

# *Desarrollo de un sistema basado en casos para la identificación de opacidad de la cápsula posterior en el software PANDOC*

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias  
Informáticas

**Autor:**

Maidevis Rodríguez Moreno

**Tutores:**

Ing. Michel Alvarez Cancio

Lic. Reyder Cruz de la Osa

FACULTAD 2

18 de junio del 2015

Año 57 de la Revolución

## Declaración de autoría

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Autor:

---

Maidevis Rodríguez Moreno

Tutores:

---

Michel Alvarez Cancio

---

Reyder Cruz de la Osa

## *Dedicatoria*

*A mis padres por siempre apoyarme en cada una de mis decisiones, por ser mis guías y encaminarme en la vida, por su cariño, su amor y su comprensión.*

*A mi hermana Gleny por ser más que mi hermana mi amiga, mi compañera y mi mayor orgullo y por haberme dado el sobrino más lindo del mundo.*

*A mi hermanito Marco, al cual espero servirle de ejemplo en la vida y en su crecimiento profesional.*

*A mis sobrinos por su inocencia y por ser tan especiales.*

*A mi novio Gabriel por siempre darme su apoyo, por entenderme, ayudarme en todo y aceptarme tal cual soy.*

*A mis abuelitos que a pesar de ya no encontrarse entre nosotros significan mucho para mí.*

*A mis amigos Yanet y Roniel, que más que amigos han sido mis hermanos en todos estos años.*

## *AGRADECIMIENTOS*

*A mis padres por haber sido siempre mi apoyo, mis guías en la vida, por entenderme y ayudarme en todo los momentos que lo he necesitado. A mi mamita por ser mi amiga, por sus consejos, su cariño, por siempre pensar en mí, por ser esa madre que todos merecen tener, no me alcanzará la vida para agradecerte por ser exactamente como eres. A mi papito por aceptar mis decisiones y mis sueños y ayudarme a realizarlos, por todos los sacrificios que ha hecho por mí, por su cariño y su comprensión, porque sin el este recorrido no habría sido posible, gracias.*

*A mis hermanos por estar presentes en mi vida, por ayudarme y quererme siempre. A mi hermana Gleny por apoyarme en todo, por ser mi amiga, mi compañera y mi segunda mamá, por darme la oportunidad de experimentar el sentimiento de ser tía, por siempre estar para mí cuando lo he necesitado, por conocerme como nadie y entenderme como solo ella puede hacerlo. A mi hermanito Marco por su cariño y su ingenuidad, espero poder ser un ejemplo a seguir en su vida profesional.*

*A mis sobrinos por su inocencia, su cariño y por permitirme experimentar ese sentimiento de sentirme más que tía, madre.*

*A mi cuñado Yausbe por su ayuda incondicional, por su apoyo, su cariño, por ser mi segundo papá, por sus consejos y por haberse esforzado para lograr ser hoy una persona mejor.*

*A mis tíos Mario y Rosa por ayudarme siempre que han podido y por acogerme durante todo este tiempo.*

*A mi amiga Yanet por haber estado ahí para mí siempre que lo he necesitado, por ser mi amiga desde primer año y por mantener nuestra amistad a pesar de los obstáculos que se han*

*presentado, por sus consejos y su cariño y por su futuro bebé, el que espero pueda llamarme tía.*

*A mi amigo Rony que ha sido para mí más que un amigo, un hermano, por sus consejos, su cariño, por apoyarme siempre que lo he necesitado, por poner bien en alto el nombre de la amistad y demostrarme que si existen los verdaderos amigos. Por formar parte de mi vida y permitirme formar parte de la suya, gracias hermanito.*

*A mi novio Gabriel por apoyarme en todo, por haber estado a mi lado en muchos de los momentos difíciles que me han tocado vivir en los últimos tiempos, por entenderme, por darme su amor incondicional y porque sin su ayuda este sueño no habría sido posible. Por hacerme parte de su vida y estar presente en la mía. A su familia que hoy también es la mía, en especial a Mary y a Vilma por acogerme y considerarme como un miembro más de su familia y por darme su cariño.*

*A mi amigo Yoan por su cariño, su apoyo, por aceptarme aun cuando no me entienda, por escucharme y quererme tal cual soy, por permitirme ser su amiga y disfrutar de su compañía.*

*A mis tutores por haberme apoyado incondicionalmente en todo momento.*

*A Michel por confiar en mí y darme la oportunidad de poder realizar este trabajo de diploma, porque sin su ayuda, paciencia y compromiso no habría sido posible llegar hasta aquí.*

*A Rachel y Lázaro porque a pesar de llevar poco tiempo conociéndolos hemos compartido muy buenos momentos, por hacerme reír, por ser en mi vida hoy dos bichitos tan importantes y por demostrarme que cuando menos lo esperes puede surgir una amistad sincera la que espero dure por mucho tiempo.*

*A mis amigas Malena y Yilian que aunque no las vea, siempre serán una parte importante de mí, por todos y cada uno de los momentos que pasamos juntas y que estarán por siempre en mi mente.*

*A Yisel, Ana y Maricet por haber estado presentes en momentos importantes de mi vida, por su apoyo y por contribuir de una forma u otra en mi formación personal y profesional.*

*A Claudia Mary, Mayelín y Olga por haber formado parte de mi vida de una forma u otra durante esta importante etapa. Por todos los momentos que hemos compartido juntas gracias.*

*A mis compañeros de grupo en especial a Jairo por siempre hacerme reír, por ayudarme en momentos que lo he necesitado y por tenerme en cuenta, a Juan Miguel por tantas veces que me ayudó incondicionalmente, por dedicar parte de su tiempo a repasarme cada vez que lo necesité, a Erdin, Porras, Landy y Yendry por todas las veces que me ayudaron durante todo este tiempo.*

*A la Universidad de las Ciencias Informáticas por haberme dado esta maravillosa oportunidad.*

*A mis amigos y compañeros, a los que están y a los que no. En fin a todos los que de una forma u otra han formado parte de mi vida durante el transcurso de estos años.*

## **Resumen**

La opacidad de la cápsula posterior es actualmente uno de los aspectos más importantes en la cirugía de catarata de los tiempos modernos. Sigue siendo la complicación postoperatoria tardía más frecuente tras la cirugía de catarata asociada con disminución de la agudeza visual, deterioro de la sensibilidad al contraste y problemas de deslumbramiento que conllevan importantes repercusiones sociales, médicas y económicas.

El software PANDOC provee al oftalmólogo de una herramienta por medio de la cual este es capaz de cuantificar numéricamente y detectar diferencias de opacidad (a veces imperceptibles para el ojo humano), logrando así una evaluación objetiva del grado de opacidad, minimizando el sesgo de observación entre un médico y otro. Resaltando como necesidad la identificación de la OCP en estas imágenes, para lo cual se diseña un sistema basado en casos.

