y dos puntos (:) de 5 caracteres.</li> <li>▪ Hora final: Caracteres numéricos y dos puntos (:) de 5 caracteres.</li> </ul>
<b>Resultados Esperados:</b>	Se muestra el mensaje: Se añadió con éxito el turno "xxx "

Prueba 2	
<b>Objetivo Prueba:</b>	Verificar que se ingrese los datos correctos de Turno.
<b>Precondición:</b>	Se ha accedido al sistema como usuario
<b>Descripción de la prueba:</b>	<p>En la interfaz de entrada introducir los siguientes datos:</p> <ul style="list-style-type: none"> <li>▪ Día: Selección de Lunes</li> <li>▪ Hora inicio: Vacío</li> <li>▪ Hora final: Caracteres numéricos y dos puntos (:) de 5 caracteres.</li> </ul>
<b>Resultados Esperados:</b>	Se muestra el mensaje: Por favor, corrija el siguiente error: Este campo es obligatorio

Prueba 3	

<b>Objetivo Prueba:</b>	Verificar que se ingrese los datos correctos de Turno.
<b>Precondición:</b>	Se ha accedido al sistema como usuario
<b>Descripción de la prueba:</b>	<p>En la interfaz de entrada introducir los siguientes datos:</p> <ul style="list-style-type: none"> <li>▪ Día: Selección de Lunes</li> <li>▪ Hora inicio: Caracteres numéricos y dos puntos (:) de 5 caracteres.</li> <li>▪ Hora final: Vacío</li> </ul>
<b>Resultados Esperados:</b>	<p>Se muestra el mensaje: Por favor, corrija el siguiente error:</p> <p>Este campo es obligatorio</p>

**Tabla 4.1** Relación de pruebas unitarias.

#### 4.5.2 Pruebas de integración

Las pruebas de integración evalúan el desempeño de la función GeneraHorarios, que contiene a las funciones descritas en la sección precedente en la obtención de respuestas factibles en un tiempo razonable y como se integran estos resultados con la presentación que deba hacerse de estos.

La cantidad de pruebas de este tipo a realizar depende de los escenarios que se han planteado en las pruebas anteriores.

#### 4.5.3 Pruebas con el usuario final

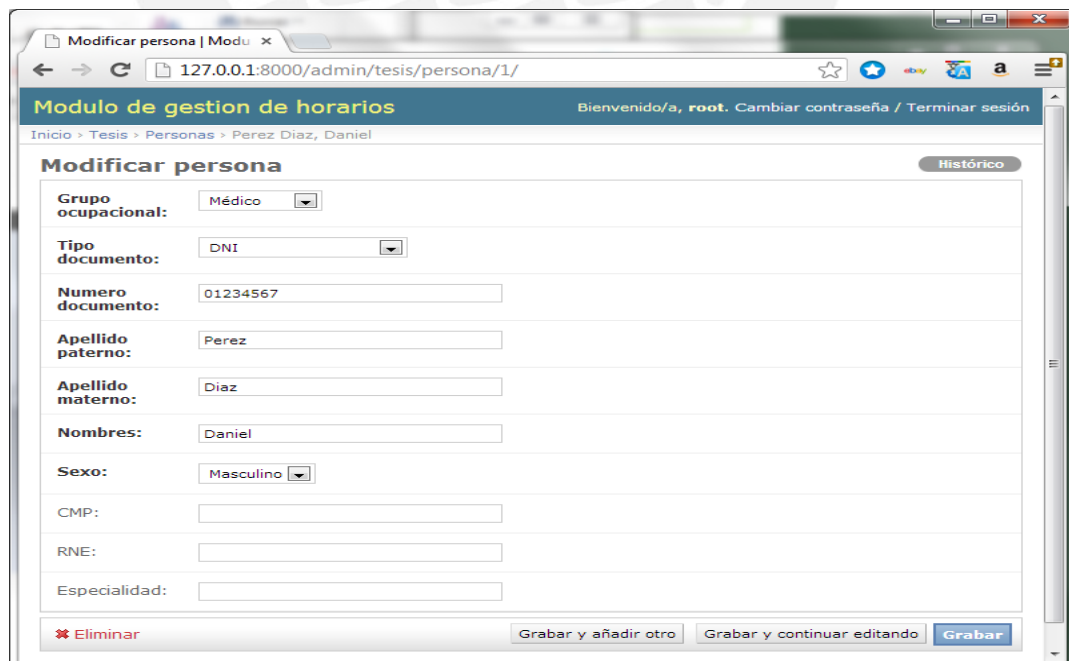
En este tipo de pruebas se verifica que la data ingresada por los usuarios en las pantallas del sistema.

