



Universidad Nacional  
**ARTURO JAURETCHE**

INGENIERÍA DE SOFTWARE I

---

## **Trabajo Final Integrador**

---

*Autor*

Emiliano SALVATORI

*Profesor*

Dr. Sergio Daniel CONDE

24 de noviembre de 2020

# Índice

<b>1. Introducción</b>	<b>4</b>
<b>2. Problemática</b>	<b>4</b>
2.1. Caso de Estudio . . . . .	4
2.2. Realizar . . . . .	4
<b>3. Tipo de Proyecto</b>	<b>5</b>
3.1. Según el grado de dificultad . . . . .	6
3.2. Según la procedencia del capital . . . . .	6
3.3. Según el contenido del proyecto . . . . .	6
3.4. Según los involucrados en la organización . . . . .	6
3.5. Según su finalidad . . . . .	6
<b>4. Estudio de Factibilidad</b>	<b>6</b>
4.1. Costes y alcances . . . . .	7
4.2. Factibilidad Técnica . . . . .	7
4.2.1. Software . . . . .	8
4.2.2. Servidor . . . . .	8
4.2.3. Modems y Redes . . . . .	9
4.3. Factibilidad Operativa . . . . .	9
4.3.1. Equipo de Trabajo . . . . .	9
4.3.2. Capacitaciones . . . . .	10
<b>5. Informe sobre Factibilidad escogida</b>	<b>11</b>
5.1. Costos totales sobre la Ponderación . . . . .	11
5.2. Informe Final . . . . .	11
<b>6. Metodología y Alcance del Proyecto</b>	<b>12</b>
6.1. Metodología . . . . .	12
6.2. Alcance del Proyecto . . . . .	13
<b>7. Requerimientos Funcionales y No Funcionales</b>	<b>13</b>
7.1. Requerimientos Funcionales . . . . .	13
7.2. En base al Paciente . . . . .	13
7.3. En base a la Reserva de Turnos . . . . .	15
7.4. En base a las Historias Clínicas . . . . .	16
7.5. En base a la Internación de Pacientes . . . . .	18
7.6. En base a las camas disponibles . . . . .	20
7.7. Requerimientos No Funcionales . . . . .	21
<b>8. Casos de Uso</b>	<b>23</b>

<b>9. Métricas</b>	<b>25</b>
9.1. Primer Sprint y el Backlog Item . . . . .	25
9.2. Consumo de horas por semana . . . . .	25
9.2.1. Primera semana . . . . .	26
9.2.2. Segunda semana . . . . .	26
9.2.3. Tercer semana . . . . .	26
9.3. Segundo Sprint y su Backlog Item . . . . .	26
9.4. Consumo de horas por semana . . . . .	27
9.4.1. Primera semana . . . . .	27
9.4.2. Segunda semana . . . . .	27
9.4.3. Tercer semana . . . . .	27
9.5. Tercer Sprint y su Backlog Item . . . . .	28
9.6. Consumo de horas por semana . . . . .	28
9.6.1. Primera semana . . . . .	29
9.6.2. Segunda semana . . . . .	29
9.6.3. Tercer semana . . . . .	29
<b>10. Selección de Ciclo de Vida</b>	<b>29</b>
<b>11. Gestión y Plan de Riesgo</b>	<b>30</b>
11.1. Valorización de la Disponibilidad . . . . .	30
11.2. Valorización de la Integridad . . . . .	31
11.3. Valorización de la Confidencialidad . . . . .	31
11.4. Inventario de Activos . . . . .	31
11.5. Valoración de activos . . . . .	31
11.6. Análisis y estimación de riesgos . . . . .	32
<b>12. Diseño del Sistema</b>	<b>32</b>
12.1. Modelo del Repositorio . . . . .	33
12.2. Modelo Cliente/Servidor . . . . .	34
12.3. Modelo de Capas . . . . .	36
<b>13. Diagrama de Clases</b>	<b>37</b>
<b>14. Modelo de Control Centralizado</b>	<b>37</b>
<b>15. Plan de Pruebas</b>	<b>38</b>
15.1. Pruebas de Fiabilidad . . . . .	38
15.2. Pruebas de Software . . . . .	39
15.3. Pruebas de Interface . . . . .	40
15.4. Fases de una Prueba de Interface . . . . .	41
15.4.1. Configuración y Desarrollo . . . . .	41
15.4.2. Validaciones . . . . .	41
15.4.3. Mantenimiento . . . . .	41
15.5. ¿Por qué implementar Pruebas de Interface? . . . . .	41

15.5.1. Directivas para una Interfaz de Usuario . . . . .	42
<b>16. Esquema de Integración</b>	<b>42</b>
16.1. Conclusión . . . . .	45
<b>Bibliografía</b>	<b>45</b>

## 1. Introducción

En el siguiente informe se detalla lo realizado como parte práctica de la materia **Ingeniería de Software I** para la **Comisión nº 1**. El presente trabajo se basa en los conceptos fundamentales sobre cómo abordar y desarrollar un **Trabajo Integral** en base a las necesidades surgidas de un caso práctico.

## 2. Problemática

### 2.1. Lea el siguiente Caso de Estudio

Nos encontramos en el País de Argentur que como consecuencia de la Pandemia que afecta a nivel mundial tiene muchas necesidades que vamos a desarrollar a continuación:

Argentur tiene las áreas de salud, infraestructura y educación con problemáticas diversas. En Salud nos encontramos con falta de instrumentación e infraestructura hospitalaria. Se requiere desarrollar un Proyecto que comprenda a los siguientes ítems:

- Los Servidores para el área de Salud.
- Desarrollar una Página Web.
- Comprar Modems y los elementos necesarios para la instalación de una red.
- Contratación de Recursos Humanos para realizar el trabajo.

Las Áreas Críticas que necesita es Reserva de Turnos con Registro, Consulta, modificación, listado y baja de Turnos.

Consultorios médicos para tener las historias clínicas con Registro, Consulta, modificación, listado y baja de Historias Clínicas de los pacientes.

Internación con Registro, Consulta, modificación, listado y baja de Internación de Pacientes. Es necesario tener en consultar si existen camas disponibles en internación

### 2.2. Realizar

1. Identificar el Tipo de Proyecto
2. Realizar cada una de las Factibilidades Técnicas con tres opciones por factibilidad técnica.
3. Realizar cada una de las Factibilidades Operativa con tres opciones por factibilidad operativa.
4. Realizar la Tabla de Ponderación de cada Factibilidad.
5. Realizar el Informe Parcial Detallado de cada Factibilidad elegida justificando la elección.

6. Realizar el Costo Total de todas las Factibilidades teniendo en cuenta la Ponderación realizada.
7. Realizar el Informe Final detallado.
8. Definir el Alcance del Proyecto
9. Definir una Metodología Scrum contemplando tres Sprints para desarrollar el Proyecto.
10. Definir Requerimientos Funcionales y No Funcionales utilizando el modelo SRS-830.
11. Desarrollar los Casos de Uso que correspondan.
12. Seleccionar el Ciclo de Vida a utilizar y justificar con un informe detalladamente la elección del mismo.
13. Elaborar un Plan de Riesgos aplicando el modelo PDCA con las acciones a tomar. Debe haber un mínimo de 5 riesgos.
14. Aplique en el Diseño del Sistema:
  - El Modelo de Repositorio detallando como va a utilizar cada componente
  - El Modelo Cliente/Servidor
  - El Modelo de Capas.
15. Desarrollo un Diseño utilizando la Descomposición Orientada a Objetos. (Desarrolle un Diagrama de Clases completo de todo el Proyecto con atributos y métodos.
16. Aplique el Modelo de Gestor.
17. Desarrolle un Plan de Pruebas y Testing detallados.
18. Desarrolle un esquema de Integración teniendo en cuenta su propuesta y explique el esquema detalladamente.

### **3. Tipo de Proyecto**

Las características de un proyecto, pueden ser variar o no en el tiempo según cuándo se evalúe, por lo que pueden ser dinámicas (cambian en lapsos de tiempo muy cortos, es decir, son de plazos más cortos) o más bien pasivos (no cambian en el mediano/largo plazo); lo que cabe resaltar es que se pueden caracterizar el tipo de proyecto a abordar de varias maneras.

En el siguiente trabajo, lo abordaremos desde distintas perspectivas para poder acotar el alcance y de esta manera definir mejor la magnitud o el peso del mismo. Se tratará de definir el tipo de proyecto a analizar según determinadas características.

### 3.1. Según el grado de dificultad

Según esta propiedad, el tipo de proyecto sería **Complejo** ya que la red que relaciona sus actividades es amplia y compleja. Muchas de las tareas que lo comprenden no se pueden ejecutar hasta que otras no estén finalizadas.

Para este tipo de proyectos es clave que se utilice una aplicación capaz de simplificar todo, como puede ser Sinnaps, la cual calcula la mejor ruta de trabajo, aprovechando todos los recursos disponibles y en función a esta red compleja de relaciones entre las actividades.

### 3.2. Según la procedencia del capital

Según esta propiedad, el tipo de proyecto sería **Públicos**, ya que su financiación es procedente de instituciones gubernamentales, más específicamente de Argentur.

### 3.3. Según el contenido del proyecto

Según esta característica, el tipo de proyecto sería de **Informática** ya que se centra específicamente en el desarrollo de software, sistemas de tecnologías de información, etc. Para ello se suelen utilizar técnicas ágiles de gestión, las cuales pueden ser: Scrum y Kanban.

Aunque también podría ser de un tipo de proyecto sobre **Producción de servicios**, ya que se solicita el desarrollo o servicio innovador, diseño de un nuevo producto. Aquí entraría la diferencia entre proyecto y proceso, por lo que deberíamos tener claro qué es un proyecto. Por eso, suelen usarse en departamento I+D para el diseño y la ejecución de los prototipos.

### 3.4. Según los involucrados en la organización

Según esta característica, el tipo de proyecto sería de **Subcontratado o externo** ya que la entidad gubernamental a cargo contrata a una consultora externa o un Interim Manager para la ejecución del proyecto de software.

### 3.5. Según su finalidad

Según esta característica, el tipo de proyecto sería de **Producción** ya que se encuentra orientado a producir un servicio bajo un objetivo determinado.

## 4. Estudio de Factibilidad

Para el proyecto que analizado, se realiza una estimación sobre las necesidades identificadas del usuario se cubren con las actuales tecnologías de software y hardware. El estudio considerará si el sistema propuesto tendrá un costo/beneficio desde un

punto de vista gubernamental, y si éste puede desarrollarse dentro de las restricciones presupuestales existentes.

Un estudio de factibilidad debe ser rápido y relativamente barato. El resultado debe informar la decisión respecto a si se continúa o no continúa con un análisis más detallado. [1]

#### 4.1. Costes y alcances

Para el presente proyecto de Software en el área de salud, se cuenta con los siguientes presupuestos en moneda argentina, segregados por área de impacto:

Presupuesto estimado	Área de impacto
Área Técnica (Software)	\$ 50.000
Área Técnica (Servidor)	\$ 1.550.000
Área Técnica (Red)	\$ 1.000.000
Área Operativa (Equipo de Trabajo)	\$ 550.000
Área Operativa (Capacitaciones)	\$ 10.000
<b>Presupuesto Total</b>	<b>\$ 3.160.000,00</b>

Para el alcance del proyecto, se estableció en primera instancia, que tenga una duración *no mayor a tres meses* (aproximadamente 4 sprints bajo la metodología Scrum). Esto permite tener tanto metas claras para el comienzo del proyecto, como también especificar los distintos factores de la puesta en marcha, centrarse de forma más focalizada en los puntos débiles (punto que sería difícil de alcanzar en caso de tener un alcance de mediano o largo plazo), como también consolidar el nuevo equipo de desarrollo.