Por tal motivo la presente investigación se enmarca en la creación de un sistema basado en casos integrado al software PANDOC que permita identificar y cuantificar objetivamente la opacidad mediante el uso de las imágenes resultantes de la cámara Scheimpflug del Pentacam.

### **Palabras claves:**

Opacidad de la cápsula posterior, catarata, sistema basado en casos, cámara Scheimpflug del Pentacam.

## Índice

INTRODUCCIÓN .....	1
CAPÍTULO 1. IDENTIFICACIÓN DE OPACIDAD CAPSULAR EN PACIENTES OPERADOS DE CATARATA: BASES CONCEPTUALES.....	5
1.1 OPACIDAD DE LA CÁPSULA POSTERIOR: CONCEPTUALIZACIÓN.....	5
1.1.1 <i>Incidencia y prevención.</i> .....	5
1.2 SISTEMAS DESARROLLADOS PARA LA IDENTIFICACIÓN DE OCP. ....	6
1.2.1 <i>POCO</i> .....	6
1.2.2 <i>EPCO</i> .....	7
1.2.3 <i>AQUA</i> .....	7
1.2.4 <i>AA</i> .....	8
1.2.5 <i>Image J</i> .....	9
1.3 PROGRAMA ANALIZADOR DE OPACIDAD CAPSULAR: PANDOC .....	9
1.3.1 <i>Interfaz visual</i> .....	9
1.3.2 <i>Herramientas y funcionalidades</i> .....	11
1.4 SISTEMAS BASADOS EN EL CONOCIMIENTO .....	12
1.4.1 <i>Sistemas Basados en Reglas</i> .....	12
1.4.2 <i>Sistemas Basados en probabilidades</i> .....	14
1.4.3 <i>Sistemas Basados en Casos</i> .....	15
1.4.4 <i>Razonamiento basado en casos</i> .....	16
1.4.5 <i>Ingeniería del Conocimiento</i> .....	18
1.5 BASES DE CONOCIMIENTOS .....	21
1.6 MOTOR DE INFERENCIAS.....	24
1.7 CONCLUSIONES DEL CAPÍTULO.....	25
CAPÍTULO 2: HERRAMIENTAS USADAS DURANTE EL DESARROLLO.....	26
2.1 METODOLOGÍA DE SOFTWARE.....	26
2.2 LENGUAJE DE PROGRAMACIÓN .....	27
2.3 ENTORNO INTEGRADO DE DESARROLLO.....	28
2.4 SISTEMA GESTOR DE BASE DE DATOS.....	28
2.5 HERRAMIENTA PARA EL MODELADO .....	30
2.6 CONCLUSIONES DEL CAPÍTULO.....	30

CAPÍTULO 3: IDENTIFICACIÓN DE LA OPACIDAD DE LA CÁPSULA POSTERIOR EN IMÁGENES DEL PENTACAM .....	31
3.1 FASE DE PLANIFICACIÓN .....	31
3.1.1 <i>Historias de Usuario</i> .....	31
3.1.2 <i>Estimación de esfuerzos por Historias de Usuario</i> .....	32
3.1.3 <i>Plan de iteraciones</i> .....	33
3.1.4 <i>Plan de entrega</i> .....	34
3.1.5 <i>Implementación de la propuesta de solución</i> .....	34
3.1.6 <i>Tarjetas CRC</i> .....	35
3.1.7 <i>Patrones de diseño</i> .....	35
3.2 COMPONENTES DE LOS SISTEMAS BASADOS EN CASOS .....	36
3.2.1 <i>Definición de los rasgos predictores y los rasgos objetivos</i> .....	37
3.2.3 <i>Dominio de definición de cada rasgo</i> .....	38
3.2.4 <i>Peso informacional de cada rasgo</i> .....	38
3.2.5 <i>Funciones de comparación de rasgos</i> .....	39
3.2.7 <i>Representación de los casos</i> .....	40
3.2.8 <i>Obtener el nuevo caso a resolver</i> .....	40
3.2.9 <i>Comparación del nuevo caso con la Base de Casos</i> .....	40
3.2.10 <i>Obtención de los 'k' casos más semejantes</i> .....	41
3.2.11 <i>Selección del rasgo objetivo</i> .....	41
3.3 CONCLUSIONES DEL CAPÍTULO.....	41
CAPÍTULO 4: RESULTADOS Y VALIDACIÓN DEL SISTEMA .....	43
4.1 FASE DE IMPLEMENTACIÓN.....	43
4.1.2 <i>Iteración 1</i> .....	43
4.1.3 <i>Iteración 2</i> .....	45
4.1.4 <i>Iteración 3</i> .....	46
4.1.5 <i>Iteración 4</i> .....	46
4.2 PRUEBAS .....	46
4.2.1 <i>Pruebas de caja blanca</i> .....	47
4.2.2 <i>Pruebas de caja negra</i> .....	48
4.3 CONCLUSIONES DEL CAPÍTULO.....	54
CONCLUSIONES GENERALES.....	55
RECOMENDACIONES .....	56

REFERENCIAS BIBLIOGRÁFICAS.....	57
ANEXOS .....	61
ANEXO 1: HISTORIAS DE USUARIOS .....	61
ANEXO 2: TARJETAS CRC .....	64
ANEXO 3: TAREAS DE INGENIERÍA GENERADAS EN LA ITERACIÓN 3 .....	66
GLOSARIO DE TÉRMINOS.....	67

## Índice de ilustraciones

Ilustración 1. Software POCO .....	6
Ilustración 2. Software EPCO.....	7
Ilustración 3. Software AQUA.....	8
Ilustración 4. Software AA .....	8
Ilustración 5. Software Image J.....	9
Ilustración 6. Interfaz visual de PANDOC .....	10
Ilustración 7. Ciclo de vida de un sistema basado en casos (20) .....	16
Ilustración 8. Ciclo para la adquisición inicial del conocimiento (23).....	20
Ilustración 9. Representación del rasgo "Vecindad" .....	37
Ilustración 10. Representación del grafo de flujo de camino básico .....	47
Ilustración 11. Análisis realizado por PANDOC al tomograma del Paciente 1.....	50
Ilustración 12. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 1.....	51
Ilustración 13. Análisis realizado por PANDOC al tomograma del Paciente 2.....	51
Ilustración 14. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 2.....	51
Ilustración 15. Análisis realizado por PANDOC al tomograma del Paciente 3.....	52
Ilustración 16. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 3.....	52
Ilustración 17. Análisis realizado por PANDOC al tomograma del Paciente 4.....	52
Ilustración 18. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 4.....	53