Se han definido dos tipos de pruebas básicas, que se presentan en la siguiente Tabla 4.2

Tipo de prueba	Descripción	Resultado
Los datos ingresados / mostrados por los formularios coinciden con los valores registrados en la base de datos.	Prueba con la pantalla de Personal: comparación de valores de la pantalla de la aplicación con los valores de la Tabla correspondiente en MySql.	Coinciden
	Prueba con la pantalla de Turnos: comparación de valores de la pantalla de la aplicación con los valores de la tabla correspondiente en MySql	Coinciden
Los datos cargados desde un archivo XML coinciden con los valores registrados en la base de datos	Se carga un archivo XML conteniendo los valores de la Tabla CIE10 y se verifica si coinciden con los valores de la tabla correspondiente en MySql	Coinciden

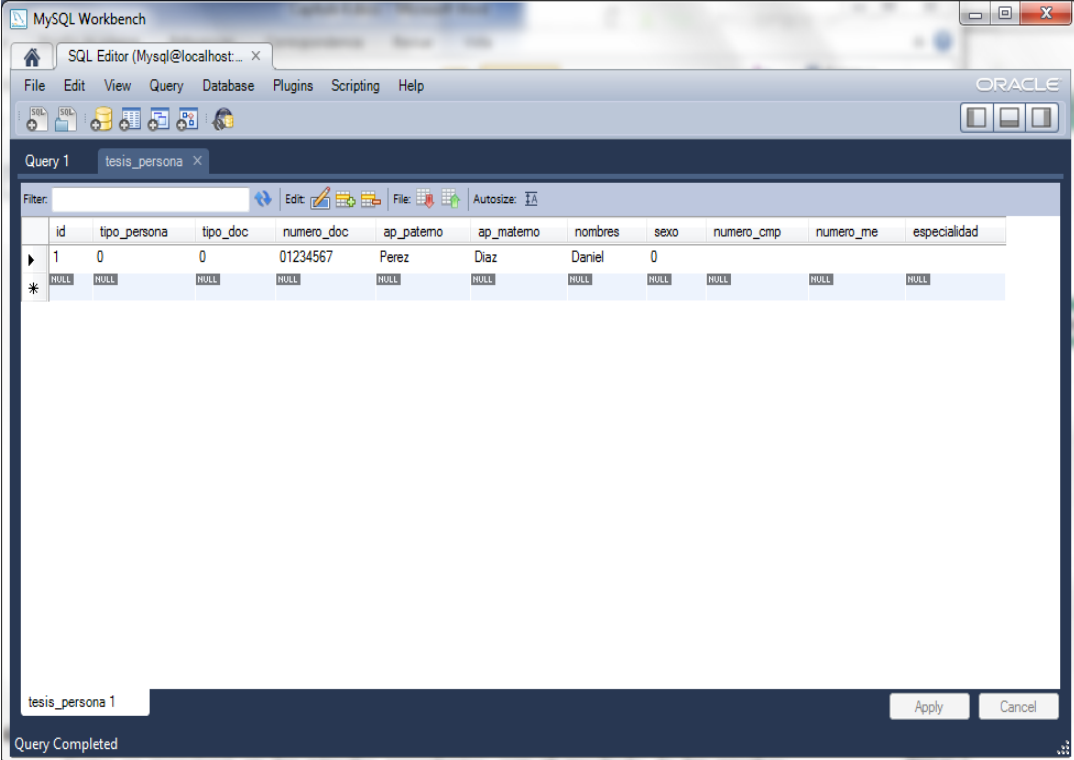
**Tabla 4.2** Tipos de prueba con el usuario final