#### 4.2. Factibilidad Técnica

A continuación se realiza un estudio de Factibilidad Técnica para distintas áreas, a saber:

- **Software:** Evaluación de la tecnología necesaria para llevar a cabo el proceso de configuración de los Servidores y el Desarrollo Web.
- **Hardware:** Evaluación de los equipos necesarios para llevar a cabo la instalación de los Servidores, y los equipos necesarios para el desarrollo de software.
- **Redes de Datos:** Evaluación de la tecnología necesaria para poder realizar el tendido y el cableado necesario para interconectar el centro de cómputo de los servidores, con las distintas áreas de desarrollo que conformaría el proyecto de software.



#### 4.2.1. Software

-	Peso	Redux	React	Angular
Aprendizaje	30 %	60	30	70
Complejidad	30 %	30	30	45
Portabilidad:	10 %	100	100	100
Diseño:	10 %	80	100	100
Reutilidad:	7 %	100	100	100
Costo:	13 %	50	40	60

La ponderación se realiza de la siguiente forma:

- **Redux:**  $30 * 60 + 30 * 30 + 10 * 100 + 10 * 80 + 7 * 100 + 13 * 50 = 5850$
- **React:**  $30 * 30 + 30 * 30 + 10 * 100 + 10 * 100 + 7 * 100 + 13 * 40 = 5020$
- **Angular:**  $30 * 70 + 30 * 45 + 10 * 100 + 10 * 90 + 7 * 100 + 13 * 60 = 6830$

**Informe de decisión:** A raíz de la ponderación evaluada, se toma como referencia la tecnología **Angular** como Framework para la creación de la página web ya que es más robusto a pesar de su alto costo de aprendizaje.

#### 4.2.2. Servidor

-	Peso	HP X3450	Qnap TS430	IBM X3700
CPU	28 %	50	40	50
RAM	20 %	40	30	100
Copias:	12 %	10	5	5
Servidores:	5 %	0	0	0
Expansión:	10 %	10	20	100
Costo:	25 %	70	80	90

La ponderación se realiza de la siguiente forma:

- **HP X3450:**  $28 * 50 + 20 * 40 + 12 * 10 + 5 * 100 + 10 * 0 + 25 * 70 = 4570$
- **Qnap TS430:**  $28 * 40 + 20 * 30 + 12 * 5 + 5 * 100 + 10 * 0 + 25 * 80 = 4570$
- **IBM X3700:**  $28 * 50 + 20 * 100 + 12 * 5 + 5 * 0 + 10 * 100 + 25 * 90 = 6710$

**Informe de decisión:** Se toma como referencia al servidor **IBM X3700** ya que su capacidad de memoria RAM soporta expansiones, tiene una amplia disponibilidad de cómputos de alta velocidad en distintos cores, y su almacenamiento es de tipo espejado.

#### 4.2.3. Modems y Redes

-	Peso	M2M Mod-Bus	M2M PRO4	RS484
<b>Soporte</b>	10 %	100	100	100
<b>Clúster</b>	15 %	100	100	20
<b>Comunicación:</b>	20 %	20	30	100
<b>Escalabilidad:</b>	12 %	10	10	80
<b>Velocidad:</b>	15 %	25	25	100
<b>Costo:</b>	28 %	80	100	20

La ponderación se realiza de la siguiente forma:

- **M2M ModBus:**  $10 * 100 + 15 * 100 + 20 * 20 + 12 * 10 + 15 * 25 + 28 * 80 = 5635$

- **M2M Pro4:**  $10 * 100 + 15 * 100 + 20 * 30 + 12 * 10 + 15 * 25 + 28 * 80 = 6395$

- **RS484:**  $10 * 100 + 15 * 20 + 20 * 100 + 12 * 80 + 15 * 100 + 28 * 20 = 6320$

**Informe de decisión:** Se toma como referencia al Modem Industrial **M2M Pro4** ya que posee mayor puertos de comunicación a diferencia del ModBus, y si bien su velocidad no es tan rápida como el del RS484, es más estable tanto en soporte como en sistemas de Clúster para una red amplia. Según sus especificaciones técnicas, su programación es mucho más abordable que los sistemas de Cisco.

#### 4.3. Factibilidad Operativa

A continuación se realiza un estudio de Factibilidad Operativa para distintas áreas, a saber:

- **Equipo de trabajo:** Evaluación de los puestos necesarios para llevar a cabo el desarrollo de software y hardware del proyecto.
- **Capacitación:** Evaluación de los conocimientos requeridos *a posteriori* necesarios para afrontar la primer puesta en marcha del proyecto.

##### 4.3.1. Equipo de Trabajo

-	Peso	Analista	Junior	Ssr	Senior
<b>Capacitación</b>	10 %	30	100	40	10
<b>Autonomía</b>	15 %	40	10	50	100
<b>Resolución:</b>	20 %	30	5	60	100
<b>Conocimientos:</b>	15 %	30	10	65	90
<b>Entregas:</b>	10 %	40	20	70	100
<b>Sueldo:</b>	30 %	20	80	35	10

La ponderación se realiza de la siguiente forma:

- **Analista:**  $10 * 30 + 15 * 40 + 20 * 30 + 15 * 30 + 10 * 40 + 30 * 20 = 2950$
- **Junior:**  $10 * 100 + 15 * 10 + 20 * 5 + 15 * 10 + 10 * 20 + 30 * 80 = 4000$
- **Ssr:**  $10 * 40 + 15 * 50 + 20 * 60 + 15 * 65 + 10 * 70 + 30 * 35 = 5075$
- **Senior:**  $10 * 10 + 15 * 100 + 20 * 100 + 15 * 90 + 10 * 100 + 30 * 10 = 6250$

**Informe de decisión:** Se elige el perfil **Senior** ya que posee los conocimientos necesarios para el desarrollo que se estaría llevando a cabo en el proyecto; por su perfil se entiende que tiene autonomía en su trabajo y en una primera instancia no estaría necesitando capacitación alguna. Si bien el elevado sueldo implica cierta incertidumbre, se entiende que gracias a su perfil y experiencia, se podrá abordar el proyecto de manera meritoria. También como se hace referencia en 5.2, en una primera instancia del proyecto, se requeriría que los conocimientos de los técnicos Seniors en Redes, sepan de configuración y puesta en marcha de Servidores.

#### 4.3.2. Capacitaciones

-	Peso	eDx	Coursera	Udemy
<b>Aprendizaje</b>	30 %	60	30	70
<b>Interacción</b>	30 %	30	30	45
<b>Plataforma:</b>	10 %	100	100	100
<b>Instructores:</b>	10 %	80	100	100
<b>Espectro de cursos:</b>	7 %	100	100	100
<b>Costo:</b>	13 %	50	40	60

La ponderación se realiza de la siguiente forma:

- **eDx:**  $30 * 60 + 30 * 30 + 10 * 100 + 10 * 80 + 7 * 100 + 13 * 50 = 5850$
- **Coursera:**  $30 * 30 + 30 * 30 + 10 * 100 + 10 * 100 + 7 * 100 + 13 * 40 = 5020$
- **Udemy:**  $30 * 70 + 30 * 45 + 10 * 100 + 10 * 90 + 7 * 100 + 13 * 60 = 6830$

**Informe de decisión:** A raíz de la ponderación evaluada, se toma como referencia la capacitación bajo **Udemy** como la plataforma por excelencia para realizar las capacitaciones al personal técnico. Al ser la plataforma líder en el mercado de cursos, tiene un espectro bastante amplio en lo que respecta a cursos sobre distintas tecnologías, permitiendo a cada profesional crear sus propios cursos. Si bien el costo es más elevado que en las otras plataformas, se tiene la posibilidad de obtener descuentos y bonificaciones si se toman varios cursos sobre una misma tecnología.

## 5. Informe sobre Factibilidad escogida

### 5.1. Costos totales sobre la Ponderación

Según lo ponderado en la sección 4 se obtienen los costes según una evaluación realizada con fecha **24 de noviembre de 2020**, la cual, según estimaciones, podrá estar vigente sólo con un alcance de 2 meses como máximo, si se tiene en cuenta la inflación de la región. En la siguiente tabla, se visualizan los costes diversificados por sector:

Factibilidad	Cantidad	Descripción	Precio U.	Total
Técnica (Web)	1	Angular	\$ 45.000	\$ 45.000
Técnica (Servidores)	3	IBM X3700	\$ 450.780	\$ 1.352.340
Técnica (Red)	4	M2M Pro4	\$ 250.300	\$ 1.001.200
Operativa (Equipo)	4	Senior	\$ 150.000	\$ 600.000
Operativa (Capacitaciones)	1	Udemy	\$ 10.000	\$ 10.000
<b>Costo Total</b>	-	-	-	<b>\$ 3.008.540</b>

Si tenemos en cuenta los costes presupuestados para las distintas áreas que se visualizan en la sección 4.1, se puede tener presente que hay un **sobranante** presupuestario de **\$ 151.460,00**, el cual en parte puede destinarse a cualquier imprevisto ocasionado en el primer período de consolidación del proyecto.

### 5.2. Informe Final

En cuanto a la **Factibilidad Técnica** desarrollada en la Sección 4.2 se comenzó investigando sobre el software para el desarrollo de la página web referido a React por sus prestaciones en el diseño de interfaces para el usuario, el cual permite exposiciones en una misma página. Aún así, el desarrollo de React no es tan acelerado como el de su competidor **Angular**, el cual mantiene una plataforma estable desde hace varios años y una comunidad muy grande en Latinoamérica. Angular por otro lado, consta de un framework de aplicaciones web de código abierto y mantenido por el gigante Google el cual también permite crear y mantener aplicaciones en una misma página, pero con el agregado que permite tener mucho más en cuenta la capacidad del modelo Vista Controlador, haciendo que el testeado de las pruebas sean mucho más fáciles. Las licencias son anuales lo que permite en una primera etapa del proyecto costearlas sin ningún problema; siendo la instalación para unas 20 máquinas aproximadamente o 10 y con un beneficio del módulo de Testing en el Servidor.

A lo que respecta con la decisión sobre el área de Servidores, al tener que montar los servicios para el cliente bajo la herramienta Angular se preponderó la capacidad de CPU. El servidor **IBM X3700** tiene la posibilidad de trabajar de forma paralelizable entre sus 7 cores con tecnología MOSPH. Asimismo, la herramienta Angular permite equilibrar la carga entre en Cliente y el Servidor, posibilitando que en caso que el Cliente que quiera interactuar con la página, y no disponga de mucho poder de cómputo, se pasa la carga al Servidor IBM.

Debido a que las áreas a comunicar en una primera instancia del Proyecto no son para nada despreciables, se tomó en cuenta el hardware **M2M Pro4** el cual permite a una gran distancia y en condiciones ambientales contraproducentes (Comunicación), mantener una conexión altamente veloz. Por otro lado, las peculiaridades del M2M permiten que tenga internamente un switch, el cual se activa de forma electrónica y como también automática para equilibrar la carga de paquetes que pueda estar recibiendo el Servidor. Asimismo soporta firewalls implementados completamente a nivel de hardware, por lo que lo convierte en un módem más robusto y confiable ante ataques externos; a diferencia de los otros dispositivos evaluados, los cuales no contaban con esta peculiaridad de tener integrados un firewall electrónico.