## Índice de tablas

Tabla 1. Historia de usuario # 1 .....	31
Tabla 2. Historia de usuario # 7 .....	32
Tabla 3. Puntos de estimación por historias de usuario.....	32
Tabla 4. Plan de duración de las iteraciones .....	34
Tabla 5. Plan de entrega .....	34
Tabla 6. Tarjeta CRC de la clase Editor.....	35
Tabla 7. Dominio de definición de cada rasgo .....	38
Tabla 8. Peso informacional de cada rasgo .....	38
Tabla 9. Tiempo de implementación de las HU de la iteración #1 .....	43
Tabla 10. Tarea de ingeniería de la funcionalidad Crear BC .....	44
Tabla 11. Tarea de ingeniería de la funcionalidad Conexión .....	44
Tabla 12. Tarea de ingeniería de la funcionalidad Reconocer imagen de tomograma.....	44
Tabla 13. Tarea de ingeniería de la funcionalidad Reconocer centro de la estructura ocular .....	44
Tabla 14. Tiempo de implementación de las HU de la iteración # 2 .....	45
Tabla 15. Tarea de ingeniería de la funcionalidad Llenar BC .....	45
Tabla 16. Tarea de ingeniería de la funcionalidad Obtener nuevo caso .....	45
Tabla 17. Tarea de ingeniería de la funcionalidad Buscar región .....	46
Tabla 18. Tiempo de implementación de las HU de la iteración # 3 .....	46
Tabla 19. Tiempo de implementación de las HU de la iteración # 4 .....	46
Tabla 20. Caso de prueba de caja blanca para el camino básico 1.....	48
Tabla 21. Caso de prueba de aceptación para la HU Reconocer el centro de la estructura ocular .....	49
Tabla 22. Caso de prueba de aceptación para la HU Definir región del caso .....	49
Tabla 23. Caso de prueba de aceptación para la HU Calcular certeza de solución .....	49
Tabla 24. Comparación de resultados .....	53
Tabla 25. Resultado de las pruebas de aceptación .....	54
Tabla 26. Historia de usuario #2 .....	61
Tabla 27. Historia de usuario #3 .....	61
Tabla 28. Historia de usuario #4 .....	61
Tabla 29. Historia de usuario #5 .....	61
Tabla 30. Historia de usuario #6 .....	62
Tabla 31. Historia de usuario #8 .....	62
Tabla 32. Historia de usuario # 9 .....	62
Tabla 33. Historia de usuario # 10 .....	63
Tabla 34. Historia de usuario # 11 .....	63
Tabla 35. Historia de usuario # 12 .....	63
Tabla 36. Historia de usuario # 13 .....	64
Tabla 37. Tarjeta CRC de la clase Conexión .....	64
Tabla 38. Tarjeta CRC de la clase Lienzo.....	64
Tabla 39. Tarjeta CRC de la clase Recortador .....	65
Tabla 40. Tarjeta CRC de la clase Configuración .....	65
Tabla 41. Tarjeta CRC de la clase Geometría Computacional .....	65
Tabla 42. Tarjeta CRC de la clase Pixel .....	66
Tabla 43. Tarjeta CRC de la clase Caso.....	66
Tabla 44. Tarea de ingeniería de la funcionalidad Cargar casos similares de la BC.....	66

Tabla 45.Tarea de ingeniería de la funcionalidad Calcular similitud de rasgos ..... 66

## **Introducción**

En la actualidad miles de personas de todo el mundo padecen de la enfermedad de cataratas, sufriendo como consecuencia la pérdida total o parcial de la visión. La catarata no es más que opacidad en el cristalino del ojo, trayendo como consecuencia que la luz se disperse dentro del mismo y no sea posible enfocar en la retina, de esta forma crea una serie de imágenes difusas. Las cataratas generan problemas con la apreciación de cambios de colores y cambios de contrastes. Es la causa más común de ceguera tratable con cirugía, tiene diversas causas pero se le atribuye mayormente a la edad, acelerando este proceso si el paciente padece de enfermedades como la diabetes o hipertensión. Con mayor frecuencia esta enfermedad tiende a aparecer en pacientes mayores de cincuenta años de edad (1).

Tras una intervención quirúrgica el paciente puede recuperar su visibilidad total o parcialmente, pero no en todos los casos la cirugía es un éxito a largo plazo, pues en muchos de ellos el paciente puede presentar complicaciones postoperatorias. La complicación postoperatoria más frecuente a largo o mediano plazo es la Opacidad de la Cápsula posterior (OCP). La cápsula posterior es una estructura que anatómicamente queda por detrás de la lente intraocular implantada en el paciente operado de catarata, para garantizar una buena recuperación visual del mismo es imprescindible la transparencia de esta estructura. La OCP trae consigo deterioro de la sensibilidad al contraste y problemas de deslumbramiento que conllevan importantes repercusiones sociales, médicas y económicas. Esta complicación postoperatoria puede aparecer meses e incluso años después de realizada la cirugía; su incidencia en la actualidad se encuentra entre 0,7% y 47,6 % en los primeros cinco años de la cirugía, según un análisis de 90 estudios publicados, siendo en Cuba una cifra considerable que asciende hasta el 50% de los casos (2).

Disímiles investigaciones se han realizado con el objetivo de identificar los principales factores que influyen en la aparición de esta anomalía postoperatoria, un importante factor es la biocompatibilidad de la lente intraocular en relación con el saco capsular (3). Otro factor influyente son los implantes adicionales como los anillos de tensión capsular (Capsular Tension Ring: CTR), los cuales a pesar de su eficacia para la reducción significativa de la OCP no logran eliminarla totalmente (4). La aplicación de las nuevas tecnologías también han permitido obtener cierto grado de avance en este campo, el uso de técnicas como Cleanbagjet, la fotólisis y la pieza de mano láser las cuales se encuentran actualmente en estudio para definir su utilidad en la reducción de la OCP. También se menciona la terapia génica. En la comunidad médica internacional no existe consenso alguno acerca de la correcta cuantificación de la OCP, se deja a cada especialista la elección de alguna de las técnicas propuestas (2).

## *Introducción*

En los últimos años en todo el mundo se han desarrollado disímiles sistemas como POCO del inglés Posterior Capsule Opacification, EPCO (Evaluación de la opacidad de la cápsula posterior) (5), AQUA (Cuantificación automática de catarata secundaria) (6), y el sistema AA del inglés Aslam Analysis (7), dichos sistemas están basados en las imágenes en retroiluminación obtenidas por las lámparas de hendiduras (8). Estos sistemas han sido desarrollados con el objetivo de servir de apoyo a los médicos a la hora de dar un correcto diagnóstico. La mayoría de ellos tienen en común una desventaja producto de las imágenes resultantes de la lámpara de hendidura, como resultado de la reflexión del artefacto, efecto producido por la reflexión de la luz en la córnea, que es conocido como imágenes de Purkinje. Han sido desarrollados varios softwares propietarios para eliminar el efecto indeseable de dichas imágenes, con la función de múltiples imágenes de un mismo ojo (9), fotografiadas en direcciones ligeramente diferentes de la mirada (10), pero ello involucra una curva de aprendizaje muy larga, ya que es un procesamiento tedioso y engorroso para el paciente.