Para la prueba correspondiente a la pantalla de Personal de la Figura 4.10, se han registrado los siguientes datos:



**Figura 4.10** Pantalla de registro de personal.

Para verificar si estos datos han sido adecuadamente grabados en la Tabla correspondiente de MySQL, se realiza una consulta y los resultados obtenidos en el Workbench se presentan a continuación en la Figura 4.11:



id	tipo_persona	tipo_doc	numero_doc	ap_patemo	ap_matemo	nombres	sexo	numero_cmp	numero_me	especialidad
1	0	0	01234567	Perez	Diaz	Daniel	0			
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

**Figura 4.11** Pantalla del Workbench de MySQL mostrando el contenido de la Tabla Personal.

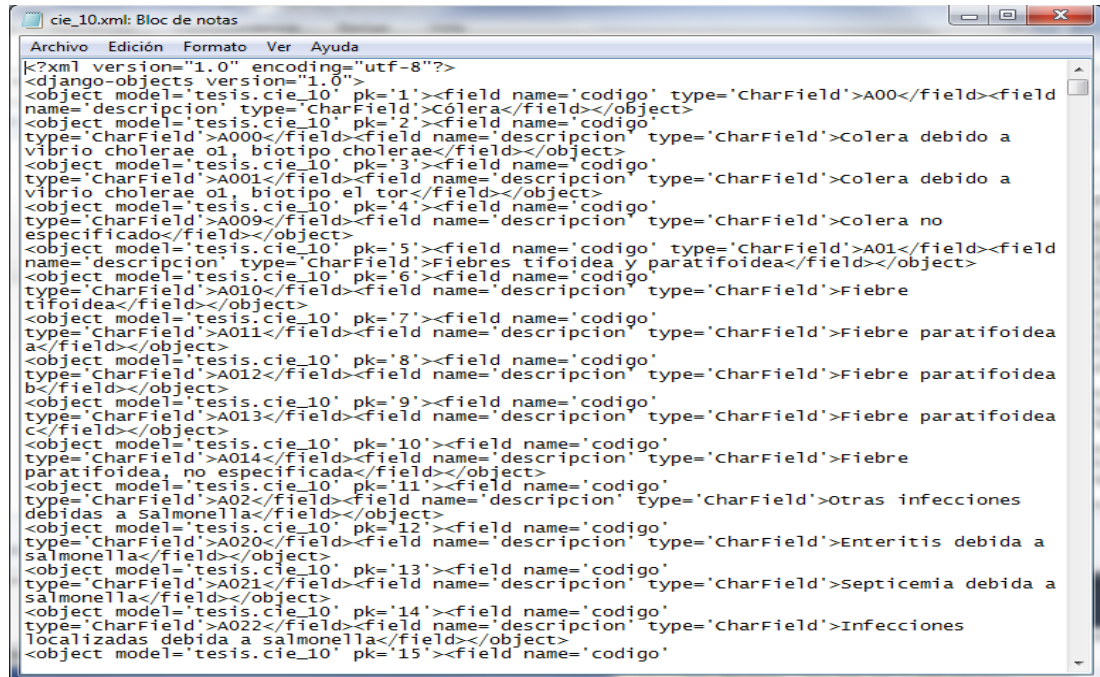
Como se aprecia en las dos figuras anteriores, los datos tanto de la pantalla de carga como del contenido de la tabla en la base de datos coinciden, demostrándose que la pantalla de Registro de Personal está funcionando adecuadamente.

