



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS  
VERTEX, CENTRO DE ENTORNOS INTERACTIVOS 3D, FACULTAD 4

## SISTEMA PARA LA ADMINISTRACIÓN DE CONTENIDOS DE LA PLATAFORMA WEB MEDICANDO

**Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas**

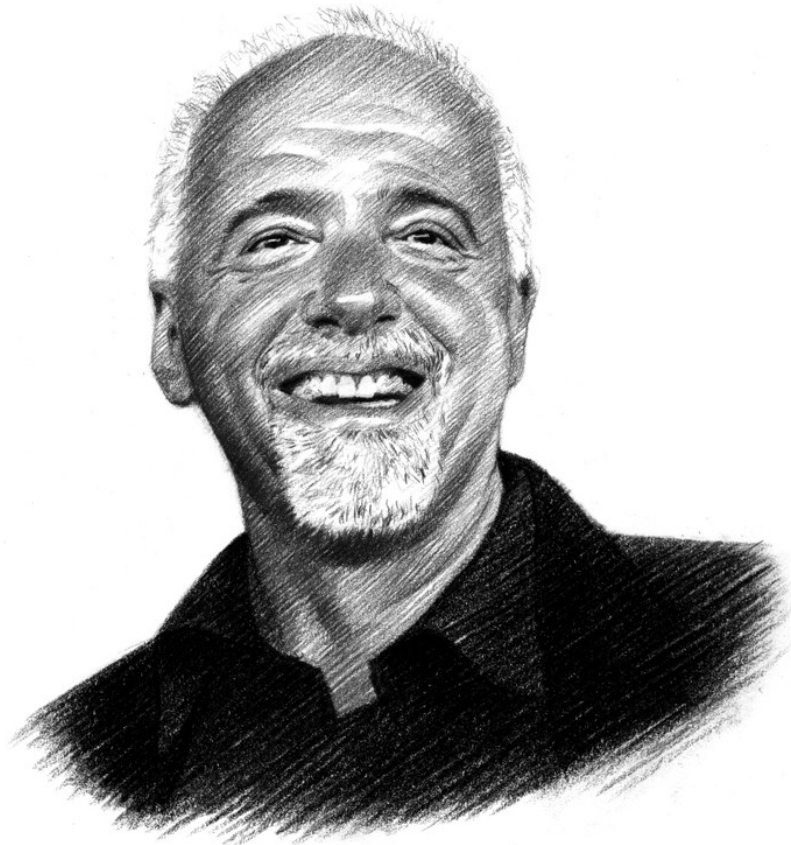
Autor: Solanch de la Caridad Ríos Portillo

Tutores: Ing. Reinaldo García Maturell

MSc. Andy Hernández Paez

Ing. Danier Lorenzo Pérez

**La Habana, 2018**



*"La gran victoria que hoy parece fácil, es el resultado de pequeñas victorias  
que pasaron desapercibidas"*  
*Pablo Coelho*

*A mis padres por ser mi ejemplo a seguir.*

*A mis familiares y amigos.*

*A Pedro Javier, por ser una de las personas más importantes en mi vida.*

---

## Agradecimientos

---

*A mis dos tesoros, mi mamá y mi hermano, por ser ejemplo a seguir y motivo de impulso para mi vida profesional, por seguir adelante a pesar de todas las dificultades que se presentan en el camino. También agradezco a mi padrastro, por ayudarme tanto en estos años y por brindarme siempre su ayuda incondicional.*

*A mi abuela por estar siempre al tanto de mis estudios y a mi tía Onelia, que a pesar de no estar en vida se que siempre ha estado a mi lado en los malos y buenos momentos de mi carrera.*

*A Pedro Javier por ser pareja, amigo y por apoyarme y ayudarme en todo momento.*

*A mis tutores Andy Hernández, por estar siempre presente y por el apoyo y la ayuda que me ha dado, a Reinaldo y Dani por la ayuda incondicional que me han dado y así salir adelante en este trabajo.*

*A mi decana Mayra Durán y a mi profesora guía Zaida por compartir momentos lindos y ayudarme en mi carrera.*

*A mis amigos Claudia, Yenisse, Yem, Dianne y Ernesto por soportarnos y ayudarnos cuando lo hemos necesitado durante estos cinco años de la carrera.*

---

## Declaración de autoría

---

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Solanch de la Caridad Ríos Portillo  
Autor

---

Ing. Reinaldo García Maturell  
Tutor

---

MSc. Andy Hernández Paez  
Tutor

---

Ing. Danier Lorenzo Pérez  
Tutor

El auge de los portales web mediante gestores de contenidos ha generado nuevos desafíos para los ingenieros de software. Cada día resulta más apremiante la necesidad de cumplir con los requerimientos de los usuarios, de crear y mantener una única aplicación para todas las plataformas, de modo que se cumpla con las restricciones del tiempo y se facilite la labor desempeñada por los profesionales. En la Universidad de las Ciencias Informáticas se encuentra el Centro de Entornos Interactivos 3D, Vertex, en el cual existe una línea de desarrollo centrada en la construcción de videojuegos y se encargan de apoyar el proceso de rehabilitación de pacientes en la esfera de la salud. En el centro Vertex recientemente se re-estructuró arquitectónicamente la plataforma web Medicando, la cual gestiona información relacionada con videojuegos y pacientes. Esta información que se muestra en dicha plataforma no se encuentra organizada y estructurada. En tal sentido, el objetivo del presente trabajo es desarrollar un sistema de administración de contenidos para la plataforma web Medicando que permita la organización y visualización de su información de manera dinámica. La solución desarrollada permite la gestión de menú, elementos de menú, secciones o contenidos, y preguntas frecuentes de dicha plataforma. El proceso de desarrollo de la propuesta de solución se guía utilizando las buenas prácticas ingenieriles propuestas por la metodología Programación Extrema. Las características y diseño de la solución se describieron mediante las historias de usuario, requisitos no funcionales, tarjetas CRC y tareas de programación. Además, se realizaron pruebas de unidad y aceptación con el objetivo de diseñar casos de pruebas consistentes para comprobar el funcionamiento adecuado de las operaciones implementadas.

**Palabras clave:** administración de contenidos, plataforma web Medicando, Vertex, videojuegos.

<b>Introducción</b>	<b>1</b>
<b>1 Fundamentación teórica</b>	<b>5</b>
1.1 Introducción del capítulo . . . . .	5
1.2 Portales web Médicos . . . . .	5
1.3 Gestión de contenidos en portales web médicos . . . . .	6
1.4 Plataforma web Medicando . . . . .	9
1.5 Sistemas Gestores de Contenidos . . . . .	10
1.5.1 Counter Management Systems (CMS) Joomla . . . . .	10
1.5.2 CMS WordPress . . . . .	10
1.5.3 CMS Drupal . . . . .	11
1.5.4 Tabla de comparación de los CMS . . . . .	11
1.6 Herramientas y tecnologías . . . . .	12
1.6.1 Metodologías de desarrollo . . . . .	12
1.6.2 Lenguaje de modelado . . . . .	14
1.6.3 Herramientas de modelado . . . . .	14
1.6.4 Servidores Web . . . . .	15
1.6.5 Marco de trabajo . . . . .	15
1.6.6 Lenguaje de programación . . . . .	17
1.6.7 Gestores de bases de datos . . . . .	17
1.6.8 Entornos de Desarrollo Integrado . . . . .	17
1.6.9 Selección del entorno de desarrollo . . . . .	18
1.6.10 Otras herramientas utilizadas . . . . .	18
1.7 Conclusiones del capítulo . . . . .	19
<b>2 Características y diseño del sistema</b>	<b>20</b>
2.1 Introducción del capítulo . . . . .	20
2.2 Propuesta del sistema . . . . .	20
2.3 Fase I: Planificación . . . . .	21
2.4 Historias de Usuario . . . . .	22

2.5	Requisitos no funcionales . . . . .	25
2.5.1	Estimación de esfuerzo por Historia de Usuario . . . . .	26
2.5.2	Desarrollo del Plan de Iteraciones . . . . .	26
2.5.3	Plan de duración de las Iteraciones . . . . .	27
2.5.4	Plan de Entrega . . . . .	27
2.6	Roles del sistema . . . . .	27
2.7	Fase II: Diseño del sistema . . . . .	28
2.7.1	Patrones de diseño . . . . .	30
2.8	Tarjetas CRC . . . . .	31
2.8.1	Tarjetas CRC . . . . .	31
2.9	Conclusiones del capítulo . . . . .	34
<b>3</b>	<b>Implementación y prueba del sistema</b>	<b>35</b>
3.1	Introducción del capítulo . . . . .	35
3.2	Fase III: Desarrollo . . . . .	35
3.2.1	Prototipos de interfaces funcionales . . . . .	39
3.3	Fase IV: Pruebas . . . . .	41
3.3.1	Pruebas de aceptación y unitarias . . . . .	41
3.3.2	Pruebas de aceptación para la Iteración I . . . . .	42
3.3.3	Pruebas de aceptación para la Iteración II . . . . .	46
3.4	Análisis de las pruebas de aceptación . . . . .	49
3.5	Conclusiones del capítulo . . . . .	51
	<b>Conclusiones</b>	<b>52</b>
	<b>Recomendaciones</b>	<b>53</b>
	<b>Acrónimos</b>	<b>54</b>
	<b>Referencias bibliográficas</b>	<b>55</b>
	<b>Apéndices</b>	<b>58</b>



---

## Índice de figuras

---

1.1	Representación del portal INFOMED . . . . .	7
1.2	Representación de VirtualMed. Portal para el aprendizaje docente-educativo . . . . .	7
1.3	Representación del portal del Centro de Ingeniería Genética y Biotecnología . . . . .	8
1.4	Representación del portal de la Biblioteca Virtual de la Universidad de Cienfuegos . . . . .	9
2.1	Prototipo de diseño del sitio. (Fuente: Elaboración propia). . . . .	22
2.2	Representación de los elementos de menú del sitio y sus secciones correspondientes . . . . .	23
2.3	. . . . .	29
2.4	Prototipo de Tarjeta Clase-Responsabilidad-Colaborador (CRC). (Fuente: Elaboración propia). . . . .	31
3.1	Representación de la sección Patologías del sistema. (Fuente: Elaboración Propia). . . . .	40
3.2	Representación de la sección Galería del sistema. (Fuente: Elaboración Propia). . . . .	40
3.3	Representación de la gestión de secciones en la administración del sistema. (Fuente: Elaboración Propia). . . . .	40
3.4	Representación de la sección Menú del sistema. (Fuente: Elaboración Propia). . . . .	41
3.5	Representación de la gestión de menú en la administración del sistema. (Fuente: Elaboración Propia). . . . .	41
6	Prueba unitaria para la tarea de ingeniería Eliminar Sección. . . . .	64
7	Prueba unitaria para la tarea de ingeniería Actualizar Sección. . . . .	65
8	Prueba unitaria para la tarea de ingeniería Visualizar Sección. . . . .	65
9	Prueba unitaria para la tarea de ingeniería Crear Sección. . . . .	66

---

## Índice de tablas

---

1.1	Comparación entre Sistemas Homólogos. (Fuente: Elaboración Propia)	9
1.2	Roles del sistema	10
1.3	Comparación entre CMS. (Fuente: Reinaldo García Maturell, 2014)	12
1.4	Comparación entre metodologías de desarrollo. (Fuente: Elaboración Propia)	14
2.1	Historia de usuario # 1	23
2.2	Historia de usuario # 2	24
2.3	Estimación de esfuerzo por Historia de Usuario. (Fuente: Elaboración propia)	27
2.4	Plan de duración de las iteraciones. (Fuente: Elaboración propia)	27
2.5	Plan de entrega de versiones. (Fuente: Elaboración propia)	27
2.6	Roles de usuario. (Fuente: Elaboración propia)	28
2.7	Tarjeta CRC # 1	32
2.8	Tarjeta CRC # 2	32
2.9	Tarjeta CRC # 3	32
2.10	Tarjeta CRC # 4	33
3.1	Tarea de ingeniería # 1	35
3.2	Tarea de ingeniería # 2	36
3.3	Tarea de ingeniería # 3	36
3.4	Tarea de ingeniería # 4	36
3.5	Tarea de ingeniería # 5	37
3.6	Tarea de ingeniería # 6	37
3.7	Tarea de ingeniería # 7	37
3.8	Tarea de ingeniería # 8	38
3.9	Tarea de ingeniería # 9	38
3.10	Tarea de ingeniería # 10	38
3.11	Tarea de ingeniería # 11	39
3.12	Tarea de ingeniería # 12	39
3.13	Prueba de aceptación # 1	42
3.14	Prueba de aceptación # 2	43
3.15	Prueba de aceptación # 3	44

3.16	Prueba de aceptación # 4	44
3.17	Prueba de aceptación # 5	45
3.18	Prueba de aceptación # 6	45
3.19	Prueba de aceptación # 7	46
3.20	Prueba de aceptación # 8	46
3.21	Prueba de aceptación # 9	47
3.22	Prueba de aceptación # 10	47
3.23	Prueba de aceptación # 11	48
3.24	Prueba de aceptación # 12	48
25	Historia de usuario # 3	59
26	Historia de usuario # 4	59
27	Historia de usuario # 5	60
28	Tarea de ingeniería # 13	61
29	Tarea de ingeniería # 14	61
30	Tarea de ingeniería # 15	62
31	Tarea de ingeniería # 16	62
32	Tarea de ingeniería # 17	62
33	Tarea de ingeniería # 18	63
34	Tarea de ingeniería # 19	63
35	Tarea de ingeniería # 20	63
36	Tarea de ingeniería # 21	63
37	Tarea de ingeniería # 22	64

La gestión de contenidos está relacionada con todos los procedimientos y procesos involucrados en la agregación, transformación, catalogación, agrupación, autorización, presentación y distribución de información útil para nuevos propósitos. La gestión de contenidos abarca una función importante del profesional de la información, esta es una actividad fundamental en el entorno académico y profesional.

En la concepción actual la popularidad, versatilidad, uso y aplicación de los sistemas de gestión de contenidos (SGC), han transformado la manera tradicional de desarrollar sitios y aplicaciones web, la cual se basa en lenguajes avanzados de programación. El control de los procesos en sitios web es un elemento fundamental, por lo que es posible delinear un panorama en el cual las herramientas de gestión documental incorporan a sus prestaciones. [1].

En Cuba existen diferentes sistemas que gestionan información entre los que se encuentran:

- **VirtualMed:** Es un portal que administra los contenidos que serán mostrados en el mismo. Provee múltiples funcionalidades y soluciones esenciales para las dificultades que enfrenta el proceso docente-educativo, soportado en redes [2].
- El Centro de Gestión de Información (CGI) del Centro de Estudios Avanzados de Cuba (CEAC), tiene como misión proveer fuentes de información electrónicas, así como orientación y ayuda informativa a los investigadores del centro, y con miras al resto de la comunidad científica. Su visión es convertirse en el principal Centro Gestor de Información Nanotecnológica para la red de los centros de investigaciones del país [3].
- El portal Infomed(Centro de Información para la medicina) se encuentra soportado sobre el sistema gestor de contenidos Plone, constituyendo una herramienta para el personal de salud, tanto extranjeros como cubanos. Su propósito es trabajar para construir colectivamente un ecosistema de personas, servicios y fuentes de información para la salud, que permita el acceso oportuno y eficiente a la información de calidad, y que despliegue las capacidades creativas de los miembros de la red como productores de información y conocimientos, para lograr las metas de salud de nuestro país y de otros. De igual modo se proyecta como una organización líder en el campo de la información en ciencias de la salud, sustentada en una dinámica y eficiente red de conocimientos de alto valor profesional y humano [4].
- La biblioteca virtual de la universidad de Cienfuegos, constituye uno de los principales servicios que brinda la red universitaria y tiene como objetivo facilitar el acceso y gestión de la información y conocimientos en diferentes áreas del saber [5].

Estos sistemas permiten que el cliente que interactúe con el sitio se sienta satisfecho con la estructura de información que brinda. La gestión de la investigación, el desarrollo y la innovación constituye un factor clave de éxito para la competitividad de las industrias de alta tecnología.

La [Universidad de las Ciencias Informáticas \(UCI\)](#) como parte del proceso de informatización de la sociedad cubana, ha creado un conjunto de proyectos de los cuáles varios están enmarcados en el área de la Telemedicina. Con el fin de apoyar algunos procesos de rehabilitación, el centro de Entornos Interactivos 3D (Vertex) ha incursionado en el desarrollo de videojuegos, que ofrecen la oportunidad de controlar al paciente mediante los mismos, permitiéndoles llevar el tratamiento al hogar y con una perspectiva recreativa. Los videojuegos desarrollados, almacenan cierta información que es utilizada por la plataforma web Medicando, una herramienta que proporciona al personal de salud un medio para llevar un control de la evolución de los pacientes. También permite llevar el control de los doctores y pacientes que interactúan con el sistema, además de los tratamientos y las estadísticas que estos videojuegos generan. A partir del establecimiento de roles a los usuarios del sistema, es posible establecer un control de acceso y el nivel de seguridad al mismo.