El Pentacam es un equipo oftalmológico de alta tecnología capaz de reconstruir imágenes tridimensionales de alta resolución del polo anterior del ojo. Estas imágenes son reconstruidas a partir de múltiples fotografías tomadas mediante una cámara rotacional del sistema Scheimpflug con que cuenta el equipo. Este moderno sistema Scheimpflug con que cuenta el Pentacam es capaz de capturar imágenes en múltiples meridianos (hasta 50 imágenes en 2 segundos) en un único escaneo automatizado (11). El análisis de estos tomogramas en 3D es de gran utilidad para la cuantificación objetiva de la OCP. Los tomogramas del Pentacam tienen además la ventaja de no presentar destellos de luz que interfieran con el análisis como sucede con las fotografías en retroiluminación, tomadas por los sistemas de adquisición de imágenes en lámpara de hendidura (3) (11).

En la Universidad de las Ciencias Informáticas (UCI) se encuentra el Centro de Investigación AIRI (Artificial Intelligence: Research and Innovation), el cual tiene como principal resultado y línea de investigación el software PANDOC (Programa Analizador de Opacidad Capsular), el mismo fue realizado en colaboración con el Instituto Cubano de Oftalmología “Ramón Pando Ferrer”. El software PANDOC provee al oftalmólogo de una herramienta por medio de la cual este es capaz de cuantificar numéricamente y detectar diferencias de opacidad (a veces imperceptibles para el ojo humano), logrando así una evaluación objetiva del grado de opacidad, minimizando el sesgo de observación entre un médico y otro. El software PANDOC actualmente no identifica de forma automática la OCP, por lo que surge como necesidad el desarrollo de un sistema basado en casos para su identificación y cuantificación objetiva, mediante el uso de las imágenes resultantes de la cámara Scheimpflug del Pentacam.

Dada la situación problemática expuesta anteriormente, se plantea como **problema a resolver**: ¿Cómo identificar las estructuras de opacidad capsular en tomogramas Scheimpflug del Pentacam? A raíz de

esto, se toma como **objeto de estudio**: Sistemas basados en el conocimiento para la detección de opacidad de la cápsula posterior, enmarcándose en el **campo de acción**: Identificar OCP en imágenes resultantes de los tomogramas de Scheimpflug del Pentacam en pacientes operados de catarata. En aras de dar solución al problema planteado se tiene como **objetivo general**: Implementar un sistema basado en casos integrado al software PANDOC para identificar la OCP mediante tomogramas Scheimpflug del PENTACAM.

Teniendo en cuenta el problema a resolver se formuló la siguiente Idea a Defender:

Con el desarrollo de un sistema basado en casos, se dotará al software PANDOC de una funcionalidad para la detección de la incidencia de la OCP en pacientes sometidos a cirugía de catarata, minimizando el sesgo de observación entre un especialista y otro.

Para dar cumplimiento al objetivo de la investigación se trazan las siguientes tareas investigativas:

- Estudio de los sistemas basados en el conocimiento
- Estudio del uso de bases de conocimientos en reconocimiento de patrones.
- Descripción de los pasos a seguir para el uso de un sistema basado en casos.
- Diseño de la solución propuesta en función de los requisitos especificados.
- Implementación del sistema basado en casos.
- Integración del sistema basado en casos al software PANDOC.
- Verificación mediante pruebas el funcionamiento del sistema.

Durante el desarrollo de la investigación es necesario utilizar varios métodos científicos, los cuales se mencionan a continuación.

**Analítico-Sintético:** Se utiliza con el objetivo de analizar las teorías, documentos e información referente al tema en cuestión, para luego extraer los elementos más importantes que se relacionan con la identificación de la OCP en pacientes operados de catarata.

**Histórico-Lógico:** Permite estudiar y analizar la trayectoria y evolución de software de análisis de OCP en pacientes operados de catarata, para poder comprender el nivel de desarrollo alcanzado por este tema en el mundo, además con estas experiencias se podrá tener un mayor entendimiento y comprensión de este tipo de sistemas, basándose en las experiencias existentes.

**Inductivo-Deductivo:** Se utiliza con el objetivo de inducir una serie de conocimientos referentes a la cuantificación objetiva de la OCP, para luego arribar a razonamientos que conlleven a la deducción de conocimientos que pueden ser aplicables al problema en cuestión.

La presente investigación está estructurada en cuatro capítulos; a continuación se describe el objetivo principal de cada uno de ellos:

### **Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases**

**Conceptuales.** En este capítulo se realiza la descripción y análisis desde el punto de vista teórico del problema general en que se enmarca la investigación, además se realiza un estudio del arte de los sistemas basados en el conocimiento y el razonamiento basado en casos como técnicas de la inteligencia artificial a utilizar.

**Capítulo 2: Herramientas usadas durante el desarrollo.** En este capítulo se describen las herramientas y tecnologías a utilizar durante el desarrollo de la solución. Son expuestos los argumentos que conllevan a la selección de dichas herramientas y tecnologías. Se explican las principales características de las metodologías de software y las ventajas que ofrece la escogida en la presente investigación.

### **Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del**

**Pentacam.** En este capítulo se realiza una descripción detallada de la solución propuesta con los aspectos definidos en la fundamentación teórica. Esta solución contiene los artefactos necesarios para el desarrollo del sistema a construir, entre ellos se encuentran las historias de usuarios y las tarjetas CRC.

**Capítulo 4: Resultados y validación del sistema.** En este capítulo se muestra el conjunto de pruebas realizadas a la herramienta y los resultados que arrojaron las mismas. Además se valida que el diseño realizado cumpla con la calidad requerida y el sistema implementado satisface las necesidades del cliente.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

## **Capítulo 1. Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales.**

En el presente capítulo se realiza la descripción y análisis desde el punto de vista teórico del problema general en que se enmarca la investigación. Como puntos fundamentales se abordan los principales conceptos asociados al dominio del problema, como el de incidencia y prevención de la OCP. También se realiza un estudio del arte de los principales sistemas para la identificación de OCP que existen en el mundo, se estudian los sistemas basados en el conocimiento y el razonamiento basado en casos como técnicas de la inteligencia artificial a utilizar, las mismas ayudarán en la identificación de la opacidad capsular en pacientes sometidos a cirugía de catarata.

### **1.1 Opacidad de la Cápsula Posterior: Conceptualización.**

La cápsula posterior es una estructura que anatómicamente queda por detrás de la lente intraocular implantada en el paciente operado de cataratas, para garantizar una buena recuperación visual del mismo es imprescindible la transparencia de esta estructura. Desafortunadamente no son pocos los casos en los que se desarrolla esta anomalía, la opacificación de la cápsula posterior repercute negativamente en el resultado visual del paciente (12). En el campo de la oftalmología esto es conocido como OCP y constituye la complicación tardía más importante en la cirugía de cataratas en los tiempos modernos (13).