En cuanto a la **Factibilidad Operativa** desarrollada en la Sección 4.3, se comenzó evaluando los costos que existen en el mercado laboral para la región mediante algunas Bolsas de Trabajo como pueden ser: LinkedIn, Universo Bit, CompuTrabajo o Bumerang. Se evaluaron que los conocimientos técnicos en lo que respecta tanto para el sector de Servidores, el operativo de Redes y el de Desarrollo Web, llegando a la conclusión que *en esta primera instancia del proyecto* será necesario contar con los conocimientos más avanzados en cada uno de los aspectos mencionados, teniendo mayor significación momentáneamente el área de Servidores y Red que el de Desarrollo Web. Es por ello que se llegó a la conclusión de tener 2 técnicos **Senior** para la parte de Servidores, uno para la parte de Redes y otro para el Desarrollo Web. Quien esté a cargo del área de Redes, también será necesario que cuente con conocimientos de configuración de Servidores, en una primera instancia.

Angular influyó mucho en la decisión acerca de las capacitaciones brindadas por la plataforma **Udemy**, la cual, tiene muchos de sus cursos dictados por profesionales dentro del área de Google, con subtítulos y con la posibilidad de tener cursos más avanzados sin costos adicionales. La plataforma también permite contactarse tanto con el personal que imparte el curso como también los alumnos, haciendo que el aprendizaje sea aún mucho más fluido que en otras plataformas.

## 6. Metodología y Alcance del Proyecto

### 6.1. Metodología

La **metodología adoptada** por el equipo de trabajo es **Scrum**, el cual permite el trabajo colaborativo, obteniendo los mejores resultados posibles a medida que el proyecto avanza.

Scrum es indicado para proyectos que constan de entornos complejos, donde se necesita obtener resultados de forma rápida, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

El proceso parte de la lista de objetivos/requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente (Product Owner) prioriza los objetivos balanceando el valor que le aportan respecto a su coste (que el equipo estima considerando la Definición de Hecho) y quedan repartidos en iteraciones y entregas.

## 6.2. Alcance del Proyecto

El **Alcance del Proyecto** consta de tres sprints los cuales se listan a continuación:

1. **Primer Sprint:** Abarca del Lunes 14/09/2020 al Lunes 5/10/2020.  
**Propósito:** Desarrollo del ABM de Pacientes junto con la Reserva de Turnos.
2. **Segundo Sprint:** Abarca del Lunes 05/10/2020 al Lunes 02/11/2020.  
**Propósito:** Desarrollo de las Historias Clínicas.
3. **Tercer Sprint:** Abarca del Lunes 02/11/2020 al Lunes 30/11/2020.  
**Propósito:** Desarrollo de los Registros de Internación, junto con el ABM de Camas.

El propósito completo de la implementación es que al cabo de las tres iteraciones, queden implementados los tres desarrollos que conlleva cada uno de los sprints. Cabe destacar que lo no se tiene en cuenta en estas etapas el testeó regresivo de cada uno de los módulos.

## 7. Requerimientos Funcionales y No Funcionales

### 7.1. Requerimientos Funcionales

La lista de Requerimientos Funcionales a partir del caso citado en el apartado 2.1 se pueden dividir según a las entidades que lo conciernen. A continuación se hace un listado de Requerimientos Funcionales (RF).

### 7.2. En base al Paciente

Los Requerimientos Funcionales que se obtienen a partir de la entidad **Paciente** son los siguientes:

<b>ID del Requerimiento:</b>	RF101
<b>Nombre:</b>	Registrar Paciente
<b>Función del servicio:</b>	El Administrador podrá dar de alta en el sistema un Paciente ingresando el Apellido, Nombre, DNI, dirección y teléfono de la persona.
<b>Características:</b>	Todos los datos son obligatorios para dar de alta el Paciente. Quien puede efectuar el alta es el Administrador
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF102
<b>Nombre:</b>	Consultar Paciente
<b>Función del servicio:</b>	Se podrá consultar a la base de datos el Paciente una vez que haya sido dado de alta.
<b>Características:</b>	La lista de Pacientes podrá ser consultada a la base de datos por el Administrador ingresando el DNI o el Apellido de la persona. Los demás campos no serán de ingreso para posibilitar la búsqueda. Dentro de esta Opción se tendrá acceso al menú de Consultar Turno (RF202), Consultar Historia Clínica (RF302), Consultar Internación (RF402)
<b>Requerimientos Relacionados:</b>	RF202, RF302, RF402
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF103
<b>Nombre:</b>	Modificar Paciente
<b>Función del servicio:</b>	Permite corregir los datos de un Paciente determinado.
<b>Características:</b>	El Administrador podrá corregir o volver a ingresar los datos de un Paciente que ya se encuentre previamente en la base de datos. La opción será desplegada mediante el menú de Consultar Paciente (RF102), como también Anular Paciente (RF104)
<b>Requerimientos Relacionados:</b>	RF102, RF104
<b>Prioridad:</b>	Media
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF104
<b>Nombre:</b>	Anular Paciente
<b>Función del servicio:</b>	El Administrador podrá eliminar un Paciente previamente establecido en la base de datos.
<b>Características:</b>	Mediante el menú Consultar Paciente (RF102) será posible desplegar una opción que permita eliminar el Paciente solicitado. La búsqueda del Paciente se hará por DNI o por Apellido exclusivamente.
<b>Requerimientos Relacionados:</b>	RF102
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF105
<b>Nombre:</b>	Imprimir Paciente
<b>Función del servicio:</b>	Permite al Administrador imprimir un Paciente que se encuentre previamente en la base de datos.
<b>Características:</b>	El Administrador mediante el menú Consultar Paciente (RF102), podrá imprimir toda la información previamente cargada en la base de datos referida al Paciente en cuestión.
<b>Requerimientos Relacionados:</b>	RF102
<b>Prioridad:</b>	Baja
<b>Período de Desarrollo:</b>	Primer Sprint

### 7.3. En base a la Reserva de Turnos

Los Requerimientos Funcionales que se obtienen a partir de la entidad **Reserva de Turnos** son los siguientes:

<b>ID del Requerimiento:</b>	RF201
<b>Nombre:</b>	Registrar Turno
<b>Función del servicio:</b>	El Administrador podrá dar de alta en el sistema un Turno ingresando el Apellido, Nombre, DNI, dirección y teléfono de la persona.
<b>Características:</b>	Todos los datos son obligatorios para dar de alta el Turno. Quien puede efectuar el alta es el Administrador
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF202
<b>Nombre:</b>	Consultar Turno
<b>Función del servicio:</b>	Se podrá consultar a la base de datos el Turno una vez que haya sido dado de alta.
<b>Características:</b>	La lista de Turnos podrá ser consultada a la base de datos por el Administrador ingresando el DNI o el Apellido de la persona. Los demás campos no serán de ingreso para posibilitar la búsqueda. Dentro de esta Opción se tendrá acceso al menú de Consultar Paciente (RF202), Consultar Historia Clínica (RF302), Consultar Internación (RF402).
<b>Requerimientos Relacionados:</b>	RF202, RF302, RF402.
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint



<b>Número identificador:</b>	RF203
<b>Nombre:</b>	Modificar Turno
<b>Función del servicio:</b>	Permite corregir los datos de un Turno determinado.
<b>Características:</b>	El Administrador podrá corregir o volver a ingresar los datos de un Turno que ya se encuentre previamente en la base de datos. La opción será desplegada mediante el menú de Consultar Turno (RF202)
<b>Requerimientos Relacionados:</b>	RF202
<b>Prioridad:</b>	Media
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF204
<b>Nombre:</b>	Anular Turno
<b>Función del servicio:</b>	El Administrador podrá eliminar un Turno previamente establecido en la base de datos.
<b>Características:</b>	Mediante el menú Consultar Turno (RF202) será posible desplegar una opción que permita eliminar el Turno solicitado. La búsqueda del Turno se hará por DNI o por Apellido exclusivamente.
<b>Requerimientos Relacionados:</b>	RF202
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Primer Sprint

<b>Número identificador:</b>	RF205
<b>Nombre:</b>	Imprimir Turno
<b>Función del servicio:</b>	Permite al Administrador imprimir un Turno que se encuentre previamente en la base de datos.
<b>Características:</b>	El Administrador mediante el menú Consultar Turno (RF202), podrá imprimir toda la información previamente cargada en la base de datos referida al Turno en cuestión.
<b>Requerimientos Relacionados:</b>	RF202
<b>Prioridad:</b>	Baja
<b>Período de Desarrollo:</b>	Primer Sprint

#### 7.4. En base a las Historias Clínicas

Los Requerimientos Funcionales que se obtienen a partir de la entidad **Historias Clínicas** son los siguientes:

<b>ID del Requerimiento:</b>	RF301
<b>Nombre:</b>	Registrar Historia Clínica
<b>Función del servicio:</b>	El Administrador podrá dar de alta en el sistema una Historia Clínica ingresando el Apellido, Nombre, DNI, dirección y teléfono de la persona.
<b>Características:</b>	Todos los datos son obligatorios para dar de alta el Historia Clínica. Quien puede efectuar el alta es el Administrador
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Segundo Sprint

<b>Número identificador:</b>	RF302
<b>Nombre:</b>	Consultar Historia Clínica
<b>Función del servicio:</b>	Se podrá consultar a la base de datos la Historia Clínica una vez que haya sido dado de alta.
<b>Características:</b>	La lista de Historias Clínicas podrá ser consultada a la base de datos por el Administrador ingresando el DNI o el Apellido de la persona. Los demás campos no serán de ingreso para posibilitar la búsqueda. Esto puede ser accedido también desde Consultar Paciente (RF102)
<b>Requerimientos Relacionados:</b>	RF102, RF202
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Segundo Sprint

<b>Número identificador:</b>	RF303
<b>Nombre:</b>	Modificar Historia Clínica
<b>Función del servicio:</b>	Permite corregir los datos de una Historia Clínica determinada.
<b>Características:</b>	El Administrador podrá corregir o volver a ingresar los datos de una Historia Clínica que ya se encuentre previamente en la base de datos. La opción será desplegada mediante el menú de Consultar Historia Clínica (RF302), como también de Consultar Paciente (RF102) y Consultar Turno (RF202)
<b>Requerimientos Relacionados:</b>	RF302, RF102, RF202
<b>Prioridad:</b>	Media
<b>Período de Desarrollo:</b>	Segundo Sprint

<b>Número identificador:</b>	RF304
<b>Nombre:</b>	Anular Historia Clínica
<b>Función del servicio:</b>	El Administrador podrá eliminar un Historia Clínica previamente establecido en la base de datos.
<b>Características:</b>	Mediante el menú Consultar Historia Clínica (RF02) será posible desplegar una opción que permita eliminar la Historia Clínica solicitada. La búsqueda de la Historia Clínica se hará por DNI o por Apellido exclusivamente.
<b>Requerimientos Relacionados:</b>	RF302
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Segundo Sprint