A pesar de ser una herramienta útil a la hora de apoyar el diagnóstico de pacientes mediante videojuegos, Medicando mantiene algunas deficiencias que necesitan ser atendidas. Dado lo antes expuesto se define como **situación problemática**:

- Es nula la interacción del paciente con la plataforma, por lo que los servicios que presta no son visibles en la misma.
- No permite que los interesados puedan seleccionar la información que necesitan y exportarla o extraerla para análisis posteriores.

Teniendo en cuenta la situación antes descrita, se presenta el siguiente **problema de la investigación**: ¿Cómo contribuir a la organización y visualización de la información para el usuario en la plataforma web Medicando? A partir del problema planteado en la investigación, se identifica como **objeto de estudio**: Sistemas de gestión de contenidos. Para dar solución al problema de investigación planteado se define el siguiente **objetivo general**: Desarrollar un sistema para la administración de contenidos de la plataforma web Medicando. Todo lo cual delimita como campo de acción **campo de acción**: Sistema para la administración de contenidos para la plataforma web Medicando.

Para dar solución al problema planteado se definen las siguientes tareas para la investigación:

- Elaboración de un marco teórico de la investigación a través del estudio del estado del arte.
- Identificación de los principales elementos que componen los Portales web.
- Caracterización de los [CMS](#) existentes tanto a nivel nacional como internacional.
- Realización del levantamiento de requisitos funcionales y no funcionales.
- Implementación de un sistema que permita la administración de contenidos de la plataforma web Medicando, brindando así solución al problema planteado.
- Realización de pruebas para validar la correcta implementación de los requerimientos definidos.

Para la realización de las tareas expuestas anteriormente, se aplican métodos teóricos y empíricos, los que se muestran a continuación.

Los **métodos teóricos** usados son:

**Método Histórico-Lógico:** Permite profundizar cómo ha evolucionado el desarrollo de las TIC en el proceso de rehabilitación, y como se han incrementado las funcionalidades en los mismos.

**Método Analítico-Sintético:** Se emplea en el momento de buscar la esencia del tema en cuestión, concepto de gestores de contenidos; posibilitando encontrar los elementos más importantes que relacionan los CMS con la gestión de contenidos en portales web médicos, para identificar los términos y conceptos que sirvan de apoyo a la creación del marco teórico del trabajo.

Los **métodos empíricos** son:

**Estudio de la documentación:** Lo cual permite la elaboración del marco teórico de la investigación y su correcto estudio desde el punto de vista metodológico.

**Realización de pruebas o test:** Permite probar el correcto funcionamiento de un módulo o componente del código y a la aplicación, por lo que asegura que cada elemento de estos funcione correctamente por separado y conjuntamente.

**Observación:** Este método se utiliza para observar cómo es el comportamiento de los Sistemas Informáticos, así como sus principales problemas. El presente trabajo de diploma está estructurado de la siguiente manera: resumen, introducción, tres capítulos de contenido, conclusiones generales, recomendaciones, referencias bibliográficas, bibliografía consultada, glosario y anexos.

**Entrevista:** El empleo de este método constituye un medio para el conocimiento cualitativo de la situación existente, el mismo se utiliza principalmente con el objetivo de obtener información sobre los principales elementos que componen a los sistemas gestores de contenidos y los portales web médicos que deben estar presentes en la propuesta de solución.

**Nivel Matemático o Estadístico:** Este método se utiliza para el cálculo de la complejidad ciclomática en las pruebas de unidad, con el objetivo de determinar los caminos lineales independientes que pueden cubrirse con el diseño de casos de prueba. También se emplea para el análisis de los resultados de las iteraciones de pruebas ejecutadas y el nivel de satisfacción con la solución propuesta.

**Estructura de la investigación:**

**Capítulo 1:** Fundamentación Teórica. Recoge todo lo que concierne al objeto de estudio, se analizan los principales conceptos sobre los Portales Web y la gestión de contenidos en los mismos a través de una revisión bibliográfica, observando sus ventajas, limitaciones y los requisitos necesarios a tener en cuenta en su elaboración. Se realiza un estudio de las metodologías y herramientas de desarrollo que se pueden utilizar, así como su fundamentación y el estado del arte en Cuba y en otras regiones.

**Capítulo 2:** Propuesta de solución. Contiene información asociada al sistema. En este capítulo se abordan las fases de Exploración, Planificación e Iteraciones que incluye el diseño del sistema, todas estas etapas propias de la metodología de desarrollo utilizada. Se describen los artefactos obtenidos según la metodología. Se realiza el modelo de datos y se especifican los requisitos funcionales y no funcionales que deberá

cumplir la solución, además de la evaluación de los resultados parciales obtenidos en la investigación.

**Capítulo 3:** Desarrollo y Prueba del sistema. Muestra las actividades propuestas por la metodología seleccionada, diseño, implementación y prueba de la propuesta de solución del Portal Medicando, para el apoyo de la rehabilitación en pacientes con algún impedimento físico.

### 1.1. Introducción del capítulo

En el presente capítulo se analizarán los conceptos elementales que guiarán la investigación, con el objetivo de darle solución al problema planteado. Se realizará un estudio de homólogos de portales médicos desarrollados en Cuba, además de la forma en que se gestiona los contenidos en los sitios web. Se realizará un estudio de la arquitectura de información que contienen los portales, así como la manera en que los CMS muestran el contenido en la web. Se definirá la metodología, las herramientas, el lenguaje de programación a utilizar para el proceso de desarrollo del portal web.

### 1.2. Portales web Médicos

La mayoría de los portales médicos se caracterizan por su desinterés económico siendo patrocinados por instituciones, ministerios y empresas del sector de la medicina. La naturaleza en línea de estos almacenes de información, permite al usuario acceder a sus contenidos de forma ágil, sencilla e instantánea lo que sumado a la gran calidad del producto final, descubren que este tipo de estructuras constituyen una de las mejores soluciones en la construcción del conocimiento. Existe una gran variedad de portales médicos, los cuales dependen del contenido que se desee transmitir y el método que se utilice [6].

Se realizará un estudio de homólogos con el objetivo de evaluar los elementos comunes entre los sistemas médicos que gestionan información, además, de la forma en que gestionan las posiciones, los menú que contienen los sitios y a su vez la forma en que muestran los contenidos o secciones.

#### **Características de los portales médicos**

Entre las características que poseen estos portales se encuentran:

- Proporcionan información de todo tipo a médicos, pacientes y usuarios externos, así como instrumentos para realizar búsquedas.



- Contribuyen a la formación del doctorado, mediante informaciones diversas y cursos de actualización de conocimientos.
- Brindan información de manera estructurada de forma tal que el paciente se sienta satisfecho con el contenido adquirido.
- Permite practicar y entrenar sin poner en riesgo la vida de los pacientes tantas veces como quiera.

### 1.3. Gestión de contenidos en portales web médicos

Los portales basados en la gestión de contenidos aportan información y servicios con alto valor agregado, regidos por estándares de calidad y dirigidos a satisfacer las necesidades de información de los usuarios para los cuales están diseñadas. Las redes de medicina incluye especialistas de la salud, de las tecnologías y de las ciencias de la información; los cuales desarrollan funciones asistenciales, investigativas, docentes y gerenciales en las instituciones del sector de la salud en Cuba, así como en el ámbito internacional.

**Infomed:** es la primera red electrónica cubana de información para la salud y surgió como parte de un proyecto del Centro Nacional de Información de Ciencias Médicas (CNICM) de Cuba para facilitar el intercambio de información por medios electrónicos entre un conjunto de bibliotecas, centros de información y otras entidades que conforman el Sistema Nacional de Información de Ciencias Médicas (SNICM) del Ministerio de Salud Pública (MINSAP). Fue creada para mejorar los servicios de información de la salud en el país mediante la gestión de secciones en las posiciones del sitio y brindarle al usuario todo lo referente al mismo, además permite gestionar la información que se le muestra al personal de salud, mediante la administración de contenidos.

Entre los servicios que presta Infomed se encuentran:

- Acerca de...
- Grupos
- Foros
- Guías de Información
- Internet
- Red de portales
- Capacitación
- Contacto
- Participación oral
- Servicio de correo

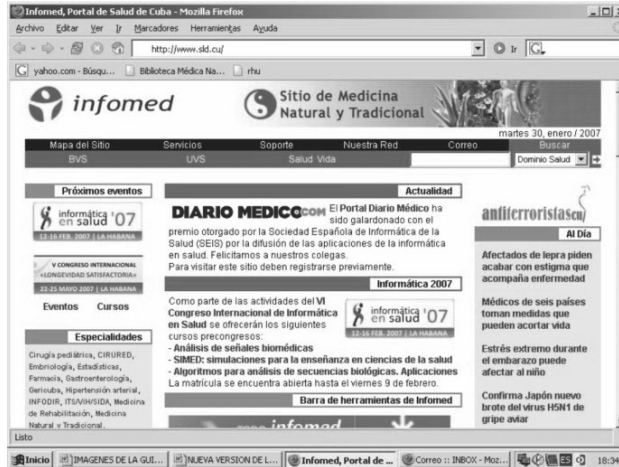


Figura 1.1. Representación del portal INFOMED

**VirtualMed:** Es un sistema gestor de contenidos que provee múltiples funcionalidades y soluciones esenciales para las dificultades que enfrenta el proceso docente-educativo, soportado en redes:

- La actualización constante de los contenidos por el profesor, un elemento importante si se considera la dinámica actual de cambio de la información sobre un tema.
- El aumento de la disponibilidad de materiales docentes y de su facilidad de adquisición.
- El control y actualización del curso del proceso evaluativo en un proceso educativo de carácter masivo.
- El registro de profesores y estudiantes propios de la gestión académica, con fines organizativos que faciliten la obtención de indicadores de volumen y calidad de servicios prestados.
- La posibilidad de generación de bases de datos que posibiliten futuras investigaciones metodológicas, mediante técnicas estadísticas clásicas o procedimientos de extracción de conocimiento en bases de datos [7].

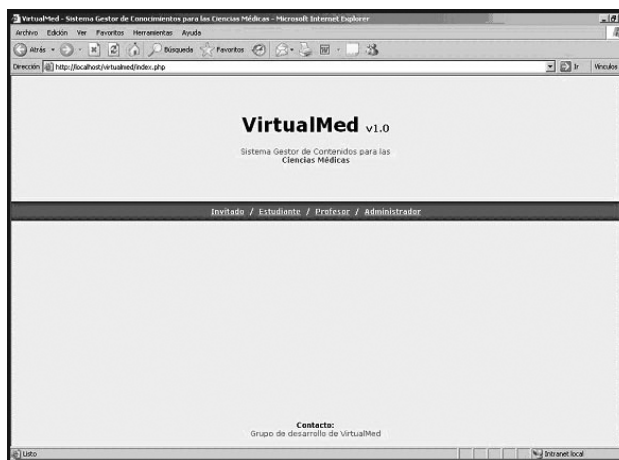


Figura 1.2. Representación de VirtualMed. Portal para el aprendizaje docente-educativo

**El Centro de Ingeniería Genética y Biotecnología (CIGB), portal de investigaciones sobre la medicina:**

- Es un sistema que provee múltiples funcionalidades y se encarga de brindarle al usuario la información referente a la rama de biotecnología, mediante la administración de contenidos, este sistema esta basado en la gestión de la información mediante las secciones, en posiciones fijas.
- Tiene un papel integrador en la esfera de la biotecnología cubana, con alta capacidad científico-técnica. Asume la responsabilidad de contribuir directamente en el desarrollo económico y social del país.
- Su desempeño se proyecta en las investigaciones, generando conocimientos para el desarrollo de nuevos productos, producción, servicios y la actividad comercial; basado en un sistema de calidad que garantiza la satisfacción de los clientes y teniendo en cuenta la dimensión ambiental [8].



Figura 1.3. Representación del portal del Centro de Ingeniería Genética y Biotecnología

**Biblioteca virtual de la universidad de Cienfuegos:**

La biblioteca virtual de la universidad de Cienfuegos es realizada en el Centro de Recursos para el Aprendizaje y la Investigación de dicha universidad y responde a las acciones del Proyecto CITMA Gestión de Contenido para la Comunidad. La creación de esta biblioteca fue mediante el empleo de recursos informáticos y bibliográficos para lograr un mejor uso y explotación del patrimonio docente e investigativo digital en la Universidad de Cienfuegos. Para su realización se utilizó el UML (*Unified Modeling Language*) como lenguaje de modelado y el Sistema Gestor de Contenidos Plone para la implementación del producto informático. El mismo fue validado utilizando estadísticas de accesos web al sistema y la técnica de Iadov para conocer el grado de satisfacción de los usuarios.

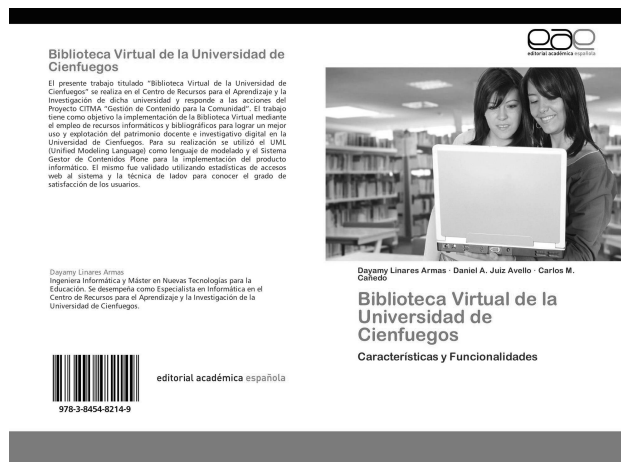


Figura 1.4. Representación del portal de la Biblioteca Virtual de la Universidad de Cienfuegos

Tabla 1.1. Comparación entre Sistemas Homólogos. (Fuente: Elaboración Propia)

Sistemas homólogos	Secciones	Posiciones	Tipos de menú	Elementos de menú
Infomed	Alto	Alto	Menú Principal (MP), Menú Lateral Derecho (MLD), Menú Lateral Izquierdo (MLI)	Acerca de..., Inicio, Contacto, Servicio de correo
VirtualMed	Medio	Medio	MP	Acerca de..., Contacto, Inicio, FAQs
Centro de Ingeniería Genética y Biotecnología	Bajo	Alto	MP, MLD	Acerca de..., Contacto, Inicio, FAQs
Biblioteca virtual de Cienfuegos	Medio	Alto	MP, MLD, MLI	Acerca de..., Contacto, Inicio, FAQs, Noticias

**1.4. Plataforma web Medicando**

Medicando es una plataforma web desarrollada con el objetivo de facilitar el seguimiento de pacientes en los tratamientos que les son asignados en el sistema, donde cada tratamiento debe estar relacionado con un videojuego de rehabilitación. Esta herramienta fue desarrollada con tecnología *HyperText Preprocessor* (PHP), para ello se utilizó como marco de trabajo el framework Yii; mientras que la gestión de los datos es administrada utilizando el SGBD MySQL y como servidor Apache Server.

Los usuarios son identificados por la plataforma en correspondencia con el rol que desempeñen, estos se asignan de acuerdo a la interacción que tendrá cada usuario con el sistema. Los roles que se definen son

Paciente, Doctor y Administrador; donde cada rol puede ejecutar un grupo de operaciones específicas [9]. La siguiente tabla describe las principales funcionalidades que son accesibles por los diferentes roles:

Tabla 1.2. Roles del sistema

<b>Roles</b>	<b>Interacción en Medicando</b>
<b>Paciente</b>	Asignado a los usuarios registrados en la aplicación que recibirán tratamientos con videojuegos. Estos pueden consultar los tratamientos que tienen asignados y acceder a la vez al videojuego asociado. También tienen la posibilidad de subir al sistema la salva de las sesiones de rehabilitación realizadas sin conexión.
<b>Doctor</b>	Responsable de las tareas de control y seguimiento de los pacientes registrados en el sistema. Puede crear y asignar tratamientos a pacientes, consultar estadísticas, gestionar videojuegos, registrar y actualizar datos de pacientes.
<b>Administrador</b>	Responsable de mantener el funcionamiento correcto de la herramienta. Posee permisos para ejecutar todas las funciones administrativas de la plataforma y es el encargado de gestionar la información. Establece los permisos pertinentes para los diferentes roles de usuarios.

## 1.5. Sistemas Gestores de Contenidos

Un **CMS** es generalmente un conjunto de herramientas, apoyado habitualmente por una base de datos, consiste en una serie de programas en un servidor web, y, opcionalmente, una serie de programas en la parte del cliente que permiten acceder a los mismos. Desde el punto de vista del cliente, se trata de un sitio web diligente, con apariencia e interfaz uniforme, con un diseño centrado en el usuario, y que permite llevar a cabo fácilmente las tareas para las que ha sido diseñado [10].

### 1.5.1. CMS Joomla

Joomla es un CMS desarrollado a partir del CMS « Mambo ». Fue creado como resultado a las diferencias entre los programadores principales y la empresa coordinadora del desarrollo. Consiguió el respaldo de programadores y diseñadores mediante un nuevo software libre y muchos cambios importantes en el código, a partir de aquí evolucionó hasta convertirse en uno de los CMS más usados y conocidos del mercado para la creación y mantenimiento de sitios web. También está desarrollado en PHP y su versión actual es la 3.8 [11].