Se puede apreciar que ambos contenidos coinciden, y podemos decir a manera de conclusión que la aplicación está mostrando adecuadamente el contenido de la base de datos (no hay que perder de vista que el id correspondiente a los días martes es 1). Por otro lado, se realizaron pruebas para verificar los datos cargados en formato XML. En este caso, debido a la complejidad de los datos y sobre todo a su cantidad, se trabajó con la Tabla CIE10. Una muestra de los datos que debían ser cargados a la base de datos se presenta en la siguiente Figura 4.12.



Luego de efectuada la importación de datos, la Tabla de MySql presenta el siguiente contenido de la Figura 4.13.

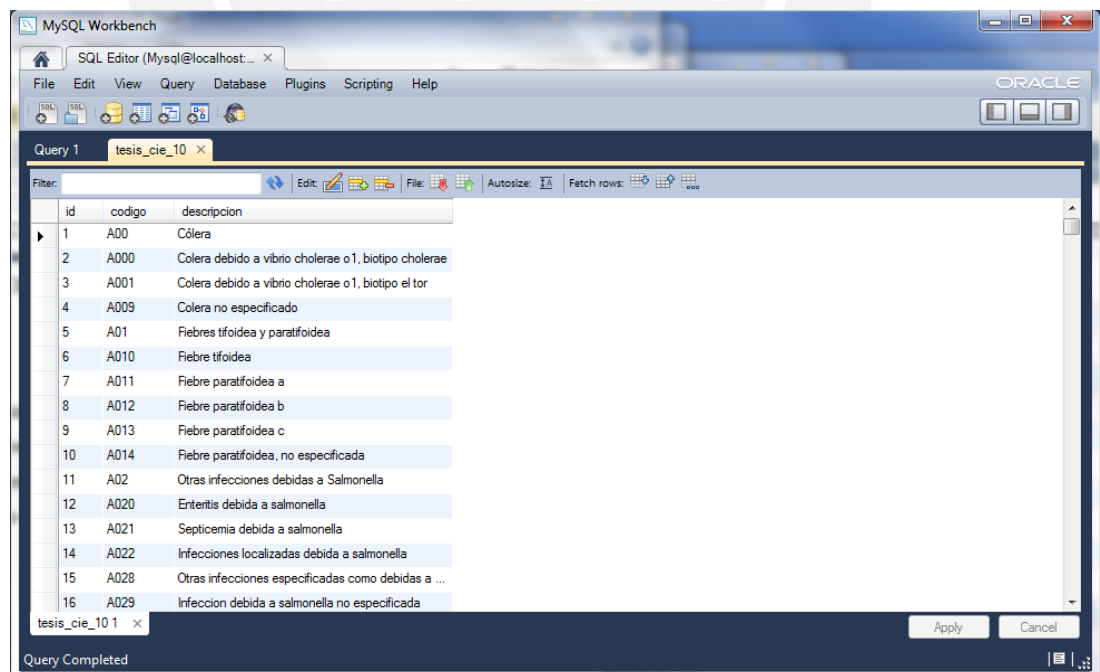
Se aprecia que ambos contenidos coinciden, entonces afirmamos que esta prueba ha sido exitosa.



```

Archivo Edición Formato Ver Ayuda
<?xml version="1.0" encoding="utf-8"?>
<django-objects version="1.0">
<object model='tesis.cie_10' pk='1'><field name='codigo' type='CharField'>A00</field><field
name='descripcion' type='CharField'>Cólera</field></object>
<object model='tesis.cie_10' pk='2'><field name='codigo'
type='CharField'>A000</field><field name='descripcion' type='CharField'>Colera debido a
vibrio cholerae o1, biotipo cholerae</field></object>
<object model='tesis.cie_10' pk='3'><field name='codigo'
type='CharField'>A001</field><field name='descripcion' type='CharField'>Colera debido a
vibrio cholerae o1, biotipo el tor</field></object>
<object model='tesis.cie_10' pk='4'><field name='codigo'
type='CharField'>A009</field><field name='descripcion' type='CharField'>Colera no
especificado</field></object>
<object model='tesis.cie_10' pk='5'><field name='codigo' type='CharField'>A01</field><field
name='descripcion' type='CharField'>Fiebres tifoidea y paratifoidea</field></object>
<object model='tesis.cie_10' pk='6'><field name='codigo'
type='CharField'>A010</field><field name='descripcion' type='CharField'>Fiebre
tifoidea</field></object>
<object model='tesis.cie_10' pk='7'><field name='codigo'
type='CharField'>A011</field><field name='descripcion' type='CharField'>Fiebre paratifoidea
a</field></object>
<object model='tesis.cie_10' pk='8'><field name='codigo'
type='CharField'>A012</field><field name='descripcion' type='CharField'>Fiebre paratifoidea
b</field></object>
<object model='tesis.cie_10' pk='9'><field name='codigo'
type='CharField'>A013</field><field name='descripcion' type='CharField'>Fiebre paratifoidea
c</field></object>
<object model='tesis.cie_10' pk='10'><field name='codigo'
type='CharField'>A014</field><field name='descripcion' type='CharField'>Fiebre
paratifoidea, no especificada</field></object>
<object model='tesis.cie_10' pk='11'><field name='codigo'
type='CharField'>A02</field><field name='descripcion' type='CharField'>Otras infecciones
debidas a Salmonella</field></object>
<object model='tesis.cie_10' pk='12'><field name='codigo'
type='CharField'>A020</field><field name='descripcion' type='CharField'>Enteritis debida a
salmonella</field></object>
<object model='tesis.cie_10' pk='13'><field name='codigo'
type='CharField'>A021</field><field name='descripcion' type='CharField'>Septicemia debida a
salmonella</field></object>
<object model='tesis.cie_10' pk='14'><field name='codigo'
type='CharField'>A022</field><field name='descripcion' type='CharField'>Infecciones
localizadas debida a salmonella</field></object>
<object model='tesis.cie_10' pk='15'><field name='codigo'
    
```