<b>Número identificador:</b>	RF305
<b>Nombre:</b>	Imprimir Historia Clínica
<b>Función del servicio:</b>	Permite al Administrador imprimir una Historia Clínica que se encuentre previamente en la base de datos.
<b>Características:</b>	El Administrador mediante el menú Consultar Historia Clínica, podrá imprimir toda la información previamente cargada en la base de datos referida a la Historia Clínica en cuestión.
<b>Requerimientos Relacionados:</b>	RF302
<b>Prioridad:</b>	Baja
<b>Período de Desarrollo:</b>	Segundo Sprint

## 7.5. En base a la Internación de Pacientes

Los Requerimientos Funcionales que se obtienen a partir de la entidad **Internación de Pacientes** son los siguientes:

<b>ID del Requerimiento:</b>	RF401
<b>Nombre:</b>	Registrar Internación
<b>Función del servicio:</b>	El Administrador podrá dar de alta en el sistema una Internación de un Paciente ingresando el Apellido, Nombre, DNI, dirección y teléfono de la persona.
<b>Características:</b>	Todos los datos son obligatorios para dar de alta la Internación. Quien puede efectuar el alta es el Administrador
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF402
<b>Nombre:</b>	Consultar Internación
<b>Función del servicio:</b>	Se podrá consultar a la base de datos la Internación una vez que haya sido dado de alta.
<b>Características:</b>	La lista de Internacións podrá ser consultada a la base de datos por el Administrador ingresando el DNI o el Apellido de la persona. Los demás campos no serán de ingreso para posibilitar la búsqueda. Dentro de esta Opción se tendrá acceso al menú de Consultar Internación (RF402), como también Consultar Paciente (RF102), Consultar Turno (RF202) y Consultar Historia Clínica (RF302)
<b>Requerimientos Relacionados:</b>	RF402, RF102, RF202, RF302
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF403
<b>Nombre:</b>	Modificar Internación
<b>Función del servicio:</b>	Permite corregir los datos de una Internación determinada.
<b>Características:</b>	El Administrador podrá corregir o volver a ingresar los datos de una Internación que ya se encuentre previamente en la base de datos. La opción será desplegada mediante el menú de Consultar Internación (RF402)
<b>Requerimientos Relacionados:</b>	RF402
<b>Prioridad:</b>	Media
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF404
<b>Nombre:</b>	Anular Internación
<b>Función del servicio:</b>	El Administrador podrá eliminar una Internación previamente establecido en la base de datos.
<b>Características:</b>	Mediante el menú Consultar Internación (RF402) será posible desplegar una opción que permita eliminar la Internación solicitada. La búsqueda del Internación se hará por DNI o por Apellido exclusivamente.
<b>Requerimientos Relacionados:</b>	RF402
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF405
<b>Nombre:</b>	Imprimir Internación
<b>Función del servicio:</b>	Permite al Administrador imprimir los datos referida a una Internación que se encuentre previamente en la base de datos.
<b>Características:</b>	El Administrador mediante el menú Consultar Internación (RF402), podrá imprimir toda la información previamente cargada en la base de datos referida a la Internación en cuestión.
<b>Requerimientos Relacionados:</b>	RF402
<b>Prioridad:</b>	Baja
<b>Período de Desarrollo:</b>	Tercer Sprint

## 7.6. En base a las camas disponibles

Los Requerimientos Funcionales que se obtienen a partir de la entidad **Internación de Pacientes** consultando a través de la entidad **Camas** para saber si es que existe la posibilidad de internación.

<b>Número identificador:</b>	RF501
<b>Nombre:</b>	Consultar Cama
<b>Función del servicio:</b>	A través del menú Registrar Internación (RF401) se podrá visualizar las camas disponibles para tal fin.
<b>Características:</b>	La lista de Internacións podrá ser consultada a la base de datos por el Administrador ingresando el DNI o el Apellido de la persona. Los demás campos no serán de ingreso para posibilitar la búsqueda. Para ocupar la cama, previamente tiene que haber sido liberada por el menú Liberar Cama (RF503). Dentro de esta Opción se tendrá acceso al menú de Consultar Internación (RF402), como también Consultar Paciente (RF102), Consultar Turno (RF202) y Consultar Historia Clínica (RF302)
<b>Requerimientos Relacionados:</b>	RF401, RF402, RF102, RF202, RF302
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF502
<b>Nombre:</b>	Ocupar Cama
<b>Función del servicio:</b>	Mediante la opción Consultar Cama, la cual se accede desde Registrar Internación (RF402) será posible ocupar una cama siempre que el estado de la cama sea <i>Libre</i> . De lo contrario se produce un error.
<b>Características:</b>	La lista de Camas disponibles podrá ser accedida desde el menú de Consultar Cama (RF501) que a su vez puede ser accedido de Registrar Internación (RF401)
<b>Requerimientos Relacionados:</b>	RF402, RF501, RF401
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

<b>Número identificador:</b>	RF502
<b>Nombre:</b>	Liberar Cama
<b>Función del servicio:</b>	Una vez que haya sido dado de alta el paciente que registro una Internación (RF401), será posible pasar el estado de la cama a <i>Libre</i> nuevamente.
<b>Características:</b>	La opción será accedida desde Consultar Cama (RF501), que se accede de forma indirecta por Registrar Internación (RF401)
<b>Requerimientos Relacionados:</b>	RF401, RF403, RF404, RF501.
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Tercer Sprint

## 7.7. Requerimientos No Funcionales

La lista de Requerimientos Funcionales a partir del caso citado en el apartado 2.1 se pueden dividir según a las entidades que lo conciernen. A continuación se hace un listado de estos Requerimientos No Funcionales (RNF).

<b>Número identificador:</b>	RNF01
<b>Nombre:</b>	Descripción del Idioma
<b>Función del servicio:</b>	El idioma del software debe de estar en Español/Castellano.
<b>Características:</b>	Dada la localización de las entidades de salud, y de los posibles usuarios que vayan a interactuar con el software, es primordial que el idioma sea Español/Castellano.
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Antes de la implementación del Primer Sprint

<b>Número identificador:</b>	RNF02
<b>Nombre:</b>	Sistema Operativo
<b>Función del servicio:</b>	Quien opere los módulos médicos y administrativos de las entidades gubernamentales, deberá de correrlo bajo Microsoft Windows 10 o en su defecto bajo Ubuntu LTS 20.04.1
<b>Características:</b>	El Sistema Operativo de base será Microsoft Windows 10 o en su defecto Ubuntu LTS 20.04.1 para de esta manera tener las actualizaciones de seguridad en un período de 4 años consecutivos.
<b>Requerimientos Relacionados:</b>	-
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Antes de la implementación del Segundo Sprint

<b>Número identificador:</b>	RNF03
<b>Nombre:</b>	Gestor de Base de Datos
<b>Función del servicio:</b>	Se establece que el Gestor de Base de Datos utilizado en todos los módulos sea MySQL.
<b>Características:</b>	Se establece que el SGBD sea MySQL ya que tiene un soporte mucho más amplio y un tiempo de respuesta de dudas y errores mucho más alto que otros gestores por ser de código abierto como también de mayor uso en los últimos 10 años.
<b>Requerimientos Relacionados:</b>	RNF02
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Antes de la implementación del Segundo Sprint

<b>Número identificador:</b>	RNF04
<b>Nombre:</b>	Backend de Interfaz de Usuario
<b>Función del servicio:</b>	Se establece que el Backend sea gestionado mediante las herramientas de desarrollo de código abierto Node.JS
<b>Características:</b>	Dado que la herramienta está pensada para ser utilizada desde un navegador web, el cual permite mayor agilidad a la hora de montar distintos puntos de administración como también una menor necesidad de recursos de la computadora implementada.
<b>Requerimientos Relacionados:</b>	RNF03
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Antes de la implementación del Segundo Sprint

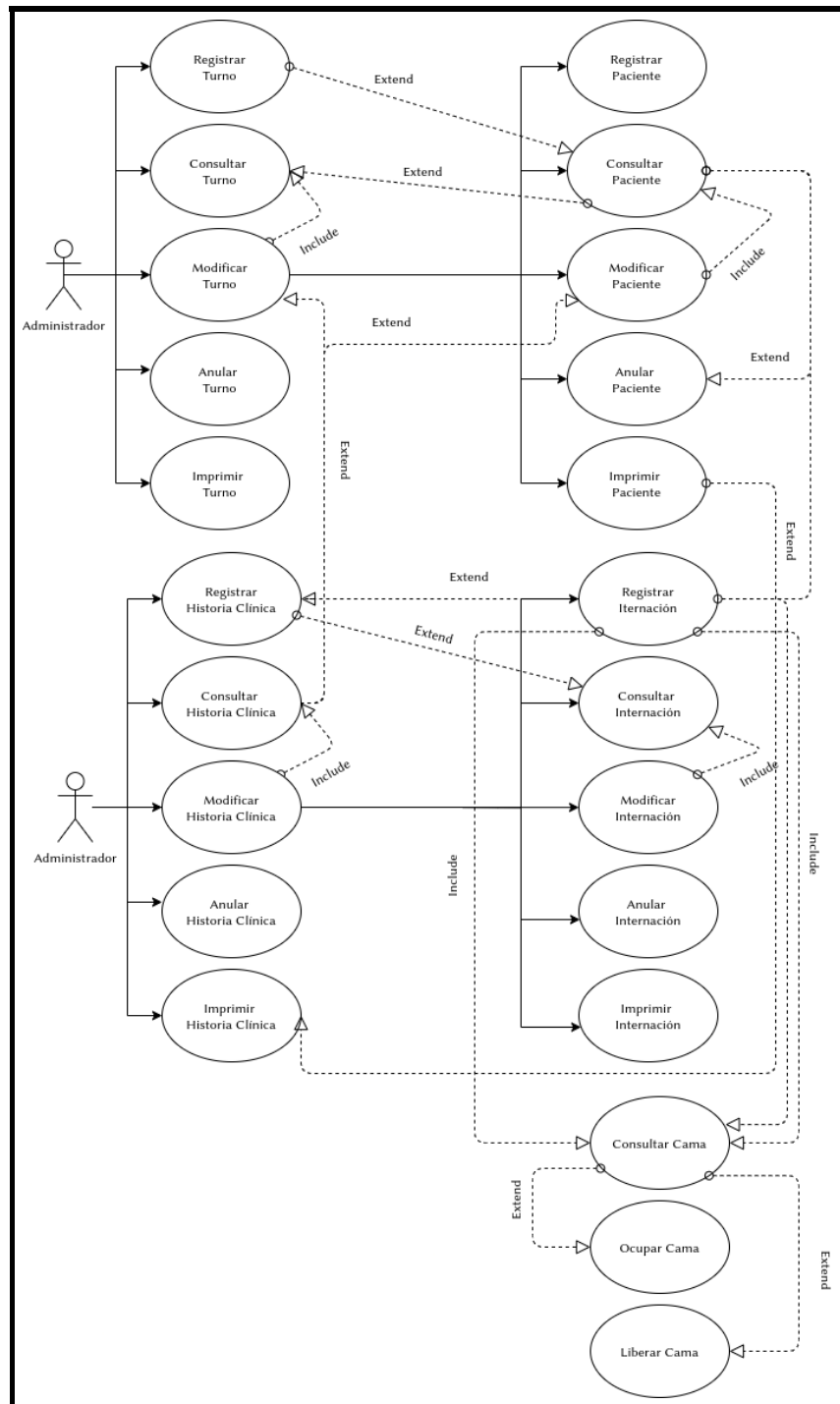
<b>Número identificador:</b>	RNF05
<b>Nombre:</b>	Frontend de Interfaz de Usuario
<b>Función del servicio:</b>	Se establece que el Frontend sea desarrollado mediante Angular
<b>Características:</b>	Se determina que el Frontend sea desarrollado mediante el entorno de desarrollo proporcionado por el Angular por varias razones: es de código abierto, permite programación orientada a objetos, es multiplataforma y permite tener estrecha relación entre el cliente y el servidor
<b>Requerimientos Relacionados:</b>	RNF03, RNF04
<b>Prioridad:</b>	Alta
<b>Período de Desarrollo:</b>	Antes de la implementación del Segundo Sprint

## 8. Casos de Uso

A continuación se pueden visualizar los casos de uso que se tienen que llevar a cabo al finalizar los tres sprints programados para los distintos módulos.