### 1.5.2. CMS WordPress

WordPress es un sistema gestor de contenido que se diseñó inicialmente como una plataforma de blogs. WordPress contiene gran cantidad de complementos lanzados por desarrolladores independientes. De hecho, cada aspecto del sitio web con respecto a la creación, organización y motor de búsqueda, mejoran el

funcionalidad de la interfaz de usuario. Este sistema gestor de contenidos contiene plugins que permiten simplificar tareas en particular.

**Los plugins** son complementos (*software*) que aumentan las capacidades y posibilidades de WordPress, estos se usan para mejorar WordPress en diferentes áreas como *marketing*, redes sociales, seguridad, diseño web, contenido, tráfico web, etc.

**Los temas** son plantillas que se utilizan en WordPress para modificar la apariencia y diseño del sitio. Existe un sinfín de temas gratuitos y *premium* (de pago) que ayudan a adaptar a WordPress a cualquier tipo de necesidad y tipo de página web.

**Los widgets** son pequeños bloques de información que se utilizan en el CMS WordPress. Ayudan a darle al usuario un mayor control sobre el diseño y contenido de su sitio Web o blog. Los widgets se pueden expandir y usar de diferente forma dependiendo del tema y plugins que se instalen [12].

### 1.5.3. CMS Drupal

Drupal es un CMS que le permiten al usuario poner al día sus páginas sin el conocimiento técnico y asegurar que encaja el flujo de trabajo de su organización. Drupal es un CMS o sistema de gestión de contenidos que se utiliza para crear sitios web dinámicos y con gran variedad de funcionalidades, además, es un sistema de gestión de contenido web muy versátil, con licencia GNU/GPL, es un software libre, escrito en PHP, que cuenta con una amplia y activa comunidad de usuarios y desarrolladores que colaboran conjuntamente en su mejora y ampliación.

Drupal se destaca por su flexibilidad, adaptabilidad y por su interfaz de administración que permite crear fácilmente nuevos tipos de contenido estructurados. También contiene gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitios web, tiendas online, foros de discusión, webs corporativas [13].

### 1.5.4. Tabla de comparación de los CMS

Para realizar la selección del CMS a utilizar se tuvieron en cuenta aspectos y características comunes en cada uno de los sistemas gestores de contenidos: requisitos de sistemas, seguridad, soporte, facilidad de uso, rendimiento, administración, interoperabilidad, flexibilidad, extensiones o comercio electrónico, la cual se relaciona a continuación.

La columna **peso** refleja la importancia del aspecto estudiado dentro de los requisitos funcionales del portal a desarrollar.

Las valoraciones se dividen en tres niveles: **Regular:** el CMS ofrece un soporte correcto pero estrictamente básico para el aspecto analizado; **Bien:** el CMS cuenta con funcionalidades para un aspecto estudiado que mejoran sensiblemente las ofrecidas por el resto de sistemas gestores; **Excelente:** el CMS dispone, ya sea en la instalación básica o en módulos que estén disponibles, de las mejores características o funcionalidades posibles para un aspecto concreto.

Tabla 1.3. Comparación entre CMS. (Fuente: Reinaldo García Maturell, 2014)

Aspectos de CMS	Peso	Drupal 8.2.4	WordPress 4.9.4	Joomla 3.8
Facilidad de instalación y administración	****	Bien	Excelente	Excelente
Facilidad de uso	***	Excelente	Excelente	Excelente
Portencia gráfica y estructural	***	Excelente	Excelente	Excelente
Gestión de usuarios y workflow	***	Bien	Regular	Excelente
Funcionalidad web 2.0	***	Bien	Excelente	Excelente
Posibilidades de extensión e integración	**	Bien	Excelente	Excelente
Seguridad	**	Bien	Regular	Bien
Soporte, tamaño de la comunidad	**	Excelente	Excelente	Excelente
Soporte nativo semántico (RDF, RDFa, ...)	****	Bien	Regular	Bien
Plataforma	**	PHP	PHP	PHP
Arquitectura	****	Bien	Bien	Excelente

Se decide escoger elementos de la arquitectura de información de cada uno de los CMS, debido a que ofrece características a través de plantillas, widgets y módulos, además de extensiones y adaptaciones. Estos CMS permiten la integración entre sus diferentes componentes, lo que posibilita obtener una base para la creación de comunidades virtuales. Además, permite la gestión de módulos, secciones y posiciones en un sitio web, por lo que el sistema que se desarrollará se basará en la gestión de secciones o contenidos y módulos de la plataforma web Medicando.

## 1.6. Herramientas y tecnologías

### 1.6.1. Metodologías de desarrollo

La metodología de desarrollo es un proceso de software detallado y completo. Adicionalmente una metodología define con precisión los artefactos, roles y actividades, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto y guías para el uso de herramientas de apoyo. Para el desarrollo del sistema propuesto se realiza un estudio de las metodologías ágiles y tradicionales, las mismas proponen una estrecha relación entre el cliente y el equipo de desarrollo, no existe un contrato tradicional, se aplican a proyectos con requerimientos cambiantes o imprecisos, y están dirigidas fundamentalmente para equipos pequeños; se realizan entregas funcionales continuamente y el cliente es parte del equipo de desarrollo [14].

#### Programación Extrema

Metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Programación Extrema (XP) se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente ade-

cuada para proyectos con requisitos imprecisos y cambiantes, y donde existe un alto riesgo técnico.

**XP** utiliza un enfoque orientado a objetos como paradigma preferido de desarrollo, y engloba un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades estructurales: planeación, diseño, codificación y pruebas. Esta metodología tiene como base la simplicidad y como objetivo principal la satisfacción del cliente; para lograrlo se deben tomar en cuenta cuatro valores fundamentales: Comunicación, Simplicidad, Retroalimentación y Coraje [15]:

- Planificación.
- Diseño.
- Desarrollo.
- Pruebas.

**Herramientas de la XP:**

- Historias de usuario.
- Casos de prueba de aceptación.
- Tareas de ingeniería.
- Tarjetas CRC (Clase-Responsabilidad-Colaboración).

**Ventajas y desventajas** Una de las ventajas de XP es que se adapta al desarrollo de sistemas pequeños y grandes; optimiza el tiempo de desarrollo; y la realización en pareja para complementar los conocimientos; el código es sencillo y entendible, además de la poca documentación a elaborar para el desarrollo del sistema. Las desventajas son que no se tiene la definición del costo y el tiempo de desarrollo; el sistema va creciendo después de cada entrega al cliente y nadie puede decir que el cliente no querrá una función más; se necesita de la presencia constante del usuario, lo cual en la realidad es muy difícil de lograr.

**SCRUM**

La metodología Scrum para el desarrollo ágil de software es un marco de trabajo diseñado para lograr la colaboración eficaz de equipos en proyectos, que emplea un conjunto de reglas y artefactos y define roles que generan la estructura necesaria para su correcto funcionamiento. Scrum utiliza un enfoque incremental que tiene como fundamento la teoría de control empírico de procesos. Esta teoría se fundamenta en transparencia, inspección y adaptación; la transparencia, que garantiza la visibilidad en el proceso de las cosas que pueden afectar el resultado; la inspección, que ayuda a detectar variaciones indeseables en el proceso; y la adaptación, que realiza los ajustes pertinentes para minimizar el impacto de las mismas [16].

**(RUP)**

RUP define la gestión de proyectos de software como el arte de balancear los objetivos, administrar riesgos y superar restricciones para entregar un producto que cumpla con las necesidades de los clientes y usuarios. La disciplina de gestión de proyecto de RUP provee: un marco para la gestión de proyectos orientados al desarrollo de software; guías prácticas para la planificación, dirección de personal, ejecución, monitoreo y supervisión de proyectos; además, provee un marco para la gestión de riesgos. Esta disciplina



está enfocada principalmente en los aspectos más importantes de un proceso de desarrollo iterativo: gestionar los riesgos; planificar un proyecto iterativo, en todo su ciclo de vida y para cada una de las iteraciones en particular; supervisar el progreso de un proyecto iterativo, y sus métricas.

### Selección de la metodología a emplear en la solución propuesta

Tabla 1.4. Comparación entre metodologías de desarrollo. (Fuente: Elaboración Propia).

	Tiempo de desarrollo	Tamaño mínimo del equipo	Dinamismo	Nivel de experiencia	Participación del cliente	Artefactos Ingenieriles
<b>XP</b>	1-3 meses	2	Sugeto a cambio	Bajo	Si	Pocos
<b>RUP</b>	9 meses	6-7	No resiste el cambio	Alta	No constante	Muchos
<b>SCRUM</b>	1-3 meses	3-4	Sugeto a cambio	Medio	Si	Casi Ninguno

Después de realizado el estudio y análisis de estas metodologías de desarrollo, se ha llegado a la conclusión de utilizar XP, ya que se adaptada al software a desarrollar, así como a las condiciones de trabajo, de forma general. También es una metodología que se implanta en proyectos con poca duración, donde su ciclo de vida es corto.

Propone que la comunicación y satisfacción del cliente es lo principal. No se hace mucho énfasis en la documentación, sólo es más importante definir los requerimientos y las pruebas de calidad.

Es importante destacar de esta metodología, que dada la poca experiencia que tiene el cliente en cuanto a Portales Web, los requerimientos actuales pueden estar ajustados a futuros cambios, este es un punto donde la metodología es flexible, ya que permite administrar estos cambios de forma óptima.

Además, uno de los objetivos de importancia que aplica esta metodología, es potenciar al máximo el trabajo en grupo, donde los jefes de proyecto, los clientes y desarrolladores son parte del equipo y están involucrados en el desarrollo del software. A pesar de las semejanzas entre las metodologías en la tabla anterior se demuestra que XP tiene mayor agilidad.

### 1.6.2. Lenguaje de modelado

Como lenguaje de modelado se utilizará [Lenguaje de modelado unificado \(UML, por sus siglas en inglés\)](#). UML es un estándar OMG3 diseñado para visualizar, especificar, construir y documentar software orientado a objetos. Es una simplicidad de la realidad. El trabajo de UML es esencial porque con él se logra establecer un listado de requerimientos y estructuras necesarias para idear un sistema antes del proceso de implementación. Además, en este trayecto se logra comprender mejor lo que se está construyendo y en algunos casos se describen las oportunidades de simplificación y reutilización.

### 1.6.3. Herramientas de modelado

Para el modelado de la propuesta de solución se trabajará con la herramienta CASE Visual Paradigm (VP). Herramienta multiplataforma de gran alcance y fácil de utilizar, además propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Es de código abierto

y forma parte de las políticas del país de utilizar herramientas libres, ya que no se deben pagar licencias comerciales.

Visual Paradigm aporta un gran apoyo en cuanto a la generación de diagramas UML como es el caso de los diagramas de clases, diagramas de casos de uso del sistema y diagramas de secuencias. Por otro lado ayuda en el proceso de captura de requisitos con el diagrama de requisitos, en el modelado de bases de datos con los diagramas de entidad-relación, diagramas ORM4 y por último en las matrices de trazabilidad.

#### 1.6.4. Servidores Web

Un servidor es el programa que debe estar ejecutándose en el equipo servidor para que pueda ofrecer su servicio y responder a peticiones de usuarios remotos. En el se almacenan los documentos HTML para que la página web sea accesible a usuarios externos [17].

##### Paquete XAMPP

XAMPP es un servidor independiente multiplataforma, de software libre, que consiste principalmente en la base de datos MySQL, el servidor web APACHE y los interpretes para los lenguajes de script: PHP y PERL estas herramientas ayudan para que la aplicación se maneje en la web o en una máquina de escritorio. Permite instalar de forma sencilla Apache en el ordenador, sin importar el sistema operativo (Linux, Windows, MAC o Solaris). Incluye PhpMyAdmin para administrar las bases de datos de MySQL [18].

XAMPP, permite la instalación de complementos como WordPress, Drupal, Joomla!, entre otras; lo cual con WAMP no es posible. Esta herramienta se encuentra contenida en las principales herramientas que se utilizaron para la creación de la plataforma web Medicando.

#### 1.6.5. Marco de trabajo

Un framework es un conjunto de clases cooperativas que construyen un diseño reutilizable para un tipo específico de software. Un Framework proporciona la arquitectura partiendo el diseño en clases abstractas y definiendo sus responsabilidades y colaboraciones. Un desarrollador realiza una aplicación haciendo subclases y componiendo instancias a partir de las clases definidas por el Framework.

Un framework es una aplicación semi-completa, de control invertido, por lo que un Framework difiere de una librería de clases: En una librería de clases, el control del flujo se encuentra en el código de la aplicación que realiza llamadas a los métodos de la librería de clases.

En un framework, el control del flujo está en código del framework que realiza llamadas al código de la aplicación [19].

##### Marco de trabajo Yii

Yii es un framework PHP de alto rendimiento, basado en componentes para la implementación de sitios web a gran escala. Permite reutilización máxima en la programación Web y puede acelerar significativamente el proceso de desarrollo de aplicaciones. Es ligero y equipado, con sofisticados mecanismos de caché, es especialmente adecuado para aplicaciones de alto tráfico, tales como Portales, foros, sistemas de comercio electrónico, entre otros.

Yii sobresale a otros frameworks PHP al ser eficiente, rico en características y claramente documentado. Es cuidadosamente diseñado desde cero para estar en forma para el desarrollo serio de una aplicación web. No es ni un subproducto de algún proyecto ni un conglomerado de trabajo de terceros. Es el resultado de la rica experiencia de los autores con el desarrollo de aplicaciones Web y su investigación de los marcos de programación web y las aplicaciones más populares [20].

### Principales características

- Patrón de diseño MVC.
- Objetos de accesos a la Base de Datos (DAO), generador de consultas (query builder), guardado activo (Active Record) y migración de base de datos.
- Integración con jQuery.
- Entradas de formulario y validación.
- Reproductores (Widgets) de Ajax, como autocompletado de campos de texto y demás.
- Soporte de autenticación incorporado. Además soporta autorización vía *role-based access control* (RBAC del español Control de acceso basado en roles) jerárquico.
- Personalización de aspectos y temas.
- Generación compleja automática de WSDL, especificaciones y administración de peticiones Web service.
- Internacionalización y localización (I18N and L10N). Soporta traducciones, formato de fecha y hora, formato de números, y localización de la vista.
- Esquema de *caching* por capas. Soporta el cache de datos, cache de páginas, cache por fragmentos y contenido dinámico. El medio de almacenamiento del cache puede ser cambiado.
- El manejo de errores y *logging*. Los errores son manejados y personalizados, y los log de mensajes pueden ser categorizados, filtrados y movidos a diferentes destinos.
- Las medidas de seguridad incluyen la prevención *cross-site scripting* (XSS), prevención *cross-site request forgery* (CSRF), prevención de la manipulación de *cookies*.
- Herramientas para pruebas unitarias y funcionales basados en PHPUnit y Selenium.
- Generación automática de código para el esqueleto de la aplicación, de funcionalidades como Crear, Obtener, Actualizar y Borrar, por sus siglas en inglés (CRUD).
- Generación de código por componentes de Yii y la herramienta por línea de comandos cumple con los estándares de XHTML.
- Cuidadosamente diseñado para trabajar bien con código de terceros. Por ejemplo, es posible usar el código de PHP o Zend Framework en una aplicación Yii.

El marco de trabajo que se utilizará es Yii, debido a que este es un framework genérico de programación web que puede ser utilizado para todo tipo de aplicaciones Web. Es liviano de ejecutar y está equipado con soluciones de cacheo sofisticadas, por lo que es adecuado para desarrollar foros, CMS, Sistemas de comercio electrónico (e-commerce) y aplicaciones de gran tráfico como Portales [20], además Yii v2.0 es utilizado por el centro Vertex para la realización de la plataforma web Medicando.

Yii sobresale frente a frameworks (PHP) en su eficiencia, su gran cantidad de características y su clara documentación. La curva de aprendizaje es baja, lo que hace posible que en poco tiempo pueda ser dominado. Es satisfactorio para empezar una aplicación desde 0 ya que con sólo una tabla en la base de datos es capaz de generar un sitio con las operaciones CRUD ya implementadas y sin escribir una sola línea de código, esto sin dudas reduce el tiempo de trabajo, brindando la posibilidad de hacer más en menos tiempo. Está basado en [Modelo Vista Controlador \(MVC\)](#), para acceso a Base de datos.

### 1.6.6. Lenguaje de programación

**PHP-Hypertext Pre-Processor** Es un lenguaje de programación multiplataforma de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. La última versión estable es la 5.5.12.

En la actualidad es ampliamente usado en entornos de desarrollo web por su facilidad de uso, su integración perfecta con HTML y su versatilidad de uso en diferentes Sistemas Operativos. Tanto es su expansión, que se calcula su uso en torno a más de 20 millones de sitios web y un millón de servidores en todo el mundo.

### 1.6.7. Gestores de bases de datos

Un gestor de base de datos es un software que permite introducir, organizar y recuperar la información de las bases de datos, así como administrarlas. Existen distintos tipos de gestores de bases de datos: relacional, jerárquico, entre otros.