#### **1.1.1 Incidencia y prevención.**

La incidencia de la OCP en la actualidad se encuentra entre 0,7 y 47,6 % en los primeros cinco años de la cirugía, según un análisis de 90 estudios publicados. El amplio rango de variación reportado probablemente responda a diferentes criterios de selección de la muestra, edad, técnica quirúrgica, tipo de lente intraocular (LIO) y sistemas de evaluación de la opacidad (2).

Debido a esto, las investigaciones recientes han dirigido su atención a estrategias de prevención de la OCP. Estas incluyen el diseño de los lentes intraoculares, técnicas quirúrgicas modificadas y/o mejoradas, implantes adicionales como los anillos de tensión capsular, el empleo de fármacos y agentes citotóxicos (2).

El conocimiento de los mecanismos etiopatogénicos involucrados en el desarrollo de la opacidad de la cápsula posterior tras la cirugía de catarata ha permitido concluir que para prevenir la aparición de esta complicación se necesita: bloquear la migración de las células epiteliales cristalinas residuales, impedir

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

la proliferación y/o transformación fibroblástica de las LECs o eliminarlas completamente del saco capsular (14).

## 1.2 Sistemas desarrollados para la identificación de OCP.

Los sistemas más usados en la actualidad por el oftalmólogo que le permite visualizar la OCP son los sistemas basados en la lámpara de hendidura, también conocida como biomicroscopio. Es un instrumento de visualización directa de la estructura ocular, pero su limitación radica en que la evaluación del grado de opacidad de la cápsula posterior la hace el médico de manera subjetiva según su apreciación visual, lo que puede variar según la experiencia del cirujano (11). Es por esta razón que algunos autores han desarrollado soluciones informáticas para lograr una evaluación objetiva y reproducible de esta complicación. A continuación se describen algunos de estos sistemas, así como la influencia de la calidad de las imágenes en la obtención de resultados investigativos de elevada calidad.

### 1.2.1 POCO

POCO (Posterior Capsule Opacity): En este sistema las imágenes obtenidas son por retroiluminación, las mismas son evaluadas usando un análisis de los píxeles basado en la diferencia de texturas. Esta evaluación es semiobjetiva, ya que el examinador debe elegir el área afectada y clasificarla según su severidad: el programa divide la zona dentro de la capsulorrexis en 56 pequeños segmentos de igual área, el examinador debe marcar los segmentos que estén opacificados en más del 50% de su área y luego asignar a éstos un nivel de severidad (leve, moderado o severo). El resultado viene dado en forma de porcentaje de opacificación de 0 a 100%. Se calcula la severidad de la opacificación mediante la fórmula (5):

$$[(\text{área de grado 1} \times 1) + (\text{área de grado 2} \times 2) + (\text{área de grado 3} \times 3)] / \text{área total}.$$

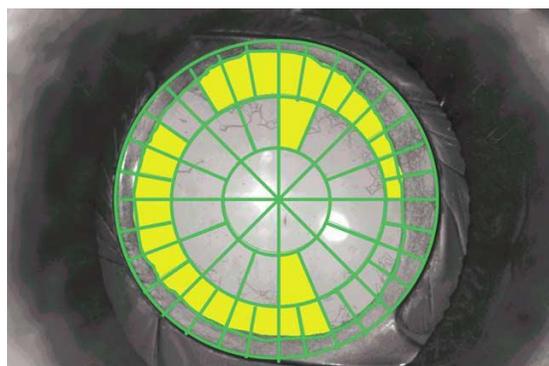


Ilustración 1. Software POCO

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

## 1.2.2 EPCO

EPCO (Evaluación de la opacidad de la cápsula posterior) desarrollado por Tetz y Col, en 1997. Se obtienen imágenes de la lente por retroiluminación, se transfieren al programa, se marca el área a estudiar (algunos autores analizan el área tras la óptica, otros el área dentro de la capsulorrexis anterior y también es posible examinar la zona central comprendida por 3-4 mm), luego se remarcan las zonas opacificadas con el ratón y se clasifican subjetivamente en 4 grupos: mínimo, leve, moderado y severo. El índice de OCP viene dado al multiplicar el grado de opacificación por el área seleccionada. El resultado obtenido es un índice de 0 a 4. Al igual que el software POCO calcula la severidad de la opacificación mediante la fórmula (5):

$$[(\text{área de grado 1} \times 1) + (\text{área de grado 2} \times 2) + (\text{área de grado 3} \times 3)] / \text{área total.}$$

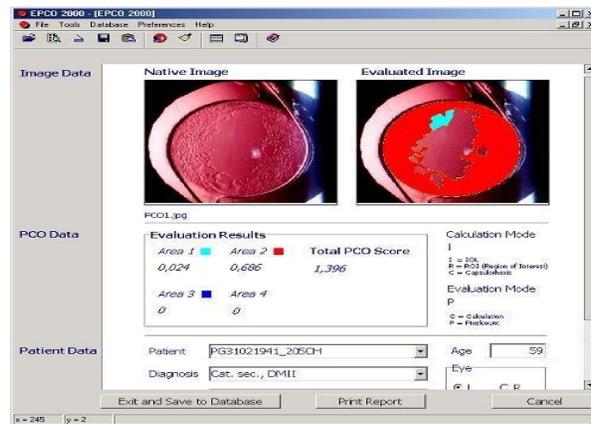


Ilustración 2. Software EPCO

## 1.2.3 AQUA

AQUA (Automated Quantification of After-Cataract): Se obtienen imágenes con retroiluminación y se importan al programa. Se selecciona y analiza el área dentro de la capsulorrexis. Este programa está basado en texturas. Para calcular la no homogeneidad de la imagen se calcula la entropía (grado de desorden) de un mapa de bits. Es totalmente automático y no tiene pasos subjetivos. El resultado que se obtiene es un índice de 0 a 10 (6).

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

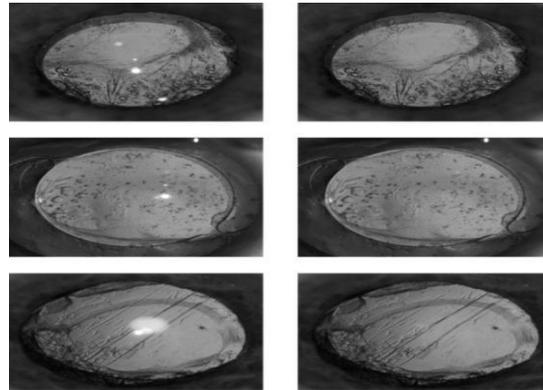


Ilustración 3. Software AQUA

## 1.2.4 AA

Este sistema fue diseñado y programado utilizando la plataforma de programación MatLab. Todo el diseño de la programación y la escritura de código fueron por TM Aslam y para mayor claridad el sistema se conoce por lo tanto como el análisis Aslam (AA) (15). El primer problema encontrado fue uno de iluminación desigual en las imágenes. Incluso con las grandes áreas de reflejos de luz aberrantes retirados, utilizando la fusión con imágenes similares, pero vírgenes, una iluminación de fondo generalmente variables de toda la imagen puede causar errores en el análisis de imágenes. Aunque un sistema para la eliminación de registro basado de imágenes de luz está siendo probado y desarrollado por TM Aslam, todavía se necesita una curva de aprendizaje larga y sigue influyendo el efecto negativo de la iluminación de fondo.