Cabe resaltar que el primer sprint será el que más carga horaria de trabajo tendrá ya que se deben implementar todo lo concerniente al **Paciente** como también al **Turno** que lleva a cabo la Recepcionista.





## 9. Métricas

### 9.1. Primer Sprint y el Backlog Item

A continuación se presentan las métricas del primer Sprint realizado para obtener así los primeros hitos, a saber:

- Desarrollo del ABM de Pacientes
- Desarrollo de la Reserva de Turnos

Cabe destacar que en este primer sprint, se tiene la mayor cantidad de horas hombre trabajadas por día ya que a diferencia de los otros dos sprints, en este se deben implementar dos módulos. Es por ello que se pensaron 3 semanas de **40hs cada una**, es decir, 8hs de trabajo diario.

Se pueden ver los Requerimientos que se encuentran en el **Backlog** del equipo:

Backlog Item	Tiempo Estimado en hs
Registrar Paciente	30
Consultar Paciente	11
Modificar Paciente	9
Anular Paciente	8
Imprimir Paciente	11
Registrar Turno	11
Consultar Turno	15
Modificar Turno	10
Anular Turno	9
Imprimir Turno	6
<b>Total</b>	<b>120</b>

### 9.2. Consumo de horas por semana

A continuación se visualizan las horas consumidas en cada una de las 3 semanas que dura el sprint:

### 9.2.1. Primera semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Registrar Paciente	9	8	4	6	3	30
Consultar Paciente	1	2	3	4	1	11
Modificar Paciente		2	3	3	1	9
Total de Hs por semana						50

### 9.2.2. Segunda semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Anular Paciente	2	3	3			8
Imprimir Paciente	3	1	3	4		11
Registrar Turno		2	3	3	3	11
Consultar Turno	4	3	3	2	3	15
Total de Hs por semana						45

### 9.2.3. Tercer semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Modificar Turno	1	3	6			10
Anular Turno	1	2	3	2	1	9
Imprimir Turno			3	2	1	6
Total de Hs por semana						25

## 9.3. Segundo Sprint y su Backlog Item

A continuación se presentan las métricas del Segundo Sprint realizado para obtener así el siguiente hito:

- Desarrollo del ABM de las Historias Clínicas
- Testeo Regresivo para el ABM de Pacientes
- Testeo Regresivo para el ABM de Turnos

Cabe mencionar que a diferencia del Primer Sprint, este tiene una menor cantidad de horas hombre trabajadas en el nuevo módulo **Historias Clínicas** dado que sólo se deben implementar este, a diferencia del anterior donde se debían de implementar dos. Es por ello que se intercala con un testeo regresivo para el módulo de **Pacientes** y el de **Turnos**, ya que estos últimos se relacionan con el nuevo.

Se pueden ver los Requerimientos que se encuentran en el **Backlog** del equipo:

#### 9.4. Consumo de horas por semana

Backlog Item	Tiempo Estimado en Hs
Registrar Historia Clínica	32
Consultar Historia Clínica	12
Modificar Historia Clínica	10
Anular Historia Clínica	7
Imprimir Historia Clínica	12
Testeo Regresivo Pacientes	30
Testeo Regresivo Turnos	17
<b>Total</b>	<b>120</b>

A continuación se visualizan las horas consumidas en cada una de las 3 semanas que dura el sprint:

##### 9.4.1. Primera semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Registrar Historia Clínica	7	6	6	7	6	32
Consultar Historia Clínica	5	3	2	2		12
Testeo Regresivo Pacientes				2	5	7
	Total de Hs por semana					51

##### 9.4.2. Segunda semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Registrar Historia Clínica	7	6	6	7	6	32
Consultar Historia Clínica	5	3	2	2		12
Testeo Regresivo Pacientes				2	5	7
	Total de Hs por semana					51

##### 9.4.3. Tercer semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Registrar Historia Clínica	7	6	6	7	6	32
Consultar Historia Clínica	5	3	2	2		12
Testeo Regresivo Pacientes				2	5	7
	Total de Hs por semana					51

## 9.5. Tercer Sprint y su Backlog Item

A continuación se presentan las métricas del Tercer Sprint realizado para obtener así el siguiente hito:

- Desarrollo de los Registros de Internación
- Desarrollo del ABM de Camas.
- Desarrollo e implementación de la clase Médico.
- Testeo Regresivo para las Historias Clínicas.

Como se dijo en la sección dedicada al Diagrama de Clase, se suma al tercer Sprint, la clase **Médico**, destinada a organizar de mejor manera las relaciones existentes entre la Internación, que luego impactarán en las Historias clínicas.

Se sigue con una merma en las horas hombre dedicadas a esta última etapa del desarrollo, es por eso que se decidió incluir en este Sprint la implementación de la clase Médico, para poder de esta forma, poder balancear la carga horaria de los desarrolladores. Aún así, se necesitará realizar horas extras en el equipo ya que sobrepasa en 10 horas, a las horas semanales acordadas en los 2 Sprints anteriores. Estas horas de más, estarán abocadas en la primer semana.

Para ello se intercala con un testeo regresivo para el módulo de **Registro de Internación** y el **ABM de las Camas**, junto con la clase **Médico** y un testeo regresivo para las **Historias Clínicas**, ya que estos últimos se relacionan con el módulo nuevo.

Se pueden ver los Requerimientos que se encuentran en el **Backlog** del equipo:

## 9.6. Consumo de horas por semana

Backlog Item	Tiempo estimado en Hs
Registrar Internación	32
Consultar Internación	13
Modificar Internación	9
Anular Internación	8
Imprimir Internación	11
Consultar Cama	20
Ocupar Cama	7
Liberar Cama	9
Implementación de Médico	13
Testeo de Historias Clínicas	8
<b>Total</b>	<b>130</b>

A continuación se visualizan las horas consumidas en cada una de las 3 semanas que dura el sprint:

### 9.6.1. Primera semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Registrar Internación	8	7	9	5	3	32
Consultar Internación	1	3	2	6	1	13
Modificar Internación			2	4	3	9
Total de Hs por semana						54

### 9.6.2. Segunda semana

Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Anular Internación	3	1	4			8
Imprimir Internación	3	2	4	2	1	12
Consultar Cama	4	3	5	6	2	20
Total de Hs por semana						40

### 9.6.3. Tercer semana

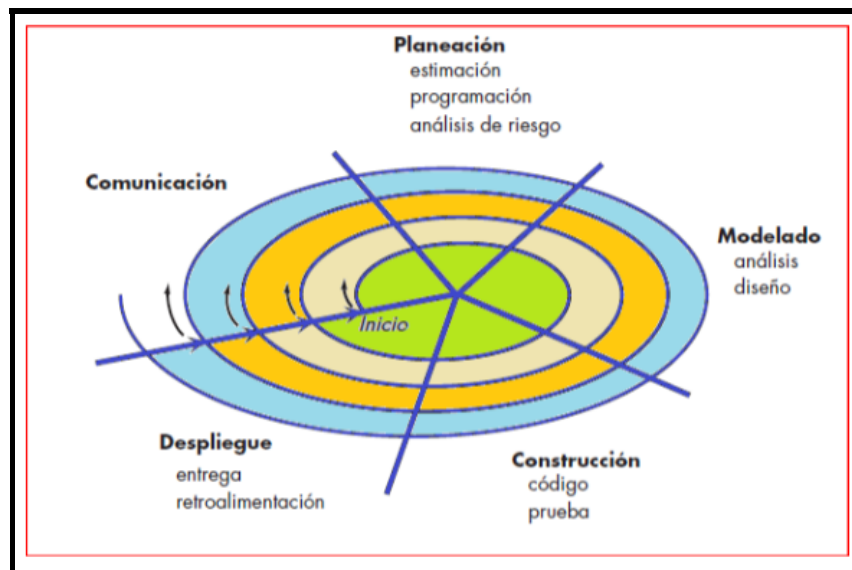
Requerimientos	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Liberar Cama	1	3	2	1		7
Liberar Cama	1	3	2	1	2	9
Implementación de Médico	4	2	6	1		13
Testeo de Historias Clínicas			2	4	2	8
Total de Hs por semana						30

## 10. Selección de Ciclo de Vida

En este proyecto se podría utilizar un modelo en espiral, priorizando la idea de poder solucionar problemáticas que vayan surgiendo con el transcurso de su desarrollo, pudiendo volver a etapas anteriores y combinándolo con la idea del modelo en cascada, en caso que todo se desarrolle de forma normal.

La elección de este modelo se justifica por la metodología de desarrollo que se propone realizar desde el comienzo del proyecto, implementando de forma evolutiva los requerimientos al cliente, siendo los mismos impulsados por el riesgo.

A medida que se avance con el proyecto, se logran hitos de entrega con versiones de los módulos descritos anteriormente, cada vez más completas hasta alcanzar la totalidad de los requerimientos deseados.



En cada circuito alrededor del espiral se plantean objetivos a cumplir para los módulos a desarrollar, tomando como iniciativa principal, la reducción de riesgos. En la etapa de planeación se determinan los ajustes al proyecto. Al finalizar el proyecto, el ciclo de vida continuará estando vigente para poder adaptarse a las mejoras que se planteen para el sistema en su completitud.

## 11. Gestión y Plan de Riesgo

A continuación se hace un Plan de Riesgos sobre el proyecto encausado, tomando en cuenta los siguientes puntos:

- Valorización de la Disponibilidad
- Valorización de la Integridad
- Valorización de la Cofidencialidad

### 11.1. Valorización de la Disponibilidad

Valor asignado	Criterio de evaluación
0	No aplica o no es relevante
1	Debe estar disponible al menos el 10 % del tiempo
2	Debe estar disponible al menos el 50 % del tiempo
3	Debe estar disponible al menos el 99 % del tiempo

## 11.2. Valorización de la Integridad

Valor asignado	Criterio de evaluación
0	No aplica o no es relevante
1	Debe estar disponible al menos el 10 % del tiempo
2	Debe estar disponible al menos el 50 % del tiempo
3	Debe estar disponible al menos el 99 % del tiempo

## 11.3. Valorización de la Confidencialidad

Valor asignado	Criterio de evaluación
0	No aplica o no es relevante
1	Debe estar disponible al menos el 10 % del tiempo
2	Debe estar disponible al menos el 50 % del tiempo
3	Debe estar disponible al menos el 99 % del tiempo

## 11.4. Inventario de Activos

A continuación se realiza el inventario de activos que posee el proyecto general.