**MySQL** Es un SGBD relacional multiplataforma. Posee un sistema de privilegios y contraseñas que es muy flexible y seguro. Soporta grandes bases de datos. MySQL tiene soporte para comandos SQL para chequear, optimizar y reparar tablas. Es muy rápido, seguro y fácil de usar, esto significa que es un servidor bastante apropiado para acceder a bases de datos en internet. Presenta características como son: conectividad segura, transacciones y claves foráneas, replicación, búsqueda de indexación de campos de texto y disponibilidad en gran cantidad de plataformas y sistemas.

### 1.6.8. Entornos de Desarrollo Integrado

El [Entorno de desarrollo integrado \(IDE, por sus siglas en inglés\)](#), es una aplicación compuesta por un conjunto de herramientas útiles para un desarrollador. Los IDE están diseñados para maximizar la productividad del programador proporcionando componentes muy unidos con interfaces de usuario similares. Los IDEs presentan un único programa en el que se lleva a cabo todo el desarrollo. Generalmente, este programa suele ofrecer muchas características para la creación, modificación, compilación, implementación y depuración de software. Puede ser exclusivo para un lenguaje de programación o utilizarse para varios. Está

compuesto por un editor de código (con facilidades como resaltado de sintáxis, completamiento de código y navegación entre clases), un compilador y herramientas de automatización de la compilación, un depurador y en algunos casos un constructor de interfaz gráfica [21].

### 1.6.9. Selección del entorno de desarrollo

Se utilizará PhpStorm como entorno de desarrollo integrado para los desarrolladores de PHP, el mismo hereda toda la funcionalidad relacionada con la web de IntelliJ IDEA para editar PHP, HTML, CSS, JavaScript, XML, trabajando con VCS, SQL, además agrega soporte avanzado para las otras herramientas específicas para el desarrollo web. PhpStorm le ofrece las siguientes funciones avanzadas:

- Editor Inteligente: para PHP, HTML, CSS, JavaScript y XML, que incluye sintaxis resaltado, búsqueda de documentación y refactorizaciones.
- Codificación sin errores: análisis de código sobre la marcha, resaltado de errores y soluciones rápidas.
- Navegación de proyecto y código: vistas de proyecto especializadas, vistas de estructura de archivos y vistas rápidas saltando entre archivos, clases, métodos y usos.
- Integraciones VCS: soporte listo para usar para Subversion, Perforce, Git y CVS con listas de cambio y fusionar.
- Sincronización FTP: actualice el servidor usando FTP o SFTP.
- Soporte SQL: asistencia de codificación, consola SQL y navegador de base de datos.
- JetBrains PhpStorm es multiplataforma. Funciona en Windows, OS X y Linux y trae el todo el rango de precisión herramientas de desarrollo, todo unido para crear un desarrollo conveniente.

### 1.6.10. Otras herramientas utilizadas

#### *Jquery*

jQuery es un framework Javascript, producto que sirve como base para la programación avanzada de aplicaciones, que aporta una serie de funciones o códigos para realizar tareas habituales, este producto tiene gran aceptación por parte de los programadores muy buena y un grado de penetración en el mercado muy amplio, lo que hace suponer que es una de las mejores opciones. Además, es un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del framework. Contiene gran cantidad de plugins o componentes, lo que hace fácil encontrar soluciones ya creadas en jQuery para implementar asuntos como interfaces de usuario, galerías, votaciones, efectos diversos, etc [22].

#### *Bootstrap*

*Bootstrap*, es un framework originalmente creado por Twitter, que permite crear interfaces Web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio Web al tamaño del dispositivo en que se visualice. Esta técnica de diseño y desarrollo se conoce como *responsive design* o diseño adaptativo. Es compatible con todos los navegadores, y cualquier persona puede comenzar con Bootstrap.

El beneficio de usar responsive design en un sitio Web, es principalmente que el sitio Web se adapta automáticamente al dispositivo desde donde se acceda. Lo que se usa con más frecuencia es el uso de Media Queries (consultas), que es un módulo de CSS3 que permite la representación de contenido para adaptarse a condiciones como la resolución de la pantalla y si trabajas las dimensiones de tu contenido en porcentajes, puedes tener una Web muy fluida capaz de adaptarse a casi cualquier tamaño de forma automática [23].

## 1.7. Conclusiones del capítulo

- El estudio de los sistemas homólogos permitió identificar las características comunes asociadas a los portales médicos que sirven de base para conformar la propuesta de solución.
- De la revisión de los CMS estudiados se obtuvieron elementos arquitectónicos que se permitieron ampliar las funcionalidades de la propuesta a desarrollar, además de definir la arquitectura que presentará el sitio mediante los gestores de contenidos Joomla, Wordpress y Drupal.
- A partir de las revisiones realizadas sobre las herramientas y tecnologías se adopta:
  - La selección del marco de trabajo Yii en su versión 2.0 permite una construcción estructurada, escalable y extensible de aplicaciones web, optimizando el empleo de lenguajes como HTML5, CSS3, JavaScript y PHP v5, implementándose sobre el IDE Netbeans 8.0 para minimizar el tiempo de escritura y depuración de códigos. Además, como complemento para garantizar el correcto funcionamiento del framework, se selecciona el paquete de programas XAMPP, que incluye el Apache Server como servidor web, PHP como lenguaje de programación y MySQL como SGBD.
  - La selección de XP como metodología de desarrollo garantiza la generación de artefactos que son necesarios para una mayor comprensión de la solución a desarrollar.

## 2.1. Introducción del capítulo

La concepción y el diseño de una solución informática parte de la aplicación de buenas prácticas de ingeniería de software. En este sentido, teniendo en cuenta los principales artefactos ingenieriles propuestos por la metodología XP se guían las características funcionales y de diseño de la solución propuesta. Además, se describen las [Historia de usuarios \(HU\)](#) distribuyéndolas en dos iteraciones y se confecciona su plan de entrega teniendo en cuenta sus puntos estimados. También se describe y estructura la propuesta de solución mediante el patrón arquitectónico Modelo Vista Controlador y sus tarjetas CRC.

## 2.2. Propuesta del sistema

Teniendo en cuenta la situación problemática, el problema de la investigación y el objetivo general, como propuesta de solución se consideró desarrollar un sistema informático que permita la administración de los contenidos que serán visualizados desde la plataforma web Medicando. El propósito de esta solución informática es estructurar y organizar la información con la que interactuarán tantos los pacientes, especialistas en rehabilitación como otros usuarios que actúen como invitados. Se puede decir que cada uno de los componentes de la aplicación contienen el patrón MVC, los usuarios tienen navegación plena por el sitio; mientras que los administradores son los encargados de monitorear las opciones individuales que ofrece el mismo. Al sistema solo podrá acceder el personal autorizado, de acuerdo a la implementación de un sistema de autenticación basado en roles. La solución estará compuesta de manera general por las siguientes funcionalidades:

- **Gestión secciones:** Encargada de renderizar [24] a cada una de las posiciones (contenedores), el contenido referente a estas y mostrarlo en la página según el orden de prioridad.
- **Gestión de menú:** Responderá por la necesidad de informar a los usuarios del sistema mediante los lugares de navegación del sitio mediante el menú principal y de esta forma dinamizar el sitio.

- **Gestión de elementos de menú:** Encargada de la generación de los elementos del menú, los cuáles permitirán usabilidad y fiabilidad del sistema. Además, permite a cada elemento del menú renderizar el contenido o secciones mediante la dirección url (enlace).
- **Gestión de FAQs:** Permite crear, eliminar, modificar, visualizar y listar las preguntas asociadas a una temática determinada.

La plataforma se implementó siguiendo los principios del patrón arquitectónico MVC, la propuesta de solución estará basada de igual forma en dicho patrón. Medicando cuenta con los paquetes funcionales orientados a la gestión de Doctores, Pacientes, Tratamientos, Videojuegos, Noticias, FAQs, Servicios, Contactos, Componentes, así como administración de menú, módulos, configuración de secciones y posiciones orientados a la visualización del sistema de manera dinámica.

Se define además, una propuesta de plantilla basada en espacios reservados en el diseño del sitio web, donde a cada uno de estos se renderizan cada una de las secciones definidas. A partir de la propuesta de plantilla se establecen tres posiciones específicas que se ubican en la Figura 2.1. Donde se define un procedimiento para determinar por cada espacio, los contenidos o secciones que se encuentren activos.

De esta forma en la sección superior se ubicará siempre el menú principal del sistema que de acuerdo a sus características se gestiona de forma independiente a los contenidos, lo que permite asociar a cada elemento del menú las secciones correspondientes; o sea, determinar en cada vista que secciones estarán visibles. A continuación se muestra un prototipo referente al sitio en el cual se observa los elementos del menú y las secciones que deben estar visibles para el mismo:

**BANNER:** Esta posición se encuentra en el límite superior de la plantilla, en ella se visualizará el menú de navegación.

**CONTAINER:** Presentan el carrusel que contiene dicho sitio; además de las secciones o contenidos referentes a él, los cuáles se visualizarán de forma dinámica.

**FOOTER:** Se visualizará los diferentes sitios en internet a los que pueden acceder los usuarios, además de los últimos videojuegos subidos al sitio.

En las interfaces de comunicación diseñadas se utilizaron los colores: verde claro y oscuro en degradado, así como el blanco. La tipografía utilizada para visualizar las informaciones pertinentes es Arial 12 con un interlineado de 1.5.

El sitio presenta el diseño web adaptativo (*responsive*), el cuál permite ajustar el contenido del mismo al ancho del área de despliegue de cada dispositivo, por lo que permite que se visualice en distintas modalidades. El diseño de la información es variable y relativo; y la composición se presenta en forma dinámica, puesto que cada elemento se ajusta a las condiciones técnicas de despliegue.

### 2.3. Fase I: Planificación

Es la fase en la que se define el alcance general del proyecto. En esta el cliente define lo que necesita mediante la redacción de HU y establece la prioridad de cada una. Luego, los programadores estiman los



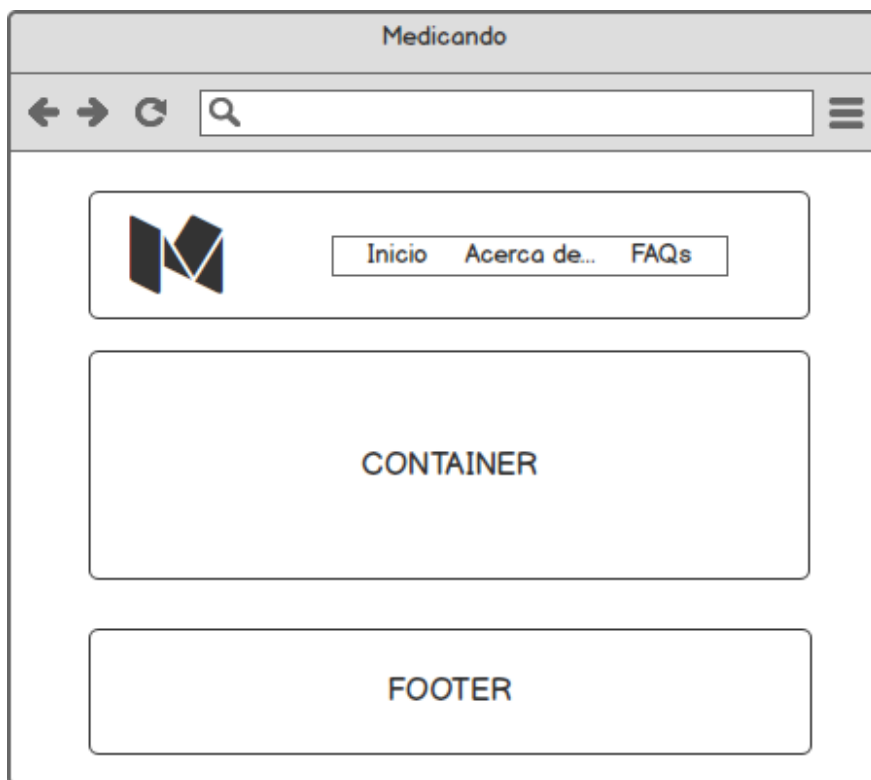


Figura 2.1. Prototipo de diseño del sitio. (Fuente: Elaboración propia).

tiempos de desarrollo en base a esta información. Las estimaciones realizadas en esta fase son primarias, debido a que estarán basadas en datos de muy alto nivel y podrían variar cuando se analicen en cada iteración. Además se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses [25, 26].

## 2.4. Historias de Usuario

Las **HU** son la técnica que utiliza **XP** para especificar los requisitos de software, estas deben ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a las tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra **HU**. Las estimaciones de esfuerzo, asociado a la implementación de las historias, la establecen los programadores, utilizando como medida el punto. Un punto, equivale a una semana ideal de programación (5 días laborables). Las historias, generalmente, valen de 1 a 3 puntos [25]. A continuación, se describen algunas de las **HU** definidas para llevar a cabo el desarrollo del sistema, el resto pueden ser consultadas en los anexos de la investigación.



Figura 2.2. Representación de los elementos de menú del sitio y sus secciones correspondientes

Tabla 2.1. Historia de usuario # 1

Historia de usuario	
<b>Número:</b> 1	<b>Nombre:</b> Administrar menú.
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 2.5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Solanch de la C. Ríos Portillo	
<b>Descripción:</b> Permite al administrador crear, modificar, eliminar y mostrar menú que esté activo: <ul style="list-style-type: none"> <li>• Id</li> <li>• Nombre</li> <li>• Estado</li> <li>• Fecha de creado</li> <li>• Fecha de actualizado</li> </ul>	
<b>Observaciones:</b> Al acceder a la acción de crear menú se muestra una opción de retorno.	

Continúa en la próxima página

Tabla 2.1. Continuación de la página anterior

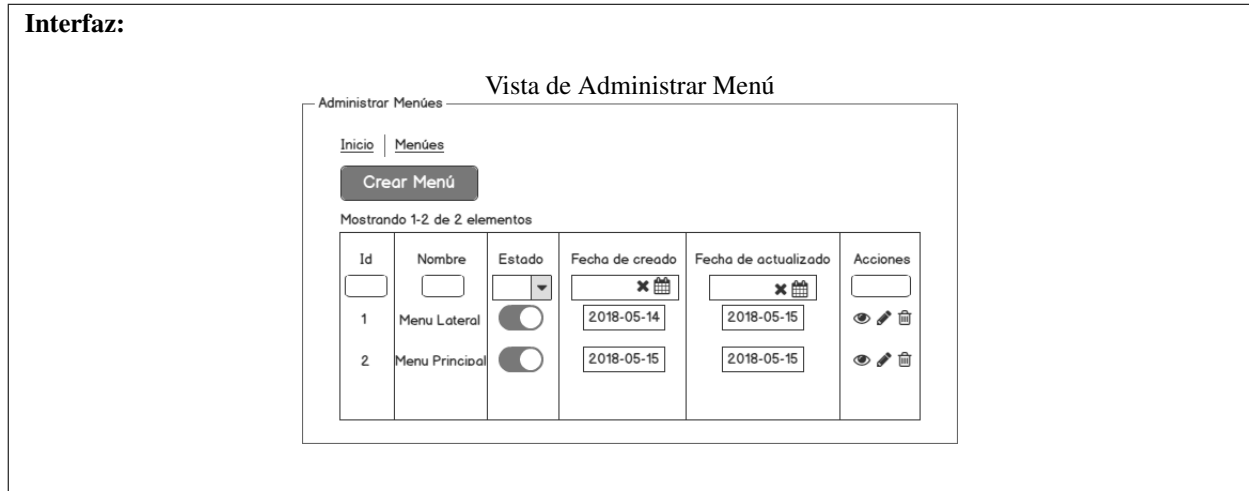
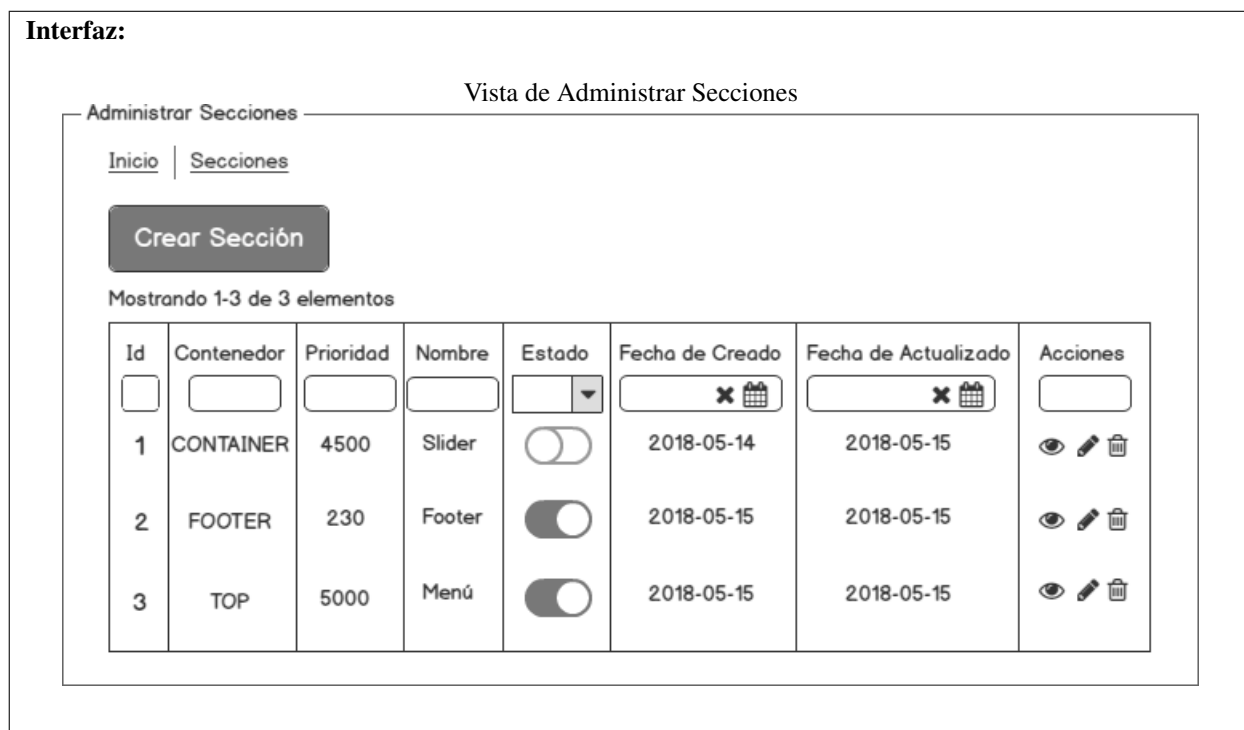


Tabla 2.2. Historia de usuario # 2

Historia de usuario	
<b>Número:</b> 2	<b>Nombre:</b> Administrar secciones.
<b>Usuario:</b> Paciente	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 3.0	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Solanch de la C. Ríos Portillo	
<b>Descripción:</b> Permite al administrador crear, modificar, eliminar y mostrar las secciones que contendrá el sitio: <ul style="list-style-type: none"> <li>• Id</li> <li>• Nombre</li> <li>• Posiciones</li> <li>• Estado</li> </ul>	
<b>Observaciones:</b> Al acceder a la acción de crear sección se muestra una opción de retorno.	