Figura 4.12 Archivo de datos CIE10 en formato XML.



id	codigo	descripcion
1	A00	Cólera
2	A000	Colera debido a vibrio cholerae o1, biotipo cholerae
3	A001	Colera debido a vibrio cholerae o1, biotipo el tor
4	A009	Colera no especificado
5	A01	Fiebres tifoidea y paratifoidea
6	A010	Fiebre tifoidea
7	A011	Fiebre paratifoidea a
8	A012	Fiebre paratifoidea b
9	A013	Fiebre paratifoidea c
10	A014	Fiebre paratifoidea, no especificada
11	A02	Otras infecciones debidas a Salmonella
12	A020	Enteritis debida a salmonella
13	A021	Septicemia debida a salmonella
14	A022	Infecciones localizadas debida a salmonella
15	A028	Otras infecciones especificadas como debidas a ...
16	A029	Infeccion debida a salmonella no especificada

Figura 4.13 Contenido de la Tabla CIE10 en MySql.

#### 4.6 Correcciones finales

Como se mencionó en los párrafos anteriores, con el resultado de las pruebas obtenidas se procede a realizar las correcciones necesarias en la codificación o en el diseño. Luego, para verificar si estas modificaciones han corregido efectivamente el resultado obtenido, se deben volver a ejecutar las pruebas para ver si cumplen o no con lo necesario.



## **5. Capítulo V: Observaciones, conclusiones y recomendaciones**

En este capítulo, parte final del proyecto, se presentan las observaciones más importantes realizadas durante el trabajo, así como las conclusiones a las que se han llegado en relación a los objetivos planteados en el Capítulo 1. Por último, se presentan algunas recomendaciones a ser consideradas por la Dirección del hospital.

### **5.1 Observaciones**

Se puede resumir las siguientes observaciones realizadas durante la ejecución del proyecto:

- 1) Actualmente la asignación de citas a los pacientes del servicio de Medicina Física y Rehabilitación del Hospital San José se realiza en forma manual; luego de un dialogo entre el paciente y el tecnólogo que le fue asignado.