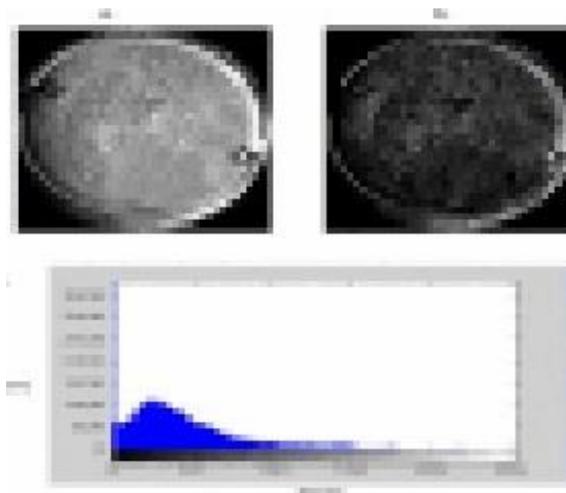


Ilustración 4. Software AA

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

## 1.2.5 Image J

Desarrollado por el National Institute of Health de Estados Unidos. Está basado en los cálculos realizados por Grewal y Jain, quienes determinaron que para los tomogramas del Pentacam un área de 4 mm corresponde a 16 752 píxeles en la imagen (16). Se utilizan las herramientas Find Edges (hallar bordes) y Measure (medición) para calcular el valor promedio de la intensidad de píxeles del área seleccionada en los tomogramas Scheimpflug, como indicador de la opacidad capsular posterior, expresado en unidades (U). Esta herramienta es creada con otros fines, lo que hace que la curva de aprendizaje sea larga, además de que no diferencia los píxeles de fondo a la hora de dar el resultado.

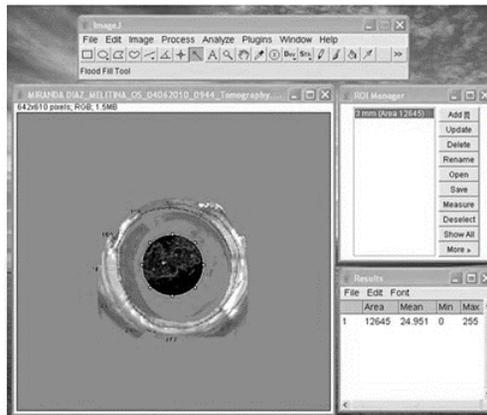


Ilustración 5. Software Image J

## 1.3 Programa Analizador de Opacidad Capsular: PANDOC

El software PANDOC es una herramienta para la cuantificación y análisis de la OCP. Es un software intuitivo con una curva de aprendizaje pequeña, el cual enfoca sus resultados en pos del análisis estadístico que aporten información de alto valor para el estudio y disminución de esta anomalía postoperatoria. Es un software de nuevo tipo, ya que a diferencia del resto toma las imágenes de los tomogramas de Scheimpflug del Pentacam y brinda datos antes no vistos por el especialista como la correlación opacidad/transparencia en el centro y la periferia, el mismo comprende funcionalidades automáticas y semiautomáticas orientadas solo al uso de imágenes de tomogramas del Pentacam, lo cual hace más intuitivo la interfaz actual y la focaliza en su propósito principal (12).

### 1.3.1 Interfaz visual

A continuación se describen los principales elementos que componen la interfaz visual del software PANDOC y las funcionalidades de los mismos, mostrados en la figura:

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

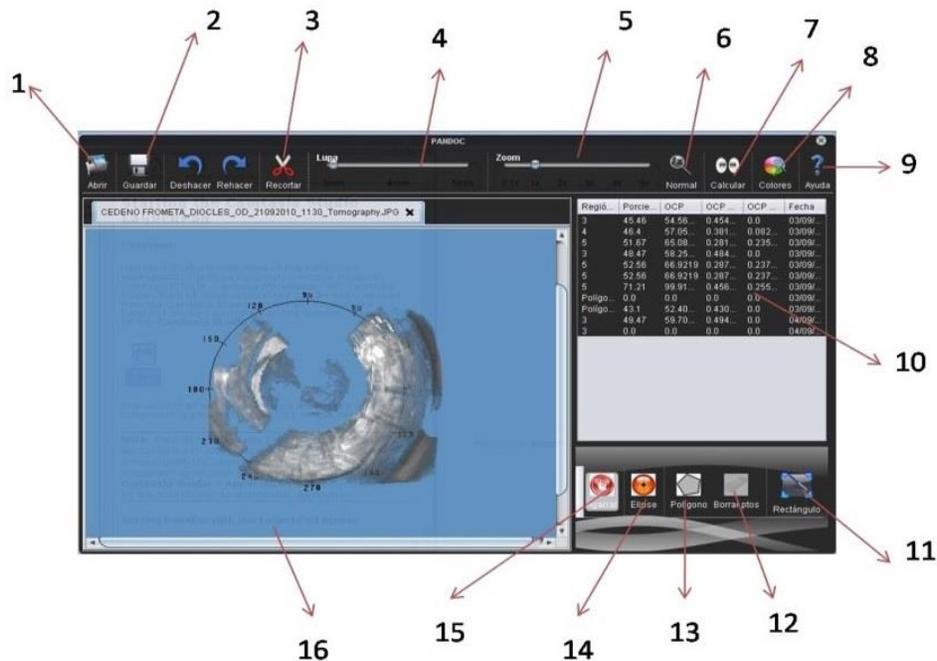


Ilustración 6. Interfaz visual de PANDOC

1. (Botón abrir): Se utiliza para explorar los archivos y cargar la imagen médica resultante del tomograma de Scheimpflug del PENTACAM.
2. (Botón guardar): Se usa para guardar la imagen médica después de haber sido tratada por el programa, obteniendo solo el área de interés.
3. (Botón recortar): Se usa para obtener un segmento de la imagen después de haber sido seleccionado con la herramienta rectángulo.
4. (Herramienta lupa): Se utiliza para seleccionar la dimensión del área circular que será utilizada para el cálculo de la OCP.
5. (Herramienta zoom): Se utiliza para magnificar la imagen en la proporción seleccionada por el usuario.
6. (Botón normal): Se utiliza para llevar a la imagen a su estado original.
7. (Botón calcular): Se utiliza para realizar el cálculo de la OCP en el área de la imagen seleccionada por el usuario previamente.
8. (Botón color): Se utiliza para cambiar el color de las líneas de selección de área.
9. (Botón ayuda): Se utiliza para mostrar la ayuda del software.
10. (Panel resultados): Se utiliza para mostrar al usuario los cálculos realizados sobre la imagen cargada.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