ID	Activo	Descripción	Propietario
1	Servicio Eléctrico	Electricidad	Proveedor
2	Servicio de Internet	Internet	Proveedor
3	Personal Profesional	RRHH	Gerencia de RRHH
4	Sistema de Aplicación	Software de desarrollo e implementación	Dpto. de Informática
5	Instalaciones	PCs, notebooks, impresoras, sillas, etc	Dpto. de Infraestructura
6	Información Digital	Información en soporte electrónico	Dpto. de Informática
7	Información Impresa	Información en formato papel	Dpto. de Informática
8	Identidad de Pacientes	Datos ingresados al sistema	Dpto. de Informática
9	Inmuebles	Oficinas, y todo lugar de trabajo	Dpto. de Infraestructura

## 11.5. Valoración de activos

A continuación se realiza en base a la escala de valores mostradas en la sección 11 una evaluación de los activos descriptos en el punto anterior:



<b>Id</b>	<b>Activo</b>	<b>Confidencialidad</b>	<b>Integridad</b>	<b>Disponibilidad</b>
1	Servicio Eléctrico	1	3	3
2	Servicio Internet	3	3	3
3	Personal Profesional	3	3	2
4	Sistema de Aplicación	3	3	3
5	Instalaciones	2	3	3
6	Información Digital	3	3	3
7	Información impresa	3	3	3
8	Identidad de Pacientes	3	3	3
9	Inmuebles	2	3	2

### 11.6. Análisis y estimación de riesgos

A partir de las valoraciones examinadas en 11.5, se realiza una estimación de riesgos y vulnerabilidades para tener un plan de contingencia en caso de alguna eventualidad.

<b>ID</b>	<b>Activo</b>	<b>Amenaza</b>	<b>Clasificación</b>	<b>Ocurrencia</b>	<b>Vulnerabilidad</b>	<b>Impacto</b>
1	Servicio Eléctrico	Corte del Suministro	Tecnológica	Media	Sin Plan de contingencia	Muy Alto
2	Servicio Internet	Corte del Suministro	Tecnológica	Baja	Sin Plan de contingencia	Muy Alto
3	Personal Profesional	Divulgación de la información	Humana	Baja	Falta de Confidencialidad	Alto
4	Sistema de Aplicación	Caída del Sistema	Tecnológica	Baja	Problemas con el HW o SW	Alto
5	Instalaciones	Robo de los aparatos	Tecnológica	Baja	Sin Firewalls, sin bloqueo de USB, etc	Muy Alto
6	Información Digital	Pérdida de la Información	Tecnológica	Baja	Sin Copia de Resguardo	Muy Alto
7	Información Empresa	Robo de la información	Operativa	Media	Falta de políticas de Escritorio Limpio y cuidado de los desechos	Muy Alto
8	Identidad de Pacientes	Robo de Identidades	Operativa	Media	Falta de Validación	Muy Alto
9	Inmuebles	Incendio	Física	Muy Baja	Falta de plan de Evacuación	Muy Alto

## 12. Diseño del Sistema

A continuación se hace visible el Diseño del Sistema teniendo en cuenta los siguientes puntos:

- Visualización del Modelo del Repositorio, junto con el detalle de cada componente.
- Diagrama del Modelo Cliente/Servidor
- Diagrama del Modelo de Capas.

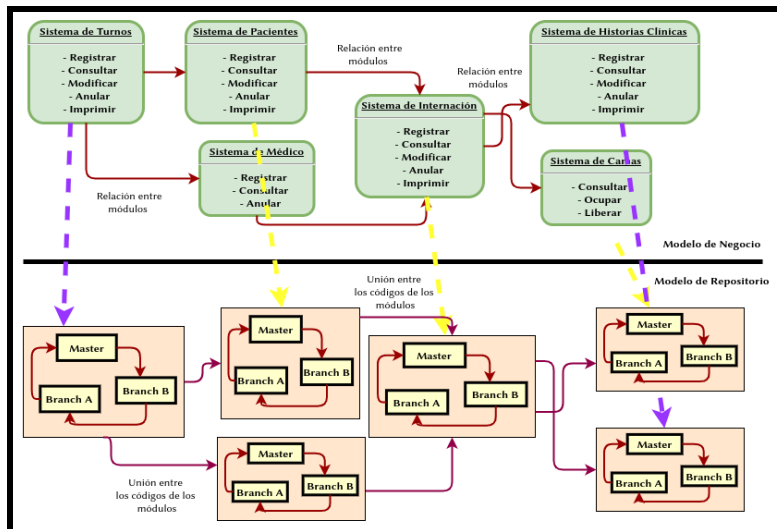
## 12.1. Modelo del Repositorio

La elección del software es una cuestión crucial para la implementación de un depósito de objetos digitales. Existen distintos modelos de tecnología según su origen y forma de adquisición: gratuito o comercial, software propietario o de código abierto, modelo de servicio de software. Se deben tener en cuenta los siguientes requisitos a la hora de elegir el repositorio de software:

- Apoyo a los diferentes formatos de archivo, escalabilidad, extensibilidad y mantenimiento del sistema.
- Aceptación de estándares de metadatos, descriptivos, de conservación, administrativos.
- Interoperatividad: cumplir con los principales protocolos de intercambio de registros de información ( OAI-PMH, Z39.50, SWORD ).
- Localización permanente de los documentos, mediante la incorporación de identificadores persistentes de objetos digitales como DOI, Handle.
- Aplicaciones de búsqueda y visualización de metadatos.
- Interfaz de búsqueda a texto completo.
- Autenticación y autorización de usuarios.
- Personalización del software (API).

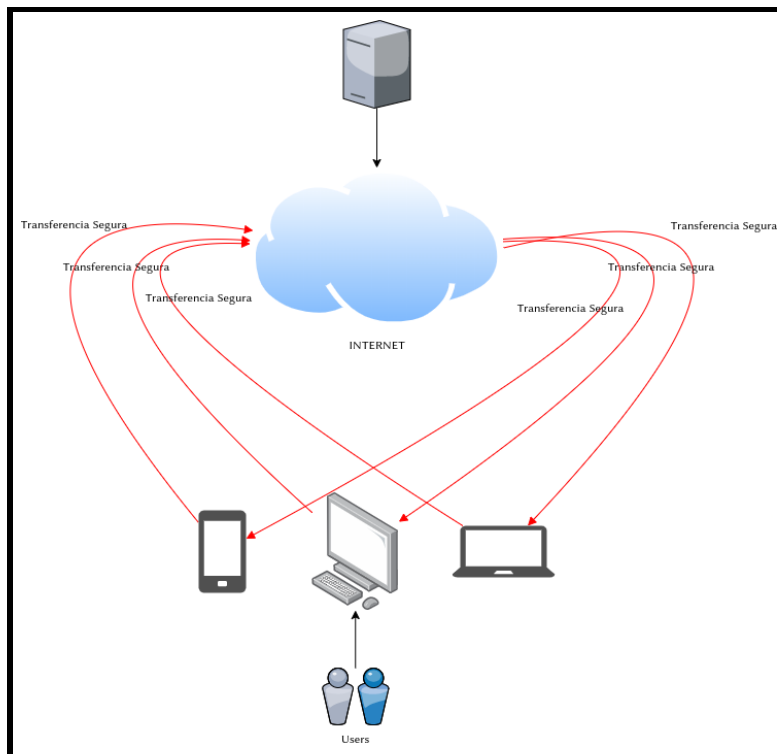
Para todo ello se hizo elección del repositorio de Github Enterprise, el cual permite una mayor velocidad de desarrollo, seguridad en cada paso del versionado, automatizaciones para cada flujo de trabajo, se permite el trabajo colaborativo definiendo qué colaboradores pueden participar y cuáles no, entre otras tantas ventajas más.

En el siguiente diagrama, podemos observar el tipo de relación que existe entre el Modelo de Negocios del proyecto y el modelo del Repositorio que se implementará.



## 12.2. Modelo Cliente/Servidor

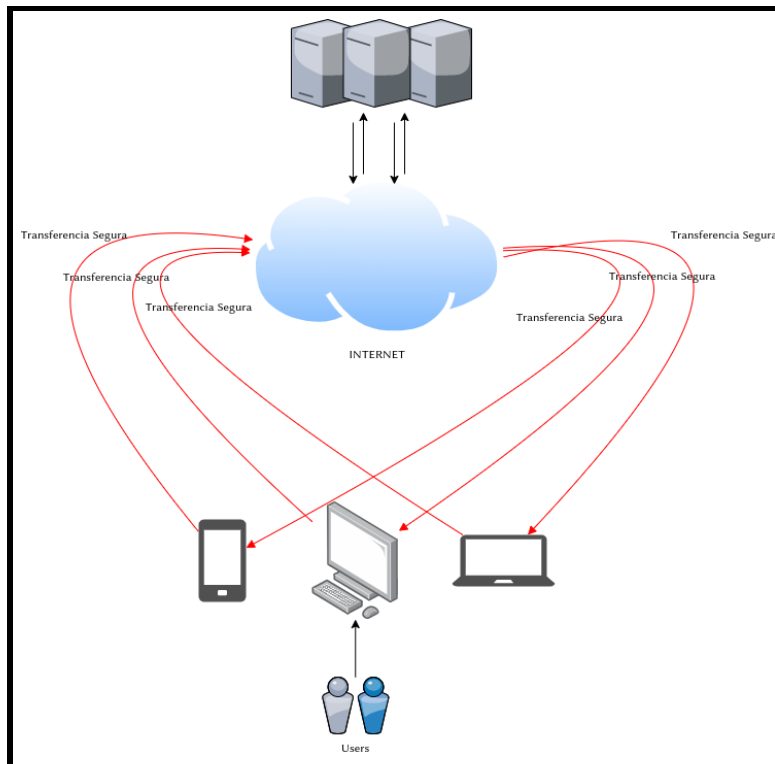
A continuación, podemos observar cómo se grafica el modelo *Cliente-Servidor* expuesto para el proyecto actual.



Cabe destacar algunos puntos:

- En una primera instancia, sólo los administradores serán a los que se les permita operar mediante el ABM del sistema de salud planteado. Aún así, el sistema comprende desde su construcción inicial, la posibilidad que muchos usuarios *consulten* el estado de sus turnos, historial clínico y demás entidades desde sus dispositivos, ya sean teléfonos celulares, computadoras de escritorio, laptops, notebooks y demás.
- Cabe destacar que las conexiones de envío y recepción son totalmente fiables y seguras, por lo que reducirá al máximo el peligro de tener ataques de infiltraciones, con el riesgo contingente de mermar la seguridad de todo el sistema completo.
- Es importante tener en cuenta como se hace mención en la sección 4.2, que el incremento de usuarios operando con el sistema de consultas, hará que la carga en la gestión tanto de consultas, como de transacciones por parte de los Administradores en el Servidor se incremente de forma exponencial, por lo que se pensó en etapas posteriores, incrementar los servidores disponibles, sin necesidad de afectar la usabilidad por parte de los usuarios.