- 2) En caso de la utilización de recursos compartidos como cubículos, camillas, sillas de ruedas, equipos; se necesita modificar el producto para agregar la estructura y funcionalidad de disponibilidad en horas para cada uno de los recursos similar a la de los tecnólogos.
- 3) Para poder realizar una mejora en los tiempos de atención y en la eficiencia del uso de recursos, es necesario que se cambie el modelo actual por un proceso automatizado que, basado en el objetivo de minimizar el tiempo de espera del paciente para su primera atención y que a la vez permita mejorar la utilización de los recursos; reduzca los “tiempos muertos” de los tecnólogos y equilibre la carga de trabajo mensual de estos.
- 4) El hospital no cuenta con presupuesto orientado a este tipo de soluciones; es por ello, que la solución planteada debe ser lo menos onerosa posible y utilizar los recursos y la base tecnológica con la que se cuenta en la actualidad.

## 5.2 Conclusiones

Las conclusiones están relacionadas a los objetivos planteados en este proyecto:

- 1) Se ha logrado implementar un sistema de información que cumple con los requisitos planteados y que permite la asignación de citas minimizando el tiempo de espera de los pacientes para acceder a su primera terapia. Además, permite un adecuado control de los pacientes desertores; es decir, aquellos que a pesar de contar con un cronograma de atención, no acuden oportunamente a sus tratamientos.
- 2) Se han definido las variables y estructuras de datos necesarias para el algoritmo de búsqueda local, que permiten generar los horarios de atención de cada tecnólogo. El detalle de estas variables y el algoritmo completo de optimización se han presentado en el Capítulo 4.
- 3) Se ha implementado el algoritmo de optimización Hill Climbing que resuelve el problema de programación dinámica de horarios para citas de terapia física y rehabilitación, basado en cálculo de una función de mérito que está conformada por el total de días de espera antes de acceder a la primera

terapia; además, de otros aspectos tales como la preferencia de cada tecnólogo por determinado tipo de tratamiento, asignación previa del tecnólogo a otro paciente, los turnos de trabajo disponibles, y las posibles modificaciones que estos podrían tener.

- 4) Se diseñaron los formatos necesarios para importar o exportar la información relacionada a los horarios vigentes y eliminados, las inasistencias del personal, y la actualización de la deserción de los pacientes.
- 5) Se ha diseñado una interfaz gráfica en Web para ingresar las restricciones del problema, así como los parámetros y valores que permiten generar y almacenar una o más planificaciones.

### 5.3 Recomendaciones

Se pueden realizar las siguientes recomendaciones:

- 1) El algoritmo, desarrollado en Python 2.7, tiene algunos parámetros que han sido definidos y probados durante la ejecución del proyecto pero que probablemente requieran un reajuste o afinación en el momento en que la herramienta entre a la fase de producción. Entre estos parámetros tenemos el periodo que se utiliza para generar los tiempos de espera iniciales, el máximo de iteraciones que deben realizar hasta antes de detener la ejecución del programa.
- 2) Medir la satisfacción y la mejora en la percepción del servicio que tendría el paciente luego de aplicada esta modificación, para ver si esta habrá mejorado con respecto a la situación vigente.
- 3) También se debe promocionar y explicar el nuevo proceso a los pacientes para evitar cualquier tipo de reclamo o inconveniente y lograr convencerlos de que los cambios se reflejarán en mejores niveles de servicio para ellos.

- 4) Luego de realizar todas las pruebas de seguridad necesaria, publicar el acceso a la aplicación desde puntos externos a la red local.
- 5) Publicar el software resultante bajo una licencia de acceso público para permitir su utilización en otras instituciones similares. Esto podría ser una oportunidad de negocio al generarse necesidades de consultoría para la instalación, configuración y personalización de la herramienta.
- 6) Analizar la pertinencia de otorgar acceso remoto de los pacientes a sus propios cronogramas. Para esto sería necesario que la máquina en la que este corriendo el sistema tenga una dirección IP pública. El análisis de las consideraciones relativas a la seguridad asociada a esta recomendación, escapan al alcance de este proyecto.