11. (Botón rectángulo): Se utiliza para realizar una selección rectangular de la imagen cargada.
12. (Botón borrar puntos): Se utiliza para borrar el último punto marcado por el usuario en una selección de área poligonal tras marcar el botón polígono.
13. (Botón polígono): Se utiliza para realizar una selección de área poligonal, uniendo los puntos de las referencias marcadas por el usuario mediante clics en la imagen.
14. (Botón elipse): Se utiliza para realizar una selección de área circular, mediante un área predefinida por el tamaño seleccionado en la herramienta lupa.
15. (Botón mano): Se utiliza para mover la imagen a través del panel imagen.
16. (Panel imagen): Se utiliza para mostrar la imagen y la selección de área realizada sobre esta.

## **1.3.2 Herramientas y funcionalidades**

A continuación se describen las principales herramientas y funcionalidades con que cuenta el software PANDOC.

### **Herramientas**

- Corte rectangular. Con esta herramienta el especialista puede seleccionar porciones más pequeñas de la imagen, si es de interés del mismo.
- Zoom. Esta herramienta permite amplificar detalles, lo cual le puede resultar interesante al oftalmólogo que analiza la información contenida en dichas imágenes.
- Polígono. Esta herramienta usando como base el algoritmo Convex Hull, le permite al usuario seleccionar un área personalizada para realizar el cálculo de la opacidad en la misma.
- Elipse. Usando esta herramienta se dibuja una elipse y el usuario puede arrastrar sobre la superficie de la imagen del tomograma del Pentacam, para la posterior cuantificación del área circular que engloba la elipse.
- Lupa. Esta herramienta es la que define el tamaño de la elipse dibujada por la herramienta "elipse", cuyos tamaños pueden ser de 3, 4 o 5mm.

### **Funcionalidades**

- Calcular el grado de la OCP sin incluir los píxeles del fondo. Esta funcionalidad permite que los píxeles de fondo no influyan de forma negativa en la obtención de la intensidad media de los píxeles.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

- Extracción de la intensidad media de píxeles de áreas irregulares de la imagen. Esta funcionalidad amplía la flexibilidad de la aplicación y le da una mayor maniobrabilidad al usuario a la hora de extraer la información que sea relevante al especialista.
- Determinar el centro de la estructura ocular, la cual mediante la identificación de variación de colores en los píxeles y trilateración celular, define automáticamente el centro de la estructura para funcionalidades posteriores.
- Calcular el valor de la OCP. Con esta funcionalidad se calcula el valor de la OCP teniendo en cuenta la intensidad de píxeles de esa área multiplicado por el porcentaje de opacidad de la misma, esta expresa dicho valor también con respecto al centro y a la periferia de la estructura ocular.

## **1.4 Sistemas Basados en el Conocimiento**

Los Sistemas basados en el Conocimiento representan un paso delante de los sistemas de información convencionales al pretender representar funciones cognitivas del ser humano como el aprendizaje y el razonamiento (17). Se pueden definir como aquel programa de ordenador que contiene la erudición de un especialista humano versado en un determinado campo de aplicación (18).

Estos sistemas tienen notables ventajas sobre los expertos humanos. Por una parte, el conocimiento contenido en ellos es más fácil de documentar y de transferir que el de los expertos humanos. Por otra parte, dicho conocimiento es remanente, es decir, permanece tras la desaparición de los expertos, por lo que constituye lo que se ha dado en llamar la memoria institucional del organismo o empresa que lo ha desarrollado. Ahora bien, presentan claras limitaciones al compararlos con los expertos humanos. Ante todo carecen por completo de creatividad y sentido común. Además solo sirven para parcelas bien acotadas de conocimiento frente a la mayor universalidad del conocimiento humano (18).

### **1.4.1 Sistemas Basados en Reglas**

En general los primeros sistemas expertos pertenecían a este tipo de sistemas (19). El proceso de solución de problemas en un sistema basado en reglas (SBR) es crear una cadena de inferencias que constituye un camino entre la definición del problema y su solución. Estos se definen a partir de un conjunto de objetos, que representen las variables del modelo considerado, ligadas mediante un conjunto de reglas, que representarán las relaciones entre las variables (20).

Las reglas permiten representar el conocimiento de la siguiente forma:

**SI** condición **ENTONCES** acción

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

Siendo la parte del **SI** de la regla (también llamada premisa o antecedente) evaluada y en caso de ser cierta, la parte **ENTONCES** (también llamada acción o consecuente) se activa, dando como resultado un nuevo conjunto de hechos.

A pesar de las múltiples ventajas de este tipo de sistemas y que la expresión del conocimiento basado en reglas se aproxima a la forma natural de razonar de los expertos humanos, también presentan inconvenientes, como por ejemplo:

1. Mantenimiento de la coherencia entre las reglas de la base de conocimiento (19):
  - Un encadenamiento infinito, que aparece cuando en la base de conocimiento encontramos reglas del tipo:

**SI A ENTONCES B**

...

**SI B ENTONCES A**

- Problemas de ampliación de la base de conocimientos

En algunas situaciones es necesario realizar una actualización del conocimiento, por ejemplo añadiendo excepciones para un determinado conjunto de reglas o bien incorporando un nuevo conocimiento. En estos casos, y con la finalidad de mantener la coherencia entre las reglas, puede ser necesario incluir un elevado número de reglas, provocando que la base de conocimiento se haga innecesariamente grande. En estas situaciones, puede ser preferible reconstruir la base de conocimiento con el coste que esto implica.

2. Dificultades para retractarse de conclusiones anteriores (19):
  - Este problema viene provocado por el carácter modular y monótono de este tipo de sistemas. Así cuando se cumple la premisa de una regla, nos da licencia para actuar sin tener en cuenta el resto del conocimiento.
3. Opacidad (19):
  - La división de la base de conocimiento en pequeñas reglas tiene como ventaja que cada una de ellas es fácil de usar individualmente, ganando el sistema en modularidad. Sin embargo se tiene

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

que pagar un precio por ello (que en muchos casos es muy elevado) consiste en una pérdida de una perspectiva global sobre el problema que estamos considerando.

## 4. Ineficiencia (19):

- Durante el proceso de inferencia, en cada iteración es necesario chequear cada regla para ver si es aplicable. Este proceso, aun cuando se han hecho avances para tratar de solucionarlo, es altamente costoso.