Continúa en la próxima página

Tabla 2.2. Continuación de la página anterior



## 2.5. Requisitos no funcionales

Después de analizar las funcionalidades que el perfil de instalación debe cumplir, es necesario analizar las propiedades o características que el mismo debe tener para un mejor funcionamiento, a esto se le conoce como requisitos no funcionales. Entre los requisitos no funcionales necesarios para el perfil de instalación del sitio Medicando se encuentran los siguientes:

### Requisitos no funcionales (RnF) 1: Usabilidad

- **RnF 1.1** La aplicación muestra la arquitectura de información referente a cada uno de los CMS estudiados (Joomla, Drupal y WordPress).

### RnF 2: Apariencia o interfaz externa

- **RnF 1.2** Los colores que debe contener el sitio en las interfaces de comunicación deben ser verde claro, blanco y verde oscuro; debido a que se basa en un portal que está enfocado a la rehabilitación de pacientes.
- **RnF 1.3** La tipografía utilizada para visualizar las informaciones pertinentes es Arial 12 con un interlineado de 1.5, para que la letra sea visualizada con facilidad.

### RnF 2: Restricciones del diseño y la implementación

- **RnF 2.1** Se utilizará como lenguaje de programación PHP en su versión 5.2.5.

- **RnF 2.2** Se empleará como gestor de bases de datos MySQL en su versión 5.6.16.
- **RnF 2.3** Como framework de desarrollo Yii en su versión 2.0.
- **RnF 2.4** Visual Paradigm como herramienta de diseño para la representación arquitectónica del sistema.
- **RnF 2.5** Balsamiq Mockups como herramienta de diseño de los prototipos de interfaz de la solución, en su versión 3.0.

### RnF 3: Software

- **RnF 3.1** El sistema puede ser visualizado mediante los navegadores Chrome en su versión 41.0.3396.79, Firefox en su versión 40.0.2 (Quantum) y Opera en su versión 32.0.2907.88, sobre las que se ejecutará a partir de estas.

### RnF 4: Hardware

- **Cliente**
  - **RnF 4.1** Se podrá ejecutar en un MicroProcesador i3-4160 o superior.
  - **RnF 4.2** Se requiere 160 Giga byte o superior de disco duro.
  - **RnF 4.3** Se deberá contar con 2 Giga byte de memoria RAM.
- **Servidor**
  - **RnF 4.4** Se podrá ejecutar en un MicroProcesador i3-4170 o superior.
  - **RnF 4.5** Se requiere 1T de disco duro.
  - **RnF 4.6** Se deberá contar con 4 Giga byte de memoria RAM.

### RnF 5: Rendimiento

- **RnF 5.1** El sistema en la funcionalidad Visualizar secciones debe tardar 10ms.

## 2.5.1. Estimación de esfuerzo por Historia de Usuario

Se realiza la estimación de esfuerzo que arroja cada **HU**, con el objetivo de obtener un correcto desarrollo del sistema. Para una mayor organización se decide además, asignar a cada iteración el conjunto de historias agrupadas en correspondencia con el módulo al que representen. A continuación, se muestra la estimación realizada:

## 2.5.2. Desarrollo del Plan de Iteraciones

Una vez definidas las **HU** y realizada una previa estimación de esfuerzos, se procede a la planificación de la etapa de implementación del sistema. En este espacio, se crea el plan de iteraciones, donde se especifica la prioridad con que se implementarán las **HU** organizadas por iteraciones. Teniendo en cuenta el esfuerzo asociado a las **HU** y a las prioridades del cliente, se define una versión que sea de valor para este.

Tabla 2.3. Estimación de esfuerzo por Historia de Usuario. (Fuente: Elaboración propia).

Historias de Usuario		Puntos estimados(semanas)
1	Administrar Menú	2.5
2	Configurar Elementos de Menú	3.0
3	Administrar FAQs	2.5
4	Administrar Secciones	3.0
5	Autenticar Usuario	1.5

### 2.5.3. Plan de duración de las Iteraciones

A continuación, se presenta el plan de duración de las iteraciones. Este plan, tiene como finalidad, mostrar la duración de cada iteración, así como el orden en que serán implementadas las **HU** en cada iteración como se muestra en la tabla siguiente:

Tabla 2.4. Plan de duración de las iteraciones. (Fuente: Elaboración propia).

Iteración	Historias de Usuario		Duración (semanas)
1	1	Administrar Menú	2.5
	2	Configurar Elementos de Menú	3.0
	3	Administrar FAQs	2.5
2	1	Administrar Secciones	3.0
	2	Autenticar Usuario	1.5
<b>Total</b>			<b>12.5</b>

### 2.5.4. Plan de Entrega

En el plan de entrega que se plantea a continuación, se hace una propuesta de las versiones (releases) del sistema. Cada versión se conformará al finalizar una iteración.

Tabla 2.5. Plan de entrega de versiones. (Fuente: Elaboración propia).

Entregable	Fecha de Inicio	Fecha de Fin
Iteración 1	5/2/2018	30/3/2018
Iteración 2	23/4/2018	28/5/2018

## 2.6. Roles del sistema

Uno de los primeros pasos a tener en cuenta ante la realización de un sistema informático, es a la población al cual va dirigido, teniendo en cuenta que esta puede estar dividida por los niveles de privilegios

que se le otorgue a cada usuario, dependiendo del rol que desempeñe. Los usuarios podrán navegar en la página inicial antes de estar autenticados, los cuales no tendrán acceso a ninguna funcionalidad, solo a noticias. A continuación, se muestra un listado con los roles de usuarios definidos para esta aplicación:

Tabla 2.6. Roles de usuario. (Fuente: Elaboración propia).

Roles	Descripción
<b>Administrador</b>	Encargado de gestionar toda la información tanto para los usuarios autenticados como para los invitados. Establece los permisos pertinentes para los distintos tipos de usuarios.
<b>Médico</b>	Tiene acceso a todo lo visible en el sitio. Es el encargado de asignar los ejercicios correspondientes a la etapa en la que se encuentre el paciente y actualizar los contenidos en el sistema.
<b>Paciente</b>	Tiene acceso a todo lo público en el sitio. Además puede jugar todos los videojuegos que el mismo contiene
<b>Invitado</b>	Usuario que navega por la plataforma sin autenticarse. El nivel de visualización de la información a la que puede acceder es definido por el Administrador.

## 2.7. Fase II: Diseño del sistema

Para lograr una arquitectura robusta en la confección de un sistema, es una buena práctica mecanismos que validen de forma teórico-práctico el buen desempeño del mismo. El framework seleccionado implementa el patrón de arquitectura Modelo Vista Controlador (MVC), el cual separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos definidos por sus nombres modelo, vista y control. Este se ve frecuentemente en aplicaciones web, donde la vista representa la página HTML, la cual maneja la visualización de la información y el código que provee de datos dinámicos a la página. El modelo es el sistema de gestión de base de datos (administra el comportamiento y los datos del dominio de aplicación) y la lógica de negocio o controlador es el responsable de recibir los eventos de entrada desde la vista. Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. A continuación, se muestra una representación del patrón MVC en una de las funcionalidades perteneciente a la propuesta de solución, correspondiente a la administración de secciones: Figura 2.3.

Representación arquitectónica del sistema.

**Vista:** Esta capa contiene las páginas HTML o vistas que son las que se encargan de visualizar la información. (Ejemplo: admin.php, create.php, update.php, delete.php, view.php)

**Modelo:** Esta capa se encarga de recibir los eventos de entrada desde la vista o capa de presentación. (Ejemplo: Section.php)

**Controlador:** Esta capa trabaja mediante un sistema de gestión de base de datos, administrando, controlando el comportamiento y dominio de aplicación, así como la lógica de negocio. (Ejemplo: SectionController.php)

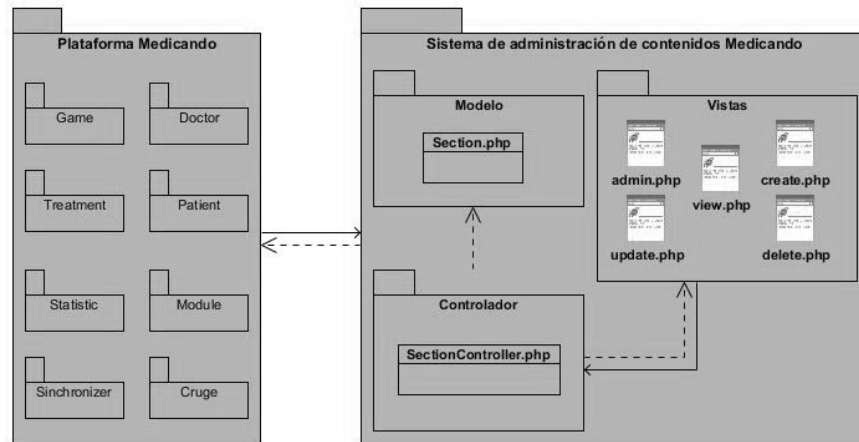


Figura 2.3

Cada clase o tarjeta CRC de la solución, se encuentra ubicada en la estructura correspondiente según funciones descritas anteriormente para cada parte que conforma al patrón MVC.



### 2.7.1. Patrones de diseño

Para obtener una correcta implementación se hace necesario tener en cuenta las mejores soluciones conocidas, con el objetivo de diseñar las clases y las funcionalidades a codificar. Para ello se analizan y proponen algunos patrones de diseño que responden ante las necesidades de desarrollo.

Los patrones de diseño, tratan los problemas que se repiten y que se presentan en situaciones particulares del diseño, con el fin de proponer soluciones a ellas. Se encargan de identificar clases, instancias, roles, colaboraciones entre estas, así como la distribución de responsabilidades [27]. En resumen, es una descripción de clases y objetos comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular. Para la asignación general de responsabilidades en el software existen patrones denominados **Patrones Generales de Asignación de Responsabilidades (GRASP, por sus siglas en inglés)**, que describen los principios fundamentales de asignación de responsabilidades a objetos, expresadas como patrones. Seguidamente se exponen algunos patrones utilizados dentro del contexto de los módulos a implementar:

- **Experto:** permite mantener el encapsulamiento de la información, puesto que los objetos utilizan su propia información para llevar a cabo las tareas asociadas a este. Por ejemplo, la clase *Section* es la encargada de proporcionar toda la información relacionada con las posiciones y las secciones que se generan en el sistema, debido a que es la que representa todos los atributos para la creación de este objeto.
- **Creador:** se evidencia en la clase controladora *SectionController*. Esta clase se encarga de modificar la configuración de las secciones de cada usuario, se ofrece así una alta cohesión. Permitiendo, de esta forma, establecer un bajo acoplamiento, lo que propicia menos dependencias funcionales y ofrece mejores oportunidades de reutilización.
- **Controlador:** este patrón se evidencia en las clases que actúan como intermediarias para el manejo de eventos. En el sistema se puede identificar en la clase *SectionController*, esta se encarga de listar y eliminar secciones creadas a partir de la plataforma.
- **Alta Cohesión:** Verifica que la información que almacena una clase debe ser coherente y debe estar, en la medida de lo posible, relacionada con la clase. En la solución propuesta las clases clientes y las servidoras, se encargan cada una de realizar una función específica, permitiendo que las mismas no se saturen y a través de la asignación de responsabilidades se garantiza la alta cohesión en las clases existentes.
- **Bajo Acoplamiento:** El acoplamiento mide el grado en que una clase está conectada a otra, tiene conocimiento de otra o, de alguna manera, depende de otra. En el desarrollo de los componentes se ve reflejado a la hora de producirse modificaciones en alguna de sus clases, ya que las mismas no están fuertemente relacionadas, por tanto, la repercusión sobre las demás será mínima.

Por su parte, el framework Yii, implementa el patrón de persistencia *ActiveRecord*, que facilita el control de la información almacenada en la base de datos. Para ello utiliza el mapeo **Object-Relational Mapping**

(ORM, por sus siglas en inglés). Se trata de una clase que se encarga de implementar todas las operaciones de consulta y modificación de una tabla concreta de la base de datos. De esta forma, la aplicación delega el trabajo con SQL, a la capa de componentes *ActiveRecord* que maneja Yii. Las funcionalidades que se ofrecen, están orientadas principalmente a las propiedades de las columnas de la tabla representada. Una vez definidos los patrones de diseño, se describen las clases utilizadas en la solución. Siguiendo la metodología XP, se analizan las HU y se descomponen en tareas independientes.

A continuación, se detallan algunos de los patrones de la "pandilla de los cuatro" (**Gang - of-Four**) que se utilizaron en el diseño:

- **Singleton:** Este patrón propone una sola instancia de la clase controladora para acceder a los datos y modificarlos, al igual que el framework Yii, que usa una sola instancia para manejar las sesiones y los *layout*.
- **Decorator:** Permite añadir funcionalidades dinámicamente a una clase con la creación de otra clase, sin necesidad de la herencia. En el framework Yii, los *layouts* decoran a las vistas, pues en un momento determinado les confieren propiedades dinámicas que no poseían.
- **Observer:** Define una dependencia de manera que cuando uno de los objetos cambia su estado, notifica este cambio a todos los dependientes. Se trata de un patrón de comportamiento, es decir, está relacionado con algoritmos de funcionamiento y asignación de responsabilidades a clases y objetos. Se evidencia en la implementación del patrón arquitectónico MVC definido en la aplicación.

## 2.8. Tarjetas CRC

A continuación, las HU son evaluadas para dividir las tareas, cada tarea representa una característica distinta del sistema y se puede diseñar una prueba de unidad que verifique cada tarea, estas tareas se representan por medio de las tarjetas CRC.

Figura 2.4. Prototipo de Tarjeta CRC. (Fuente: Elaboración propia).