Por lo dicho anteriormente, se puede establecer un diagrama de implementación para la fase final del proyecto, que se puede visualizar a continuación:



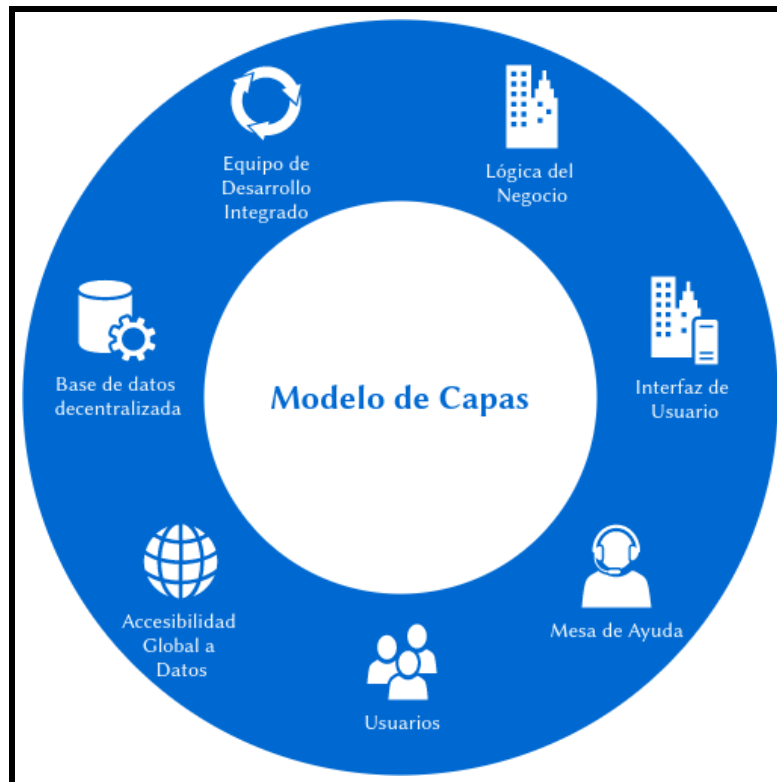
### 12.3. Modelo de Capas

Como bien se sabe, el modelo de capas permite que las aplicaciones que se vayan a diseñar, sean concebidas, desarrolladas y distribuidas en componentes, siguiendo el viejo precepto de “divide y vencerás”.

Algunas de las ventajas con las que cuenta un modelo de este tipo son las siguientes:

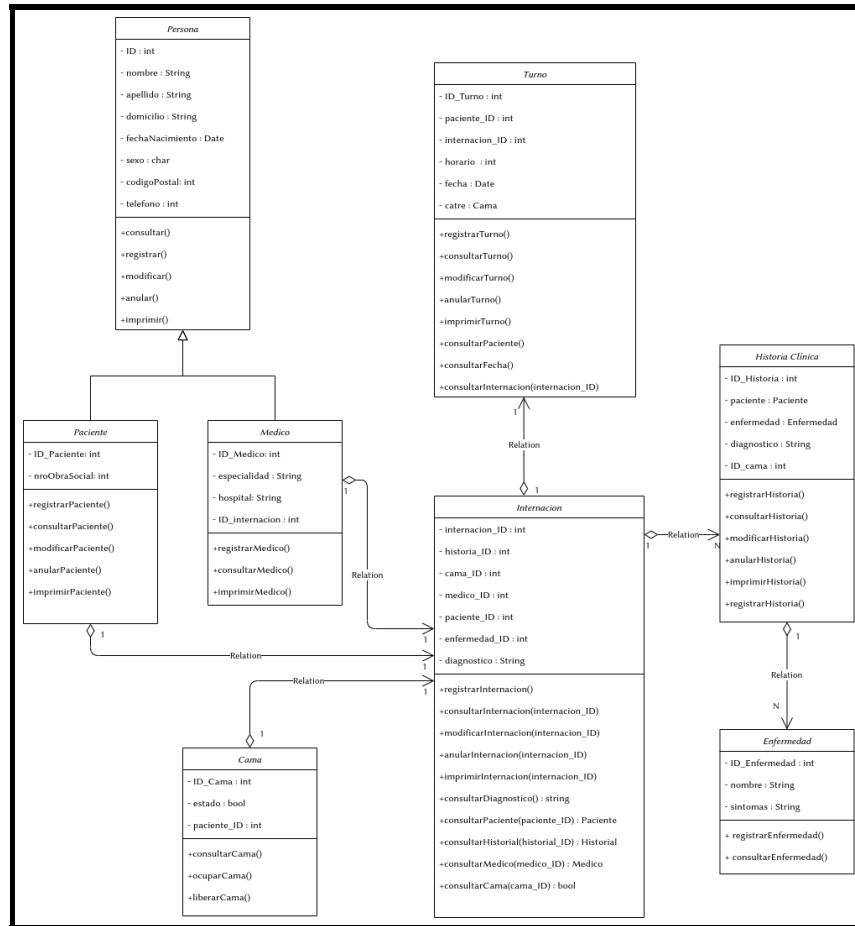
- Se reduce la complejidad a la hora de abordar el proyecto.
- Se facilita la distribución en distintos componentes para el desarrollo del proyecto.
- Se consigue mayor encapsulamiento.
- Permite una alta escalabilidad.
- Permite tener una mayor facilidad para desarrollar el software en múltiples plataformas (ya sean web, escritorio o mobile).

A continuación se expresa el Modelo de Capas a utilizar:



### 13. Diagrama de Clases

A continuación se visualiza el Diagrama de Clases implementado para el proyecto actual, en donde se tiene en cuenta las relaciones que existen entre las distintas entidades primigenias detectadas al comienzo del proyecto, en la sección :



A la hora de implementar el Diagrama de Clases, se tuvo que extraer el tipo *Médico*, generando de este modo, una nueva entidad, necesaria para poder realizar lo solicitado para la clase Interacción.

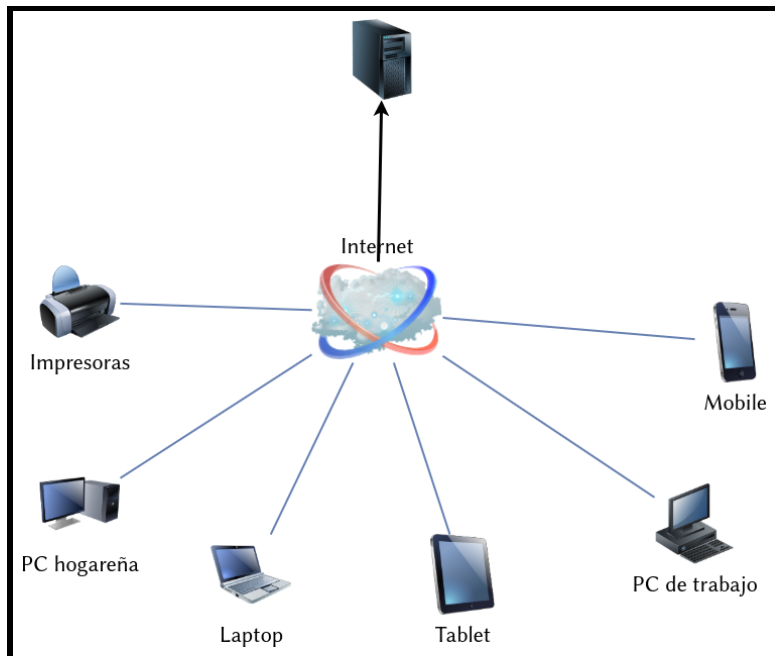
Ya que la entidad es de fácil definición, se consideró implementarla en el tercer Sprint.

### 14. Modelo de Control Centralizado

Por el tipo de proyecto que se tiene entre manos, y como se explicó anteriormente, se resuelve diagramar un Modelo de *Control Centralizado* ya que todas las consultas

y transacciones serán nucleadas en el servidor IBM X3700, el que luego, en una etapa tardía, se le adosará otros servidores para un manejo con mayor economía la carga de operaciones.

A continuación se tiene un tipo de esquema del descrito anteriormente:



## 15. Plan de Pruebas

A continuación se hace un Plan de Pruebas sobre el proyecto encausado, tomando en cuenta los siguientes puntos:

- Pruebas de Fiabilidad
- Pruebas de Software
- Pruebas de Interface

Una vez especificado cada una de las etapas por las que deberá de pasar los desarrollos del proyecto, se podrá contar con entregables comprobados para su posterior entrega y puesta en marcha de los primeros módulos del sistema.

A continuación se detalla cada uno de los puntos a acometer.

### 15.1. Pruebas de Fiabilidad

No hay duda que la fiabilidad de un programa de computadora es un elemento importante de su calidad general. Si un programa falla frecuentemente en su funcionamiento, no importa si el resto de los factores de calidad son aceptables.

¿Qué se entiende por *fallo*?

En este proyecto, definiremos como **fallo** a la falla de concordancia con los requisitos del software, determinados en la sección 7. Se debe tener en cuenta que hay distintos niveles de fallos y que pueden ir desde simplemente desconcertantes o incluso catastróficos. Puede que un fallo sea corregido en segundos mientras que otro lleve semanas o incluso meses. Teniendo en cuenta esto, la corrección de un fallo puede llevar a la introducción de otros errores que, finalmente, lleven a más fallos.

Es por ello que se decidió abordar el proyecto con un Test de Fiabilidad que cotemple los siguientes pasos:

1. Se establece un primer análisis del sistema existente, para comprender su uso en la práctica.
2. En base a ellos se realiza un set de datos para poder generar pruebas y obtener metricas sobre las fallas que van sucediendo.
3. Una vez realizadas las pruebas pertinentes, se generan aproximaciones estadísticas para concluir si el sistema tiene un margen seguro de fiabilidad o si por el contrario se estaría por debajo de ese margen aceptable.

## 15.2. Pruebas de Software

Durante el proceso de creación se utilizarán pruebas *Unitarias* y *de Integración* para comprobar que todo funciona de la manera esperada internamente por el equipo de desarrollo. Una vez que el sistema tiene una primera integración de funcionalidad, se utilizarán las *Pruebas Funcionales* para poder comprobar que los requisitos de funcionamiento se estén cumpliendo.

Un tercer paso será realizar *Pruebas de Carga* y *de Estrés* para poder asegurar que el sistema diagramado en la sección 12.2 va a ser capaz de dar respuesta al número de usuarios esperado. De este modo es posible detectar problemas que, simplemente con el código, no se podrán ver con facilidad, ayudando a encontrar los puntos neurálgicos del proyecto.

Y por último, un cuarto paso será detectar si el sistema puede ser consumido y utilizado en su completitud, por lo que se realizarán *Pruebas de Usabilidad*. Con ellas se podrán detectar problemas sobre la propia experiencia de usuario.

A continuación, se puede observar un gráfico donde se convalida las distintas pruebas realizadas al programa en conjunto:





A continuación se especifican cada una de las pruebas a realizar:

1. **Prueba Unitaria:** prueba la funcionalidad de una parte individual del programa. Se asegura de que cada parte de los programas se ejecute bien y esté alineado correctamente con el software.
2. **Pruebas Funcionales:** prueba la funcionalidad durante la creación, validación, etc. de casos de prueba
3. **Prueba de Seguridad:** prueba el mecanismo de seguridad. Incluye pruebas de penetración junto con la validación del control de acceso, cifrado, etc.
4. **Prueba de Carga:** al usar casos de prueba funcional, valida el rendimiento bajo carga.
5. **Detección de Errores en tiempo de ejecución:** aborda los problemas de carrera en tiempo de ejecución, pérdida de recursos, etc.
6. **Prueba de Flujo de Trabajo:** se realiza para asegurarse de que el motor de interfaz maneja bien el flujo de trabajo.

### 15.3. Pruebas de Interface

Una Prueba de Interfaz es un tipo de prueba de software que verifica la comunicación entre dos sistemas de software diferentes, en pocas palabras, comprueba la fiabilidad de la conexión establecida.

Una Prueba de Interfaz incluyen las pruebas de dos segmentos principales

- Segmento de la Interfaz del servidor web y servidor de aplicaciones.
- Segmento de la Interfaz del servidor de aplicaciones y del servidor de bases de datos.