## Referencias bibliográficas

- [1]. ONGEI, Oficina Nacional de Gobierno Electrónico e Informática  
2008 *Resolución Ministerial N° 381-2008-PCM.Estándares y Especificaciones de Interoperabilidad del Estado Peruano*. 13 de noviembre del 2008.
- [2]. DEARDEN, Richard  
2009 *Intro to AI: Probabilistic AI Search Planning*.
- [3]. CLAUSEN, Jensen  
1999 "Branch and Bound Algorithms -Principles and Examples".
- [4]. INSTITUTE PROJECT MANAGEMENT  
2004 *Guía de los Fundamentos de la Dirección de Proyectos*.
- [5]. KRUCHTEN, Philipp  
2001 *The Rational Unified Process, An Introduction*.
- [6]. GARCÍA MUÑOZ, José  
2010 *Estudio sobre la Implantación de un Sistema de Información [adaptado a las necesidades de los Establecimientos de Salud rurales del Perú]*. Proyecto fin de Máster en Redes de Telecomunicación para países en desarrollo. Universidad Rey Juan Carlos, Escuela Técnica Superior de Ingeniería de Telecomunicación.
- [7]. Díaz, A., Glover. F., Ghaziri, H. M., González, J. L., Laguna, M., Moscato, P. y Tseng, F. T.  
1996 *Optimización Heurística y Redes Neuronales*.

[8]. SCHAERF, Andrea<sup>1</sup> y Amnon MEISELS<sup>2</sup>

"Solving Employee Timetabling Problems by Generalized Local Search. Dipartimento di Ingegneria Elettrica, Gestionale e Meccanica, Università di Udine, Via delle Scienze 208, 33100 Udine, Italy <sup>1</sup> Department of Mathematics and Computer Science Ben-Gurion University of the Negev, Beer-Sheva, 84-105, Israel<sup>2</sup>".

[9]. SHOSHANA, Hahn-Goldberg, Michael W. CARTER, J. Christopher BECK

2011 "Dynamic template scheduling to address uncertainty in complex scheduling problems: a case study on chemotherapy outpatient scheduling". University of Toronto, Department of Mechanical and Industrial Engineering.

[10].KNOWLTON, Jim

2009. *Python*. 1ra edición. Roma: Anaya Multimedia-Anaya Interactiva.

[11].MySQL

2013 *Dispelling the Myths*. Consulta: 10 de Abril del 2013.  
<<http://bugs.mysql.com/file.php?id=2049>>.

[12].MINSA, Hospital San Jose Del Callao

2008 "Manual de Organizacion y funciones del Hospital San Jose del Callao". Tomo I: pp. 26.

2008 "Nº 022-MINSA/DGSP-V.02. Norma Técnica de Salud para la Gestión de la Historia Clínica. 2 de Octubre del 2008".

2010 "Manual de Normas y Procedimientos del servicio de Medicina Fisica y Rehabilitación."



2012 Clasificación Internacional de Enfermedades 10°.

Consulta: 10 de Diciembre del 2012.

<<http://cie10.org/>>

[13].ÁNGEL

2012 ÁNGEL. Manual Angel 3.Sistema Integral de Administración de Salud. Consulta: 10 de Septiembre del 2012.

<<http://www.proyectoangel.net/ManualAngel/WebHelp/index.htm>>

[14].QSOFT, Expertos en Software para la salud

2012 SALUS. Consulta: 8 de Agosto del 2012

<[http://www.softwaresalus.com/descripcion\\_software\\_salus.aspx](http://www.softwaresalus.com/descripcion_software_salus.aspx)>.

[15].OPTIHPER

2012 Asignación Optimizada de Horarios al Personal

<<http://users.dsic.upv.es/grupos/gps/optihper/>>.