Nombre de la clase	
Responsabilidades	Colaboradores

Como se puede observar en la Tabla 2.7, cada tarjeta contiene el nombre de la clase, una descripción de las responsabilidades o métodos asociados con la clase, así como la lista de las clases con que se relaciona o que colaboran con ella. A continuación, se describen las tarjetas definidas para la implementación de la solución:

### 2.8.1. Tarjetas CRC

Tabla 2.7. Tarjeta CRC # 1

Tarjeta CRC	
<b>Clase:</b> Section	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> <li>• <i>getContainer()</i> este método devuelve la posición a la que puede ser renderizadas las secciones activas.</li> <li>• <i>getMappedSections()</i> devuelve un mapa de secciones conteniendo el id y el nombre de las secciones.</li> </ul>	position

Tabla 2.8. Tarjeta CRC # 2

Tarjeta CRC	
<b>Clase:</b> SectionController	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> <li>• <i>actionStatus()</i> verifica el estado de la sección a mostrar.</li> <li>• <i>manageFiles()</i> permite el manejo de los ficheros php, los cuáles serán mostrados en las posiciones como secciones del sitio.</li> <li>• <i>actionDownload()</i> permite la descarga del fichero php, que se visualiza como sección en la página.</li> </ul>	section

Tabla 2.9. Tarjeta CRC # 3

Tarjeta CRC	
<b>Clase:</b> MenuItem	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> <li>• <i>getMappedMenuItems()</i> devuelve un mapa de secciones conteniendo el id y el nombre de los elementos de menú.</li> </ul>	section

Tabla 2.10. Tarjeta CRC # 4

Tarjeta CRC	
<b>Clase:</b> MenuItemController	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> <li>• <i>actionStatus()</i> verifica el estado del elemento a mostrar.</li> <li>• <i>manageFiles()</i> permite el manejo de los ficheros php, los cuáles serán mostrados en las posiciones como secciones del sitio, en el elemento del sitio que esté visualizando.</li> <li>• <i>actionDownload()</i> permite la descarga del fichero php, que se visualiza como sección en cada elemento de menú de la página.</li> </ul>	<p>menu_item</p>

## 2.9. Conclusiones del capítulo

En este capítulo se han abordado los aspectos referentes a la concepción del producto a desarrollar y sus características funcionales. Esto permitió concluir que:

- A partir de la definición de las HU se pudieron identificar las principales funcionalidades a desarrollar en correspondencia con la solución propuesta.
- La estimación del tiempo para la implementación de las HU definidas, permitió calcular una entrega final del producto en tres meses y tres semanas.
- Los patrones de diseño GRASP y GOF permitieron la correcta implementación de las funcionalidades, para la asignación general de responsabilidades y la implementación de todas las operaciones de consultas.
- Los elementos arquitectónicos y de diseño definidos, establecieron la vía para llevar a cabo la implementación y prueba del sistema. Además se adoptan los estándares de programación definidos por el equipo de desarrollo de la plataforma web Medicando.

### 3.1. Introducción del capítulo

En el presente capítulo se detallan las iteraciones realizadas durante la etapa de construcción de los módulos propuestos, además se exponen las tareas de ingeniería generadas para cada HU que fueron definidas, así como las pruebas de aceptación planificadas para el sistema. De esta forma es obtenido un producto funcional probado y listo para entregar al cliente al final de cada iteración como propone XP.

### 3.2. Fase III: Desarrollo

En esta fase, XP plantea que las HU seleccionadas para ser implementadas se realizan durante el transcurso de la iteración a la que pertenecen. Por estas razones, se lleva a cabo una revisión del plan de iteraciones y se modifican en caso de ser necesario. Como parte de este plan se descomponen las HU en tareas de ingeniería [26].

#### Tareas de ingeniería para la Iteración I

Para la primera iteración, se definieron un total de 7 tareas de ingeniería. Todas, desglosadas a partir de las HU correspondientes a Administrar menú. A continuación, se describen algunas de las tareas realizadas en esta iteración:

- HU1 Administrar menú

Tabla 3.1. Tarea de ingeniería # 1

Tarea	
Número de tarea: 1	Número de Historia de usuario: 1
Nombre de la tarea: Crear Menú	

Continúa en la próxima página

Tabla 3.1. Continuación de la página anterior

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 5 de febrero de 2018	<b>Fecha de fin:</b> 7 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite crear los diferentes tipos de menú del sitio mediante un formulario, donde los campos no pueden estar vacíos y de esta forma hacer la página dinámica.	

Tabla 3.2. Tarea de ingeniería # 2

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Actualizar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 8 de febrero de 2018	<b>Fecha de fin:</b> 12 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite actualizar los menú existentes en el sitio. Mostrará un formulario con los campos requeridos a modificar los cuáles son obligatorios, y mostrará el menú que se encuentre activo.	

Tabla 3.3. Tarea de ingeniería # 3

<b>Tarea</b>	
<b>Número de tarea:</b> 3	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Visualizar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 13 de febrero de 2018	<b>Fecha de fin:</b> 15 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite mostrar los campos del menú creado y verificar la correcta funcionalidad del mismo.	

Tabla 3.4. Tarea de ingeniería # 4

<b>Tarea</b>	
<b>Número de tarea:</b> 4	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Eliminar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 16 de febrero de 2018	<b>Fecha de fin:</b> 19 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	

Continúa en la próxima página

Tabla 3.4. Continuación de la página anterior

<b>Descripción:</b> Se implementa una funcionalidad que permite eliminar el menú creado, por lo que le envía al administrador un mensaje de certeza, y de esta forma no es recargada la base de datos.
--

Tabla 3.5. Tarea de ingeniería # 5

Tarea	
<b>Número de tarea:</b> 5	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Descargar archivo PHP	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 20 de febrero de 2018	<b>Fecha de fin:</b> 21 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite descargar el menú que se este visualizando, y de esta forma visualizar el contenido php del fichero.	

Tabla 3.6. Tarea de ingeniería # 6

Tarea	
<b>Número de tarea:</b> 6	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Listar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 22 de febrero de 2018	<b>Fecha de fin:</b> 23 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite listar menú del sitio y de esta forma muestra cada una de las acciones referente al menú y la visualización del menú se encuentra activo.	

### Tareas de ingeniería para la Iteración II

Para una segunda iteración, se elaboraron un total de 7 tareas de ingeniería. La mayoría, desglosadas a partir de las HU correspondientes a Administrar Secciones. A continuación, se describen algunas de las tareas realizadas, el resto pueden ser consultadas en los anexos de la investigación:

- **HU4 Administrar Secciones**

Tabla 3.7. Tarea de ingeniería # 7

Tarea	
<b>Número de tarea:</b> 7	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Crear Sección	

Continúa en la próxima página



Tabla 3.7. Continuación de la página anterior

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 23 de abril de 2018	<b>Fecha de fin:</b> 24 de abril de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite crear las diferentes secciones del sitio mediante un formulario, donde los campos no pueden estar vacíos, estas secciones se vinculan con los contenedores del sitio y renderizan su contenido.	

Tabla 3.8. Tarea de ingeniería # 8

<b>Tarea</b>	
<b>Número de tarea:</b> 8	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Actualizar Sección	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 25 de abril de 2018	<b>Fecha de fin:</b> 26 de abril de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite actualizar las diferentes secciones existentes en el sitio. Mostrará un formulario con los campos requeridos a modificar los cuáles son obligatorios, y mostrará la sección que se encuentre activa.	

Tabla 3.9. Tarea de ingeniería # 9

<b>Tarea</b>	
<b>Número de tarea:</b> 9	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Visualizar Sección	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 27 de abril de 2018	<b>Fecha de fin:</b> 28 de abril de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite mostrar los campos de la sección creada y verificar la correcta funcionalidad del mismo.	

Tabla 3.10. Tarea de ingeniería # 10

<b>Tarea</b>	
<b>Número de tarea:</b> 10	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Eliminar Sección	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 30 de abril de 2018	<b>Fecha de fin:</b> 1 de mayo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	

Continúa en la próxima página

Tabla 3.10. Continuación de la página anterior

<b>Descripción:</b> Se implementa una funcionalidad que permite eliminar el menú creado, por lo que le envía al administrador un mensaje de certeza, y de esta forma no es recargada la base de datos.
--

Tabla 3.11. Tarea de ingeniería # 11

Tarea	
<b>Número de tarea:</b> 11	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Descargar archivo PHP	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 2 de mayo de 2018	<b>Fecha de fin:</b> 3 de mayo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite descargar el archivo php de la sección que se este visualizando, y de esta forma verificar el contenido del fichero.	

Tabla 3.12. Tarea de ingeniería # 12

Tarea	
<b>Número de tarea:</b> 12	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Listar Sección	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 4 de mayo de 2018	<b>Fecha de fin:</b> 5 de mayo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite listar las secciones del sitio y de esta forma muestra cada una de las acciones referente a la sección y la visualización del menú se encuentra activo.	

Con las tareas de ingeniería definidas, se hace necesario establecer un conjunto de pruebas para comprobar la calidad de la solución implementada. Luego, se analizan estos casos de prueba y se ejecutan, lo que permite medir el nivel de cumplimiento con los objetivos de implementación trazados y el nivel de satisfacción del cliente.

### 3.2.1. Prototipos de interfaces funcionales

A continuación se muestran los prototipos de las interfaces funcionales principales, las cuales darán un mejor entendimiento de la propuesta de solución planteada:

#### **Gestionar Sección**

Esta funcionalidad se basa en renderizar los contenidos o secciones que se encuentren activo a cada una de las posiciones definidas.

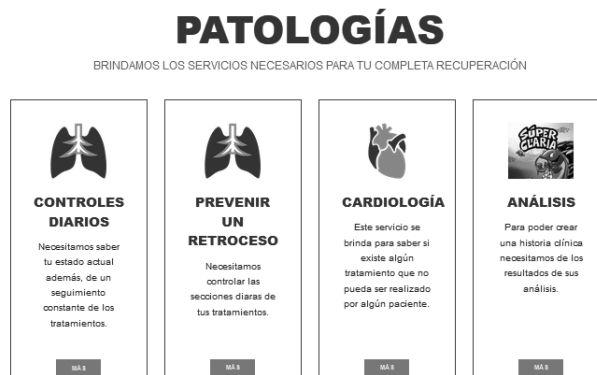


Figura 3.1. Representación de la sección Patologías del sistema. (Fuente: Elaboración Propia).



Figura 3.2. Representación de la sección Galería del sistema. (Fuente: Elaboración Propia).

Por lo que se gestiona mediante la administración del sitio de la siguiente forma:

Inicio / Secciones

### Secciones

Crear Sección

Mostrando 1-5 de 6 elementos.

ID	Contenedor	Prioridad	Nombre	Estado	Fecha de Creado	Fecha de Actualizado	Acciones
1	CONTAINER	4500	Sider	<input checked="" type="checkbox"/>	2018-05-14 12:03:05	2018-05-15 09:30:36	
2	FOOTER	201	footer	<input checked="" type="checkbox"/>	2018-05-15 09:03:26	2018-05-15 09:03:26	
3	TOP	5000	menu	<input checked="" type="checkbox"/>	2018-05-15 09:04:24	2018-05-15 09:30:15	
4	CONTAINER	4000	patologias	<input type="checkbox"/>	2018-05-15 09:29:19	2018-05-15 09:30:52	
5	CONTAINER	3999	galeria	<input type="checkbox"/>	2018-05-15 11:07:34	2018-05-15 11:07:34	

Figura 3.3. Representación de la gestión de secciones en la administración del sistema. (Fuente: Elaboración Propia).

### Gestionar Menú

Esta funcionalidad permite gestionar los menú existentes en el sitio, en este caso el menú principal, que se ubicará siempre en la sección superior del sistema, lo que de acuerdo a sus características se gestiona de

forma independiente a los contenidos, quedando así de la siguiente manera:



Figura 3.4. Representación de la sección Menú del sistema. (Fuente: Elaboración Propia).

Por lo que se gestiona mediante la administración del sitio de la siguiente forma:

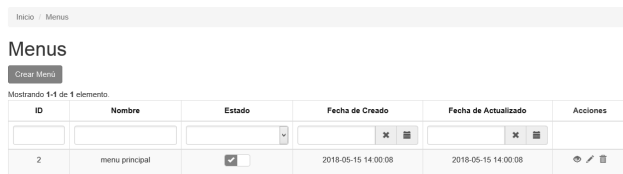


Figura 3.5. Representación de la gestión de menú en la administración del sistema. (Fuente: Elaboración Propia).

### 3.3. Fase IV: Pruebas

Las pruebas son un conjunto de actividades que se pueden planificar por adelantado y llevar a cabo sistemáticamente. Por esta razón, se deben definir en el proceso de la ingeniería del *software*. Todo esto, contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones.

La metodología *XP* divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente [25].

Se decide realizar las pruebas de aceptación a las funcionalidades implementadas, debido a que el objetivo de estas, es verificar que el sistema cumpla con los requisitos establecidos por el usuario. De esta forma se puede obtener el grado de satisfacción del cliente.

#### 3.3.1. Pruebas de aceptación y unitarias

Teniendo en cuenta las condiciones de la solución informática propuesta para el desarrollo de un sistema que permita la administración de contenidos de la plataforma web Medicando, se realizaron diversas pruebas de acuerdo a los niveles de aplicación asociados a esta disciplina, de las cuales se exponen sus resultados:

- Las pruebas funcionales basadas en el método de caja negra, se centran en verificar el cumplimiento de los requisitos funcionales del software, se emplean cuando se conoce la función específica para la que se diseñó la solución y se aplican a la interfaz del software, permitiendo obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales del programa, con el fin de encontrar la mayor cantidad de no conformidades existentes en el producto. Dentro de las

técnicas empleadas por las pruebas de caja negra se utilizó, partición de equivalencia, que según define Pressman: se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar [28].

- Una prueba de unidad es la prueba enfocada a los elementos testeables más pequeños del software. La prueba de unidad siempre está orientada a caja blanca. El diseño de casos de pruebas de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente. Es una forma de probar el correcto funcionamiento de un módulo o componente del código. Esto sirve para asegurar que cada elemento de estos funcione correctamente por separado. Los errores más frecuentes encontrados en este nivel de prueba están relacionados con las interfaces de comunicación entre componentes, interfaces entrada/salida, estructuras de datos locales, cálculos y flujos de control. Para comprobar el funcionamiento independiente de cada funcionalidad se empleó el método de caja blanca, guiado por la técnica del camino básico. Una vez determinados los caminos independientes de los requisitos funcionales importantes de la solución, se diseñaron casos de pruebas en función de dichos caminos.

Una prueba de unidad es la prueba enfocada a los elementos testeables más pequeños del software. La prueba de unidad siempre está orientada a caja blanca [29]. El diseño de casos de pruebas de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente. Es una forma de probar el correcto funcionamiento de un módulo o componente del código. Esto sirve para asegurar que cada elemento de estos funcione correctamente por separado. Los errores más frecuentes encontrados en este nivel de prueba están relacionados con las interfaces de comunicación entre componentes, interfaces entrada/salida, estructuras de datos locales, cálculos y flujos de control. Para comprobar el funcionamiento independiente de cada funcionalidad se empleó el método de caja blanca, guiado por la técnica del camino básico. Una vez determinados los caminos independientes que conforman cada tarea de programación asociada a las HU, se diseñaron casos de pruebas en función dichas tareas. Los resultados de la aplicación de esta técnica de camino básico se muestran en las Figuras 6, 7, 8 y 9.

### 3.3.2. Pruebas de aceptación para la Iteración I

Para la primera iteración, se definieron un total de 10 casos de pruebas de aceptación. Todas enfocadas a evaluar la implementación de la funcionalidad Administrar Menú. A continuación, se describen algunas de las pruebas realizadas, el resto pueden ser consultadas en los anexos de la investigación:

Tabla 3.13. Prueba de aceptación # 1

Caso de prueba de aceptación	
<b>Código:</b> HU1_P1	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Crear configuración de menú del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: crear menú para la configuración del sitio.	