En este proyecto se estableció por realizar implementar tres fases de Prueba de Interfaz en su ciclo de vida. A continuación se detallan los puntos que se harán hincapié:

## **15.4. Fases de una Prueba de Interface**

Se diagramó implementar las puebas en 3 fases:

- Configuración y Desarrollo.
- Validaciones.
- Mantenimiento.

### **15.4.1. Configuración y Desarrollo**

Después de la configuración de la interfaz y del inicio del desarrollo, es necesario verificar la configuración según los requisitos especificados anteriormente. En pocas palabras, tiene lugar la verificación descrita en la sección 15.1 la cual delimita el accionar del software.

### **15.4.2. Validaciones**

Después de la etapa de Configuración y Desarrollo, es necesaria la validación de la interfaz, mediante pequeñas pruebas de conexión realizadas entre los distintos módulos para ver que los datos ingresados sean los esperados y que estos puedan ser validados por el software codificado.

### **15.4.3. Mantenimiento**

Después de la finalización del proyecto, cuando alcanza su etapa de implementación, la interfaz está configurada para ser monitoreada en su desempeño.

Una cosa que debe tenerse en cuenta es que el código debe estar libre de defectos, para implementar esto, se deben realizar pruebas para verificar que cualquier código agregado no traiga problemas a futuros e indeseados.

Después de la realización de las pruebas, se realiza la validación de los datos y el flujo de trabajo. Por tanto, es importante mantener la funcionalidad original del programa.

## **15.5. ¿Por qué implementar Pruebas de Interface?**

Básicamente, en este proyecto se decidió que las directivas por las cuales se adoptarían Pruebas de Interfaces, serían las siguientes:

- Para garantizar que los usuarios finales o clientes no tengan ningún problema al utilizar un software en particular.
- Identificar las áreas de aplicación a las que suelen acceder los usuarios finales y comprobar si son fáciles de usar.
- Para verificar los requisitos de seguridad mientras la comunicación se propaga entre los sistemas.

- Para verificar si una solución es capaz de manejar fallas de red entre un servidor de aplicaciones y un sitio web.
- Para asegurarse de que el software sea competente en el manejo de errores.

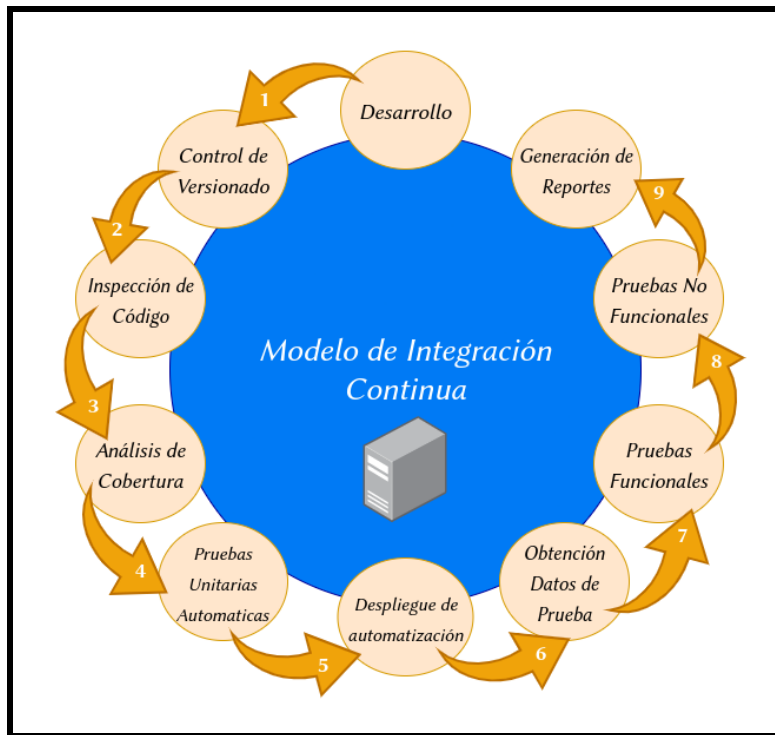
#### 15.5.1. Directivas para una Interfaz de Usuario

Para poder afrontar esto, se establecieron las siguientes directivas desde un principio para poder afrontar una interfaz de usuario exitosa:

- En lo posible, se desarrollarán códigos pequeños y que luego serán probados, en lugar de realizar directamente grandes fragmentos de códigos.
- Se deberá de evaluar alguna herramienta de automatización. Se pensó en la herramienta *Selenium* la cual permite afrontar pruebas automatizadas para cualquier aplicación web. También puede crear scripts, que es de gran ayuda para la reproducción rápida de errores, pruebas de regresión y pruebas exploratorias.
- Se marcarán los puntos de inicio y parada, que eventualmente nos ayudarán a verificar el rendimiento de la prueba.

## 16. Esquema de Integración

A continuación se puede visualizar el esquema del **Modelo de Integración Continua** que se propone implementar en el actual proyecto:



El ciclo presentado para este tipo de proyecto, comienza con la etapa de **Desarrollo**, la cual se llevará a cabo por el equipo comentado en la Sección 4.3.1, quien seguirán lo planificado en tiempo y forma según lo explicado en la sección 6.

Todo ello a medida que se vaya avanzando en la codificación en los distintos módulos, estará amparado por un **Control de Versionado** que permite la coherencia y transparencia que requiere la transversalidad del equipo de desarrollo, siguiendo los lineamientos comentados en la sección 12.1.

Al comienzo, la **Inspección de Código** se realizará mediante pares, pero a medida que vaya creciendo el desarrollo, se tiene pensado que para automatizar el proceso, a futuro se requerirá alguna herramienta como puede llegar a ser *Sonar*.

Con el **Análisis de Cobertura** se mide el grado en que el código fuente de los distintos módulos ha sido comprobado, permitiendo de esta manera determinar la calidad del test que se deba llevar a cabo, como también evidenciando las partes críticas que no hay sido comprobadas, como desestimando las partes que ya lo fueron.

Como se comentó en la sección 15.2, se realizarán **Pruebas Unitarias Automatizadas** para poder comprobar que todo funcione de manera esperable por el equipo de desarrollo.

En el proyecto actual, las pruebas integrales se aplicarán justo después de haber llevado a cabo cada Prueba Unitaria con la intención de probar los métodos aplicados en el desarrollo. Si no existen ningún problema de código y las pruebas unitarias han terminado de forma exitosa se podrá pasar al Test Integral para asegurarse de que en este punto no se produce ningún tipo de problema en la combinación de elementos

unitarios.

El motivo principal se encuentra en que el Test Integral lleva a cabo la revisión conjunta de los diferentes elementos que están presentes con el objetivo de conformar el software en su completitud. Se realiza la comprobación para ver que todo funciona de una manera adecuada en conjunto, dado que no es extraño que se produzcan alteraciones en el rendimiento.

Con esta comprobación representada por la Prueba de Integración podremos comprobar si la comunicación entre los distintos componentes presentes en el software es funcional. En el caso de ser necesario ir más allá por el posible surgimiento de nuevos subsistemas, el equipo de desarrollo tendrán que hacer la prueba específica de subsistemas, la cual vendría a ser una variación de la de integración pero profundizando en los elementos que están incluidos dentro de cada sistema.

Durante este proceso en el cual se verifican los distintos tipos de integración, el equipo de desarrollo junto con el cliente y profesionales afines, tendrán que ensamblar los módulos independientes, terminar de dar forma al software al completo y verificar el proceso a conciencia. Una de las ventajas de este esquema adoptado para el actual proyecto, se encuentra en la oportunidad de llevar a cabo pruebas de una manera paralela, lo que aporta flexibilidad extra en el proceso de calendarización. Para ello se optará por la elección de frameworks en los que las pruebas se puedan combinar con el desarrollo y con la supervisión simplificada de los procesos, especialmente en aquellos casos en los que las pruebas puedan ser un poco más complejas. El resultado garantizará que el proyecto de software pueda avanzar hacia su siguiente fase antes de darse por finalizado.

Ahora bien, mencionadas las Pruebas Unitarias y las Pruebas de Integración en la sección 15.2, el proceso de pruebas pasa por otras fases que resultan relevantes y se deben tener en cuenta para mejores resultados, como es el caso de **Despliegue de Automatización**, donde cada una de las pruebas programadas, se van testeando de forma exhaustiva para comprobar el funcionamiento de los módulos.

Una vez realizado el Despliegue de Automatización, y luego de comprobar que todas las pruebas hayan sido exitosas, se obtienen pasa a la **Obtención de Datos de Prueba** el cual será la alimentación del próximo ciclo de Integración.

Las **Pruebas Funcionales** es otro de los procesos que se tendrán que gestionar para alcanzar la mayor estabilidad y confianza en que el rendimiento sea el adecuado. Lo que hacemos en este caso, una vez que se visualizan que las conexiones están en forma a través del Test Integral, es sondear que el software que hemos diseñado y gestionado está actuando de manera conveniente teniendo en cuenta el objetivo para el cual fue creado (visualizado en la sección 6).

Asimismo se realizan las **Pruebas No Funcionales**. En esta etapa es donde se pueden realizar las denominadas *Pruebas de Carga y Estrés* mencionadas en la sección 15.2. En caso de existir errores insalvables, en esta etapa se puede realizar una *Recuperación o vuelta atrás*, la que permite volver un paso anterior y de esta manera, poder modificar, expandir, las Pruebas funcionales para que se tengan en cuenta los errores obtenidos en esta etapa de Pruebas No Funcionales. También se testean los errores referentes al proceso de instalación de cada uno de los entregables, como también se realizan pruebas estructurales y pruebas de configuración.

Una vez probado y asumido que todos los pasos fueron correctamente sucedidos

por cada una de las pruebas, entonces es el momento de la **Generación de Reportes**, etapa por la cual se pueden obtener las métricas del funcionamiento del software y con ello poder tener estadísticas del comportamiento, y cotejarlas con lo que se espera que realice. Estos reportes permiten mancomunar junto con el cliente o los usuarios finales, las distintas ideas que se tienen acerca de la evolución del proceso productivo y sobre la orientación que está teniendo en determinado momento el proyecto, permitiendo realizar una proyección del camino que se debe seguir para alcanzar los fines esperados.

## 16.1. Conclusión

Cada fase las pruebas tiene unas exigencias y unos puntos clave en los que se debe fijar la mirada. Eso es algo que se deberá tener en cuenta para que el testing resulte satisfactorio y no terminemos sufriendo demasiadas complicaciones. Por lo tanto, en la fase de la Prueba Funcional en lo único que nos tendremos que fijar es en que se produzca el tipo de soporte que diagramamos a lo largo del proyecto. Se procederá a analizar las salidas y las entradas que se produzcan al software, así como los resultados. No importa en este caso si en el diseño del software se ha encontrado algún tipo de defecto o posible mejora, dado que la cuestión en esta prueba consiste en comprobar el funcionamiento.

Por lo tanto, al finalizar, podremos decir que hemos pasado a través de un proceso que después de comprobar que todo está en su sitio nos da indicios si realmente tiene un buen rendimiento, algo clave para poder colocarse en el perfil del cliente y comprobar que vaya a obtener el nivel de satisfacción requerido.

## Bibliografía

- [1] Roger S. Pressman. *Ingeniería de Software*. McGraw Hill, 2010.