Continúa en la próxima página

Tabla 3.13. Continuación de la página anterior

<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para crear los menú existentes en el sitio.</li> </ul>
<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la configuración de menú.</li> <li>3. Selecciona la opción crear menú.</li> <li>4. Se muestra un formulario con los campos a llenar, el botón guardar y el botón de cancelar.</li> <li>5. Se muestra una lista indicando que se creó correctamente el menú del sitio.</li> <li>6. El menú se muestra en el sitio si este se encuentra activo.</li> </ol>
<p><b>Resultados esperados:</b> Se creó un menú satisfactoriamente</p>

Tabla 3.14. Prueba de aceptación # 2

Caso de prueba de aceptación	
<b>Código:</b> HU1_P2	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Actualizar configuración de menú del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: actualizar menú para la configuración del sitio.	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de menú.</li> <li>3. Selecciona la opción actualizar menú.</li> <li>4. Se muestra un formulario con los campos a actualizar, el botón guardar y el botón de cancelar.</li> <li>5. Se muestra una vista indicando los campos actualizados y la creación correcta el menú del sitio.</li> <li>6. Después de actualizado el campo deseado se muestra dicho menú en el sitio si este se encuentra activo.</li> </ol>	
<b>Resultados esperados:</b> Se actualizó el menú satisfactoriamente	

Tabla 3.15. Prueba de aceptación # 3

Caso de prueba de aceptación	
<b>Código:</b> HU1_P3	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Visualizar configuración de menú del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: visualizar menú para la configuración del sitio.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de menú.</li> <li>3. Selecciona la opción visualizar menú.</li> <li>4. Se muestra una vista con los campos rellenos anteriormente, además de los botones Volver, Actualizar, Crear y Eliminar.</li> </ol>	
<b>Resultados esperados:</b> Se visualizó el menú satisfactoriamente	

Tabla 3.16. Prueba de aceptación # 4

Caso de prueba de aceptación	
<b>Código:</b> HU1_P4	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Visualizar configuración de menú del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: eliminar menú para la configuración del sitio.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de menú.</li> <li>3. Selecciona la opción eliminar menú.</li> <li>4. Se muestra un mensaje de confirmación, con los botones Aceptar y Cancelar.</li> <li>5. Si presiona el botón Aceptar se muestra una vista indicando que el menú ha sido eliminado.</li> <li>6. Si presiona el botón Cancelar se muestra la vista que contiene la lista de los elementos.</li> </ol>	
<b>Resultados esperados:</b> Se eliminó el menú satisfactoriamente	

Tabla 3.17. Prueba de aceptación # 5

Caso de prueba de aceptación	
<b>Código:</b> HU1_P5	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Descargar archivo php	
<b>Descripción:</b> Prueba para la funcionalidad: descargar archivo php.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de menú.</li> <li>3. Selecciona la opción visualizar sección.</li> <li>4. Se muestra una vista con los campos rellenos anteriormente, así como los botones Volver, Actualizar, Crear y Eliminar, además de el botón que contiene el archivo php.</li> <li>5. Si presiona el botón Contenido php se muestra una ventana indicando si desea descargar, abrir y un botón para aceptar y otro para cancelar.</li> </ol>	
<b>Resultados esperados:</b> Se descargó el archivo php satisfactoriamente	

Tabla 3.18. Prueba de aceptación # 6

Caso de prueba de aceptación	
<b>Código:</b> HU1_P6	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Listar configuración de los Menú	
<b>Descripción:</b> Prueba para la funcionalidad: listar menú.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de menú.</li> <li>3. Si existen menú creados, se encuentran listados.</li> <li>4. Se muestra una vista con la lista de los menú creados y las acciones referentes a cada uno de ellos.</li> </ol>	
<b>Resultados esperados:</b> Se listaron los menú satisfactoriamente	



### 3.3.3. Pruebas de aceptación para la Iteración II

Para una segunda iteración, se diseñaron un total de 6 casos de pruebas de aceptación. Todas enfocadas a evaluar la implementación de la funcionalidad Administrar Secciones. Se describen algunas de las pruebas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 3.19. Prueba de aceptación # 7

Caso de prueba de aceptación	
<b>Código:</b> HU5_P1	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Crear configuración de las secciones del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: crear sección para la configuración del sitio.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para crear los menú existentes en el sitio.</li> </ul>	
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la configuración de secciones.</li> <li>3. Selecciona la opción crear sección.</li> <li>4. Se muestra un formulario con los campos a llenar, el botón guardar y el botón de cancelar.</li> <li>5. Se muestra una lista indicando que se creó correctamente la sección del sitio.</li> <li>6. La sección se muestra en el sitio si esta se encuentra activa.</li> </ol>	
<b>Resultados esperados:</b> Se actualizó la sección satisfactoriamente	

Tabla 3.20. Prueba de aceptación # 8

Caso de prueba de aceptación	
<b>Código:</b> HU5_P2	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Actualizar configuración de las secciones del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: actualizar sección para la configuración del sitio.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	

Continúa en la próxima página

Tabla 3.20. Continuación de la página anterior

<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de secciones.</li> <li>3. Selecciona la opción actualizar sección.</li> <li>4. Se muestra un formulario con los campos a actualizar, el botón guardar y el botón de cancelar.</li> <li>5. Se muestra una vista indicando los campos actualizados y la creación correcta de la sección del sitio.</li> <li>6. Después de actualizado el campo deseado se muestra dicha sección en el sitio si esta se encuentra activa.</li> </ol>
<p><b>Resultados esperados:</b> Se actualizó la sección satisfactoriamente</p>

Tabla 3.21. Prueba de aceptación # 9

Caso de prueba de aceptación	
<b>Código:</b> HU5_P3	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Visualizar configuración las secciones del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: visualizar sección para la configuración del sitio.	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de sección.</li> <li>3. Selecciona la opción visualizar sección.</li> <li>4. Se muestra una vista con los campos rellenos anteriormente, además de los botones Volver, Actualizar, Crear y Eliminar.</li> </ol>	
<b>Resultados esperados:</b> Se visualizó la sección satisfactoriamente	

Tabla 3.22. Prueba de aceptación # 10

Caso de prueba de aceptación	
<b>Código:</b> HU5_P4	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Visualizar configuración de las secciones del sitio	
<b>Descripción:</b> Prueba para la funcionalidad: eliminar sección para la configuración del sitio.	

Continúa en la próxima página

Tabla 3.22. Continuación de la página anterior

<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>
<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de sección.</li> <li>3. Selecciona la opción eliminar sección.</li> <li>4. Se muestra un mensaje de confirmación, con los botones Aceptar y Cancelar.</li> <li>5. Si presiona el botón Aceptar se muestra una vista indicando que la sección ha sido eliminada.</li> <li>6. Si presiona el botón Cancelar se muestra la vista que contiene la lista de los elementos.</li> </ol>
<p><b>Resultados esperados:</b> Se eliminó la sección satisfactoriamente</p>

Tabla 3.23. Prueba de aceptación # 11

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU5_P5	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Descargar archivo php	
<b>Descripción:</b> Prueba para la funcionalidad: descargar archivo php.	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>	
<p><b>Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de sección.</li> <li>3. Selecciona la opción visualizar sección.</li> <li>4. Se muestra una vista con los campos rellenos anteriormente, así como los botones Volver, Actualizar, Crear y Eliminar, además de el botón que contiene el archivo php.</li> <li>5. Si presiona el botón Contenido php se muestra una ventana indicando si desea descargar, abrir y un botón para aceptar y otro para cancelar.</li> </ol>	
<p><b>Resultados esperados:</b> Se descargó el archivo php satisfactoriamente</p>	

Tabla 3.24. Prueba de aceptación # 12

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU5_P6	<b>Historia de usuario:</b> 1

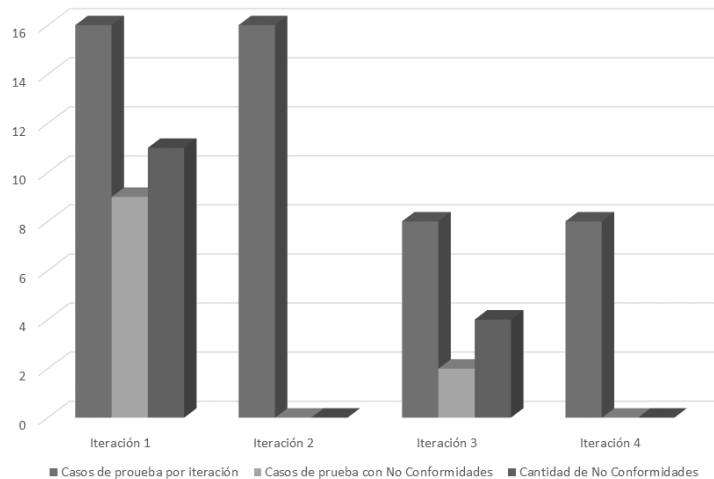
Continúa en la próxima página

Tabla 3.24. Continuación de la página anterior

<b>Nombre:</b> Listar configuración de las secciones
<b>Descripción:</b> Prueba para la funcionalidad: listar sección.
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"> <li>• El administrador debe estar previamente autenticado.</li> <li>• El administrador debe contar con una vista de configuración para actualizar los menú existentes en el sitio.</li> </ul>
<b>Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. El administrador entra a la configuración del sitio.</li> <li>2. El administrador selecciona la opción de configuración de sección.</li> <li>3. Si existen secciones creadas, se encuentran listadas.</li> <li>4. Se muestra una vista con la lista de las secciones creadas y las acciones referentes a cada uno de ellas.</li> </ol>
<b>Resultados esperados:</b> Se listó la sección satisfactoriamente

### 3.4. Análisis de las pruebas de aceptación

Se desarrollaron un total de 23 casos de pruebas de aceptación. Estas pruebas fueron realizadas de forma organizada, por cada iteración definida. A continuación, se muestra en gráficas, el nivel de satisfacción alcanzado en cada iteración.



Como se puede observar en la representación, en la primera iteración se realizaron un total de 16 casos de pruebas, de ellas 9 obtuvieron no conformidades alcanzando así, un total de 11 no conformidades, a

estas funcionalidades se les detectó que notificaban al administrador de forma directa, sin validar que los campos estuviesen vacíos. Lo que permitió corregir la falla, e incorporar la seguridad de esta operación en el sistema, por lo que para ello en la segunda iteración se realizaron pruebas de regresión a los mismos casos de pruebas, lo que permitió corregir los errores cometidos en la iteración. En tanto, en la segunda iteración, se realizaron 8 casos de pruebas, de ellas 2 obtuvieron no conformidades alcanzando así, un total de 4 no conformidades, por lo que en la segunda iteración se corrigieron las fallas cometidas, realizando las pruebas de regresión a esta iteración. Las correcciones antes descritas permitieron un 100 por ciento de satisfacción en las iteraciones finales del producto, comprobando el correcto funcionamiento de las funcionalidades implementadas.

Una vez concluida la implementación se planificó un encuentro con los usuarios finales para determinar si realmente el producto cumple con sus expectativas. De una muestra de 13 usuarios, 10 mostraron un nivel de satisfacción alto, 2, medio y 1 sólo no estuvo conforme con la solución propuesta, de acuerdo a las respuestas de las preguntas realizadas. Estableciendo un análisis de los resultados, se considera, que es factible la solución de integración propuesta.



### 3.5. Conclusiones del capítulo

En este capítulo se especificó el proceso de implementación del sistema a partir del desglose de las HU en tareas de ingeniería, lo que permitió especificar los procedimientos necesarios para dar cumplimiento a cada HU. Además se definieron y aplicaron las pruebas de aceptación y regresión a las funcionalidades desarrolladas, para dar un mejor entendimiento del sistema realizado. Estas pruebas permitieron detectar fallas en el sistema y corregirlas, además de la entrevista realizada a los usuarios finales, que permite comprobar la expectativa del producto y la satisfacción del cliente.

---

## Conclusiones

---

Con el desarrollo de la presente investigación se obtuvo un sistema para la administración de contenidos de la plataforma web Medicando, para lo cual:

- La implementación de la propuesta de solución permite la configuración e interacción dinámica con los contenidos que se visualizan en la plataforma web Medicando.
- La realización de las pruebas de aceptación, regresión y unidad permitieron detectar y corregir los errores durante la implementación, lo cual posibilitó un nivel Alto de satisfacción en el cliente y permitió la obtención de una versión estable del producto.

---

## Recomendaciones

---

A partir de los resultados obtenidos se recomienda:

- Incorporar el manejo de Plantillas, teniendo en cuenta las posiciones definidas anteriormente.
- Incorporar el manejo de Temas, Widgets y Plugins, basándose en el diseño del sitio.



- CMS** Counter Management Systems. [2](#), [3](#), [5](#), [9–11](#), [15](#), [18](#), [26](#)
- CRC** Clase-Responsabilidad-Colaborador. [32](#)
- GRASP** Patrones Generales de Asignación de Responsabilidades. [31](#)
- HU** Historia de usuarios. [21](#), [22](#), [32](#), [35](#), [36](#), [38](#), [52](#)
- IDE** Entorno de desarrollo integrado. [16](#)
- MLD** Menú Lateral Derecho. [8](#)
- MLI** Menú Lateral Izquierdo. [8](#)
- MP** Menú Principal. [8](#)
- MVC** Modelo Vista Controlador. [16](#)
- ORM** Object-Relational Mapping. [31](#)
- PHP** . [15](#)
- RnF** Requisitos no funcionales. [26](#), [27](#)
- RUP** . [12](#), [13](#)
- UCI** Universidad de las Ciencias Informáticas. [2](#)
- UML** Lenguaje de modelado unificado. [13](#)
- XP** Programación Extrema. [11–13](#), [36](#), [42](#)

---

## Referencias bibliográficas

---

- [1] Yeter Caraballo Pérez. «La gestión de contenidos en portales Web». En: *Acimed* 15.3 (2007), págs. 0-0 (vid. pág. 1).
- [2] C. Caridad Fresno Chávez Yuniór Dominguez Lobaina. *Virtualmend: un sistema gestor de contenidos de aprendizaje para la universalización de la educación superior*. ACIMED. 2007. URL: [http://scielo.sld.cu/scielo.php/script=sci\\_arttextpid=S1024-94352007000100012](http://scielo.sld.cu/scielo.php/script=sci_arttextpid=S1024-94352007000100012) (vid. pág. 1).
- [3] Deysi Alba Labaut Jorge Dayán Aguiar Cedeño Raúl Alejandro Valdés Pavón. *Propuesta del sitio Web para la Gestión de Contenidos sobre Nanociencias y Nanotecnologías del Centro de Estudios Avanzados de Cuba*. 2011. URL: <http://www.redalyc.org/html/1814/18142229401> (vid. pág. 1).
- [4] *INFOMED. Red en salud de Cuba*. 2018. URL: <http://www.sld.cu/acercade> (vid. pág. 1).
- [5] Daysi Alba Labaut Jorge Dayan Aguiar Cedeño Raúl Alejandro Valdés Pavón. «Propuesta del sitio Web para la Gestión de Contenidos sobre Nanociencias y Nanotecnologías del Centro de Estudios Avanzados de Cuba». En: (2011), pág. 11. URL: <http://www.redalyc.org/pdf/1814/181422294010.pdf> (vid. pág. 1).
- [6] Pilar Roqué. «Portales de salud». En: (2007). URL: <http://www.uoc.edu/uocpapers/4/dt/esp/roque.pdf> (vid. pág. 5).
- [7] C. Caridad Fresno Chávez Yuniór Dominguez Lobaina. *Virtualmend: un sistema gestor de contenidos de aprendizaje para la universalización de la educación superior*. ACIMED. 2007. URL: [http://scielo.sld.cu/scielo.php/script=sci\\_arttextpid=S1024-94352007000100012](http://scielo.sld.cu/scielo.php/script=sci_arttextpid=S1024-94352007000100012) (vid. pág. 7).
- [8] *CENTRO DE INGENIERÍA GENÉTICA Y BIOTECNOLOGÍA*. 2017. URL: <http://www.cigb.edu.cu/es/> (vid. pág. 8).
- [9] Juan Gabriel Valdés. «Modulos de visualización de la información paciente-especialista para la plataforma de gestión de videojuegos Medicando». En: (2016) (vid. pág. 10).
- [10] Suman Ninoriya y col. «CMS, LMS and LCMS for elearning». En: *International Journal of Computer Science Issues (IJCSI)* 8.2 (2011), págs. 1694-0814 (vid. pág. 10).

- 
- [11] José Luis Bautista. «Joomla. Creación de sitios web con contenido dinámico». En: *Campus Europeo de* (2012) (vid. pág. 10).
- [12] Sergi Tous Rodríguez. «Katworks.es: Creación de un sitio web basado en el sistema WordPress». En: (2015) (vid. pág. 11).
- [13] Juan Antonio Acedo Luque. «Diseño web con drupal 7». En: (2015). URL: [http://oa.upm.es/40023/10/TFG\\_Juan\\_Antonio\\_Acedo\\_Luque.pdf](http://oa.upm.es/40023/10/TFG_Juan_Antonio_Acedo_Luque.pdf) (vid. pág. 11).
- [14] Patricio Letelier. «Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)». En: (2006) (vid. pág. 12).
- [15] Andrés Navarro Cadavid, Juan Daniel Fernández Martínez y Jonathan Morales Vélez. «Metodologías ágiles para el desarrollo de software». En: *Prospectiva* 11.2 (2013), págs. 30-39 (vid. pág. 13).
- [16] Andrés Navarro Cadavid, Juan Daniel Fernández Martínez y Jonathan Morales Vélez. «Metodologías ágiles para el desarrollo de software». En: *Prospectiva* 11.2 (2013), págs. 30-39 (vid. pág. 13).
- [17] ÁNGEL Cobo y col. «PHP y MySQL». En: *Tecnología para el desarrollo de aplicaciones web. España. Ediciones Diaz Santos* (2005) (vid. pág. 15).
- [18] SERGIO LUJÁN MORA y ARMANDO SUÁREZ CUETO. *Introducción a Xampp y MySQL*. Jul. de 2013. URL: <https://ocw.ua.es/es/ingenieria-y-arquitectura/introduccion-a-xampp-y-mysql-2012.html> (vid. pág. 15).
- [19] Josep Maria Camps Riba. «Diseño e implementación de un marco de trabajo (framework) de presentación para aplicaciones JEE.» En: (2015). URL: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/876/1/00765tfc.pdf> (vid. pág. 15).
- [20] YII FRAMEWORK. «The Definitive Guide to Yii». En: (2011) (vid. pág. 16).
- [21] *Entorno de desarrollo integrado*. Enero 25 de 2013. URL: <https://fergarcia.wordpress.com/2013/01/25/entorno-de-desarrollo-integrado-ide/> (vid. pág. 18).
- [22] Miguel Angel Alvarez. «jQuery». En: (2015). URL: <http://www.cav.jovenclub.cu/comunidad/datos/descargas/jquery.pdf> (vid. pág. 18).
- [23] Juan Pavón Mestras. «Aplicaciones Web/Sistemas Web». En: (2013-2014). URL: <https://www.fdi.ucm.es/profesor/jpavon/web/26-Bootstrap.pdf> (vid. pág. 19).
- [24] *Revista Digital*. 2018. URL: <http://bibliotecaprofesional.com/que-es-renderizar-una-pagina-web/> (vid. pág. 20).
- [25] Gerardo Fernández Escribano. «Introducción a Extreme Programming». En: 2012, pág. 14. URL: <https://aalbertovargasc.files.wordpress.com/2011/07/presentacion-xp.pdf> (vid. págs. 22, 41).

- [26] José Joskowicz. «Reglas y Prácticas en eXtreme Programming». En: 2008, pág. 22. URL: <https://iie.fing.edu.uy/~josej/docs/XP%2020-%2020Jose%2020Joskowicz.pdf> (vid. págs. 22, 35).
- [27] Craig Larman. *UML y Patrones*. Pearson Educación eMadrid Madrid, 2003 (vid. pág. 30).
- [28] Roger S Pressman. *Ingeniería de Software. Un enfoque práctico*. Ed. por Pablo Roig Vázquez. 2010 (vid. pág. 42).
- [29] CINDY CAMPOS CHIU. «LAS PRUEBAS EN EL DESARROLLO DE SOFTWARE». Tesis doct. 2015. URL: [http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/7627/Las\\_pruebas\\_en\\_el\\_desarrollo\\_de\\_software.pdf/sequence1](http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/7627/Las_pruebas_en_el_desarrollo_de_software.pdf/sequence1) (vid. pág. 42).

# Apéndices

**Anexo 1: Historias de Usuario**

Tabla 25. Historia de usuario # 3

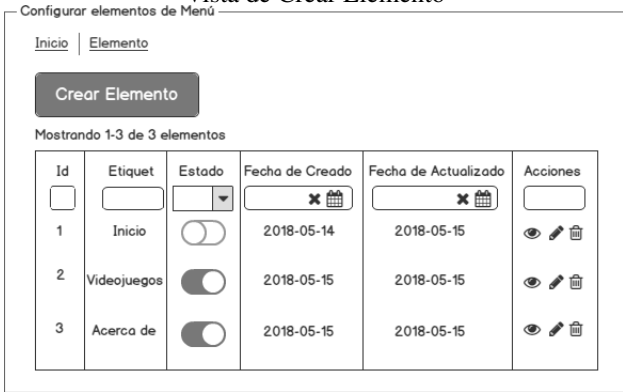
Historia de usuario																									
<b>Número:</b> 3	<b>Nombre:</b> Configurar elementos de menú. (Fuente: Elaboración propia).																								
<b>Usuario:</b> Administrador																									
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta																								
<b>Puntos estimados:</b> 2.0	<b>Iteración asignada:</b> 1																								
<b>Programador responsable:</b> Solanch de la C. Ríos Portillo																									
<p><b>Descripción:</b> Permite al administrador crear, modificar, activar y eliminar los elementos a exponer en el menú del sitio:</p> <ul style="list-style-type: none"> <li>• Id</li> <li>• Etiqueta</li> <li>• Estado</li> <li>• Fecha de creado</li> <li>• Fecha de actualizado</li> <li>• Acciones</li> </ul>																									
<b>Observaciones:</b>																									
<p><b>Interfaz:</b></p> <div style="text-align: center;"> <p>Vista de Crear Elemento</p>  </div> <p>The screenshot shows a web interface titled 'Configurar elementos de Menú'. It has two tabs: 'Inicio' and 'Elemento'. A 'Crear Elemento' button is visible. Below it, it says 'Mostrando 1-3 de 3 elementos'. There is a table with the following data:</p> <table border="1"> <thead> <tr> <th>Id</th> <th>Etiquet</th> <th>Estado</th> <th>Fecha de Creado</th> <th>Fecha de Actualizado</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Inicio</td> <td><input type="checkbox"/></td> <td>2018-05-14</td> <td>2018-05-15</td> <td> </td> </tr> <tr> <td>2</td> <td>Videojuegos</td> <td><input checked="" type="checkbox"/></td> <td>2018-05-15</td> <td>2018-05-15</td> <td> </td> </tr> <tr> <td>3</td> <td>Acerca de</td> <td><input checked="" type="checkbox"/></td> <td>2018-05-15</td> <td>2018-05-15</td> <td> </td> </tr> </tbody> </table>		Id	Etiquet	Estado	Fecha de Creado	Fecha de Actualizado	Acciones	1	Inicio	<input type="checkbox"/>	2018-05-14	2018-05-15		2	Videojuegos	<input checked="" type="checkbox"/>	2018-05-15	2018-05-15		3	Acerca de	<input checked="" type="checkbox"/>	2018-05-15	2018-05-15	
Id	Etiquet	Estado	Fecha de Creado	Fecha de Actualizado	Acciones																				
1	Inicio	<input type="checkbox"/>	2018-05-14	2018-05-15																					
2	Videojuegos	<input checked="" type="checkbox"/>	2018-05-15	2018-05-15																					
3	Acerca de	<input checked="" type="checkbox"/>	2018-05-15	2018-05-15																					

Tabla 26. Historia de usuario # 4

Historia de usuario	
<b>Número:</b> 4	<b>Nombre:</b> Administrar FAQs. (Fuente: Elaboración propia).
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 2.0	<b>Iteración asignada:</b> 1

Continúa en la próxima página

Tabla 26. Continuación de la página anterior

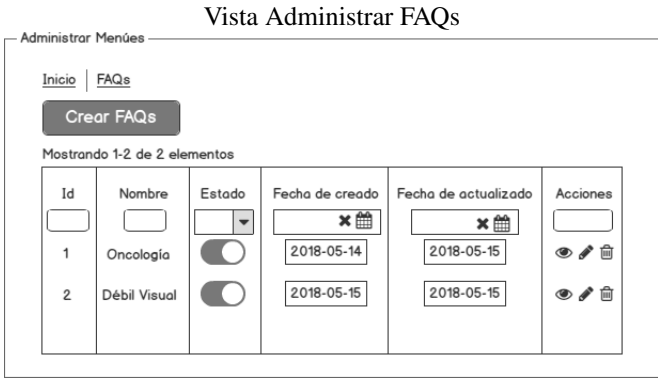
<b>Programador responsable:</b> Solanch de la C. Ríos Portillo
<p><b>Descripción:</b> Permite al administrador adicionar eliminar, modificar, listar y mostrar, las preguntas relacionadas con el sitio:</p> <ul style="list-style-type: none"> <li>• Tema</li> <li>• Título</li> <li>• Pregunta</li> <li>• Respuesta</li> </ul>
<b>Observaciones:</b>
<p><b>Interfaz:</b></p> 

Tabla 27. Historia de usuario # 5

Historia de usuario	
<b>Número:</b> 5	<b>Nombre:</b> Autenticar usuario. (Fuente: Elaboración propia).
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 2.0	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Solanch de la C. Ríos Portillo	
<p><b>Descripción:</b> Permite al administrador gestionar cada una de las posiciones del sitio:</p> <ul style="list-style-type: none"> <li>• Usuario</li> <li>• Contraseña</li> </ul>	
<b>Observaciones:</b> Los campos son obligatorios	

Continúa en la próxima página

Tabla 27. Continuación de la página anterior

**Interfaz:**

Vista de Autenticar Usuario

**Anexo 2: Tareas de Ingeniería**

- **HU Configurar elementos de menú**

Tabla 28. Tarea de ingeniería # 13

Tarea	
<b>Número de tarea:</b> 13	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Crear elementos de Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 26 de febrero de 2018	<b>Fecha de fin:</b> 27 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite crear los elementos de menú del sitio mediante un formulario, donde los campos no pueden estar vacíos y de esta forma hacer la página dinámica.	

- **HU Configurar elementos de menú**

Tabla 29. Tarea de ingeniería # 14

Tarea	
<b>Número de tarea:</b> 14	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Actualizar elementos de menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 28 de febrero de 2018	<b>Fecha de fin:</b> 28 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite actualizar los elementos de menú existentes en el sitio. Mostrará un formulario donde los campos requeridos son obligatorios y mostrará el elemento de menú que se encuentre activo.	

- **HU Configurar elementos de menú**



Tabla 30. Tarea de ingeniería # 15

Tarea	
<b>Número de tarea:</b> 15	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Visualizar elementos de menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 1 de marzo de 2018	<b>Fecha de fin:</b> 2 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite mostrar los campos de los elementos de menú creado y verificar la correcta funcionalidad del mismo.	

- **HU Configurar elementos de menú**

Tabla 31. Tarea de ingeniería # 16

Tarea	
<b>Número de tarea:</b> 16	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Eliminar elementos de menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 5 de marzo de 2018	<b>Fecha de fin:</b> 6 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite eliminar el elemento de menú creado y deseado, por lo que le envía al administrador un mensaje de certeza, y de esta forma no es recargada la base de datos.	

- **HU Configurar elementos de menú**

Tabla 32. Tarea de ingeniería # 17

Tarea	
<b>Número de tarea:</b> 17	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Listar elementos de menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 7 de marzo de 2018	<b>Fecha de fin:</b> 8 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite listar los elementos de menú existentes en el sitio y de esta forma muestra cada una de las acciones referente al elemento y la visualización del elemento que se encuentra activo.	

- **HU Administrar Menú**

Tabla 33. Tarea de ingeniería # 18

Tarea	
<b>Número de tarea:</b> 18	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Crear Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 26 de febrero de 2018	<b>Fecha de fin:</b> 27 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite crear el menú del sitio mediante un formulario, donde los campos no pueden estar vacíos y de esta forma hacer la página dinámica.	

- **HU Administrar Menú**

Tabla 34. Tarea de ingeniería # 19

Tarea	
<b>Número de tarea:</b> 19	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Actualizar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 28 de febrero de 2018	<b>Fecha de fin:</b> 28 de febrero de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite actualizar los menú existentes en el sitio. Mostrará un formulario donde los campos requeridos son obligatorios y mostrará el menú que se encuentre activo.	

- **HU Administrar Menú**

Tabla 35. Tarea de ingeniería # 20

Tarea	
<b>Número de tarea:</b> 20	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Visualizar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 1 de marzo de 2018	<b>Fecha de fin:</b> 2 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite mostrar los campos de los menues creado y verificar la correcta funcionalidad del mismo.	

- **HU Administrar Menú**

Tabla 36. Tarea de ingeniería # 21

Tarea
-------

Continúa en la próxima página

Tabla 36. Continuación de la página anterior

<b>Número de tarea:</b> 21	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Eliminar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 5 de marzo de 2018	<b>Fecha de fin:</b> 6 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite eliminar menú creado y deseado, por lo que le envía al administrador un mensaje de certeza, y de esta forma no es recargada la base de datos.	

• **HU Administrar Menú**

Tabla 37. Tarea de ingeniería # 22

Tarea	
<b>Número de tarea:</b> 22	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Listar Menú	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.3
<b>Fecha de inicio:</b> 7 de marzo de 2018	<b>Fecha de fin:</b> 8 de marzo de 2018
<b>Programador responsable:</b> Solanch de la Caridad Ríos Portillo	
<b>Descripción:</b> Se implementa una funcionalidad que permite listar los menú existentes en el sitio y de esta forma muestra cada una de las acciones referente al elemento y la visualización del elemento que se encuentra activo.	

**Anexo 3: Pruebas unitarias**


<b>Historia de usuario:</b> Administrar secciones. <b>Tarea de programación:</b> Eliminar sección.	
<b>Código:</b> <pre>public function actionDelete(\$id) {     1 \$this-&gt;findModel(\$id)-&gt;delete();      return \$this-&gt;redirect(['index']); }</pre>	<b>Grafo de flujo:</b> 
<b>Complejidad ciclomática:</b> <ul style="list-style-type: none"> <li>• <math>V(G) = 1.</math></li> <li>• <math>V(G) = A - N + 2 = 0 - 1 + 2 = 1.</math></li> <li>• <math>V(G) = P + 1 = 0 + 1 = 1.</math></li> </ul>	<b>Camino lineal independiente:</b> <ul style="list-style-type: none"> <li>• 1</li> </ul>
<b>Observación:</b> El caso de prueba diseñado para la tarea de programación Eliminar sección cubre el camino lineal independiente identificado.	

Figura 6. Prueba unitaria para la tarea de ingeniería Eliminar Sección.

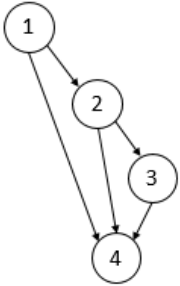
<b>Historia de usuario:</b> Administrar secciones. <b>Tarea de programación:</b> Actualizar sección.	
<b>Código:</b> <pre>public function actionUpdate(\$id) {     1 \$model = \$this-&gt;findModel(\$id);      1 if (\$model-&gt;load(Yii::\$app-&gt;request- -&gt;post()) &amp;&amp; \$model-&gt;save()) {     2 if(\$this-&gt;manageFiles(\$model)){     3     return \$this-&gt;redirect(['view', 'id' =&gt; \$model-&gt;id]);     }     }      4 return \$this-&gt;render('update', [ 'model' =&gt; \$model, ]); }</pre>	<b>Grafo de flujo:</b> 
<b>Complejidad ciclomática:</b> <ul style="list-style-type: none"> <li>• <math>V(G) = 3.</math></li> <li>• <math>V(G) = A - N + 2 = 5 - 4 + 2 = 3.</math></li> <li>• <math>V(G) = P + 1 = 2 + 1 = 3.</math></li> </ul>	<b>Caminos lineales independientes:</b> <ul style="list-style-type: none"> <li>• 1, 2, 3, 4</li> <li>• 1, 2, 4</li> <li>• 1, 4</li> </ul>
<b>Observación:</b> El caso de prueba diseñado para la tarea de programación Actualizar sección cubre todos los caminos lineales independientes identificados.	

Figura 7. Prueba unitaria para la tarea de ingeniería Actualizar Sección.


<b>Historia de usuario:</b> Administrar secciones. <b>Tarea de programación:</b> Visualizar sección.	
<b>Código:</b> <pre>public function actionView(\$id) {     1 return \$this-&gt;render('view', [ 'model' =&gt; \$this-&gt;findModel(\$id), ]); }</pre>	<b>Grafo de flujo:</b> 
<b>Complejidad cilomática:</b> <ul style="list-style-type: none"> <li>• <math>V(G) = 1.</math></li> <li>• <math>V(G) = A - N + 2 = 0 - 1 + 2 = 1.</math></li> <li>• <math>V(G) = P + 1 = 0 + 1 = 1.</math></li> </ul>	<b>Camino lineal independiente:</b> <ul style="list-style-type: none"> <li>• 1</li> </ul>
<b>Observación:</b> El caso de prueba diseñado para la tarea de programación Visualizar sección cubre el camino lineal independiente identificado.	

Figura 8. Prueba unitaria para la tarea de ingeniería Visualizar Sección.

<b>Historia de usuario:</b> Administrar secciones. <b>Tarea de programación:</b> Crear sección.	
<b>Código:</b>  <pre> public function actionCreate() {     1 \$model = new Section();      1 if (\$model-&gt;load(Yii::\$app-&gt;request-&gt;post()) &amp;&amp; \$model-&gt;save()) {     2     if(\$this-&gt;manageFiles(\$model)){     3         return \$this-&gt;redirect(['index']);     4     }else{     5         return \$this-&gt;render('create', [             'model' =&gt; \$model,         ]);     6     }     7 }     8 }     9 } </pre>	<b>Grafo de flujo:</b>  <pre> graph TD     1((1)) --&gt; 2((2))     1((1)) --&gt; 4((4))     1((1)) --&gt; 6((6))     2((2)) --&gt; 3((3))     2((2)) --&gt; 4((4))     3((3)) --&gt; 5((5))     4((4)) --&gt; 5((5))     5((5)) --&gt; 6((6)) </pre>
<b>Complejidad ciclomática:</b> <ul style="list-style-type: none"> <li>• <math>V(G) = 3</math>.</li> <li>• <math>V(G) = A - N + 2 = 7 - 6 + 2 = 3</math>.</li> <li>• <math>V(G) = P + 1 = 2 + 1 = 3</math>.</li> </ul>	<b>Caminos lineales independientes:</b> <ul style="list-style-type: none"> <li>• 1, 2, 3, 5, 6</li> <li>• 1, 2, 4, 5, 6</li> <li>• 1, 6</li> </ul>
<b>Observación:</b> El caso de prueba diseñado para la tarea de programación Crear sección cubre todos los caminos lineales independientes identificados.	

Figura 9. Prueba unitaria para la tarea de ingeniería Crear Sección.

---

#### **Anexo 4:** Encuesta de satisfacción del cliente

Modelo de encuesta aplicada a los usuarios finales <sup>1</sup> de la aplicación con el objetivo de medir el nivel de satisfacción con el uso del sistema de administración de la información para el análisis semántico de la interacción paciente-plataforma.

Cuestionario:

1. ¿Le resulta cómodo interactuar con el sistema propuesto?
2. ¿Piensa que es útil la integración del cliente con el sistema para la rehabilitación?
3. ¿Considera que la generación del Sistema para la administración de contenidos de la plataforma web Medicando contribuye a mejorar el desarrollo de pacientes en rehabilitación?
4. ¿Considera que la información generada a partir de una solicitud realizada es suficiente?

---

<sup>1</sup>13 usuarios del centro Vertex