### **1.4.2 Sistemas Basados en probabilidades**

Desde un punto de vista general, en un sistema experto probabilístico la base de conocimiento está formada por un conjunto de variables  $X_1, \dots, X_n$  y una distribución de probabilidad conjunta sobre ellas  $P(X_1, \dots, X_n)$ . Por otro lado un motor de inferencia será aquel que nos permita actualizar nuestra información sobre determinada variable (o conjunto de ellas)  $X$ , ante la presencia de un conjunto de hechos, evidencias o síntomas determinados  $E$ . En teoría de la probabilidad este motor de inferencia no es más que el cálculo de la probabilidad condicional  $P(X|E)$  (19).

En este tipo de sistemas la adquisición del conocimiento consiste en coleccionar muestras y realizar un procesamiento estadístico que produzca las probabilidades o frecuencias que forman la base de conocimiento. Dentro de sus principales dificultades encontramos:

- No son viables para explicar el razonamiento, ya que los métodos y modelos que utiliza están aún lejos de ofrecer explicaciones comprensibles (21).
- No son factibles para todo tipo de dominio, pues se dificulta construir las redes con ayuda de expertos humanos cuando existen carencias de conocimiento (21).
- El alto número de parámetros que manejan, lo que hace que sea difícil su especificación y definición (22).
- La implementación del motor de inferencia es más complicada y se vuelve costosa computacionalmente en la mayoría de los casos (22).

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

## **1.4.3 Sistemas Basados en Casos**

El enfoque que utilizan los Sistemas Basados en Casos (SBC) para la adquisición de conocimiento es una de las ventajas que se le acreditan a este tipo de sistemas; pues razonan desde episodios específicos, lo cual evita el problema de descomponer el conocimiento del dominio y generalizarlo en reglas o patrones (23).

Las ventajas de los SBC están fundamentadas en la flexibilidad para representar el conocimiento a través de los casos, la organización de la Base de Conocimiento (BC) y de las estrategias de recuperación y adaptación de los casos, además de que el usuario puede ser capaz de agregar nuevos casos a la BC sin la intervención de un experto. Otra importante ventaja que ofrecen estos sistemas es la reutilización de soluciones previas a la hora de resolver un problema, además los SBC pueden almacenar casos que en situaciones pasadas resultaron ser un fracaso, lo que permite advertir sobre problemas potenciales a evitar en situaciones futuras. Otra de las ventajas que ofrecen los SBC consiste en que dichos sistemas modelan mejor el razonamiento propio del ser humano, ya que se basa en experiencias previas para resolver nuevos problemas (24) (25).

Las principales limitaciones de un SBC radican en la definición de la función de semejanza y en lo difícil que resulta encontrar una estructura apropiada para describir el contenido de un caso y decidir cómo la memoria de casos debe ser organizada e indexada para un almacenamiento, recuperación y reutilización efectivo. Otra de las limitaciones de este tipo de sistemas es que la veracidad de su resultado depende de la aplicación de una buena ingeniería del conocimiento (25).

### **Composición**

Una característica distintiva de los sistemas basados en casos (SBC) es la separación del conocimiento (base de conocimiento) del método de solución del problema (máquina de inferencia) La construcción de la base de conocimiento lleva implícito un arduo proceso de adquisición del conocimiento y es particular para cada sistema, por lo que será necesario construirla para cada aplicación. Sin embargo, la máquina de inferencia puede reutilizarse en la construcción de varios SBC siempre que el tipo de conocimiento y el tipo de razonamiento sea similar, ya que la máquina de inferencia es el método implementado que utiliza el conocimiento de la base para resolver los problemas de su dominio (12).

Los componentes fundamentales de un Sistema Basado en Casos son: la base de casos, el módulo de recuperación y el módulo de adaptación (26). A continuación una breve descripción de dichos componentes.

### **Base de Conocimiento (BC)**

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

La BC contiene las experiencias, ejemplos o casos a partir de los cuales el sistema hace sus inferencias. Esta base puede ser generada a partir de casos o ejemplos resultantes del trabajo de expertos humanos o por un procedimiento automático o semiautomático que construye los casos desde datos existentes registrados, por ejemplo, en una base de datos (26).

## Módulo de Recuperación

En este módulo se recuperan de la Base de Casos los casos más semejantes al problema. No existe una medida de semejanza única, general, para cualquier dominio, de ahí que la eficiencia del sistema radica en la función de semejanza que se defina (26).

## Módulo de Adaptación

Después de la determinación de los casos más semejantes, las soluciones contenidas en dichos casos pueden usarse directamente como solución al nuevo problema, pero comúnmente necesitan ser modificadas (26), por lo que se ajustan las soluciones de los problemas similares a los requerimientos del nuevo problema (25).

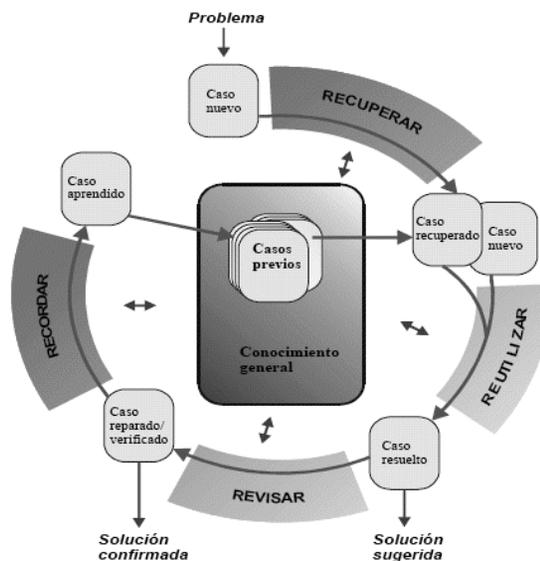


Ilustración 7. Ciclo de vida de un sistema basado en casos (26)

### 1.4.4 Razonamiento basado en casos

Modelo de razonamiento que permite resolver problemas, entender situaciones y aprender utilizando mecanismos de memorización, problemas superpuestos y criterios de optimalidad (27).

Dentro de las técnicas de inteligencia artificial, el Razonamiento Basado en Casos (RBC) es un paradigma de solución de problemas que difiere de otros enfoques y técnicas por su capacidad de utilizar el conocimiento específico adquirido en situaciones previas y utilizarlo en la situación presente. El nuevo

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

problema se resuelve buscando en su memoria, un caso similar a este que haya sido resuelto en el pasado. Además incrementa su conocimiento, almacenando el nuevo caso para ser usado en situaciones futuras (28).

La arquitectura básica de un RBC consiste en una base de casos, un procedimiento para buscar casos similares y un procedimiento de adaptación para ajustar las soluciones de los problemas similares a los requerimientos del nuevo problema (25).

El ciclo de vida de un RBC está formado esencialmente por los cuatro procesos siguientes:

1. Recuperar el caso o casos pasados más similares, pues retoma la experiencia de un problema anterior que se cree es similar al nuevo (23).
2. Reutilizar la información y conocimiento de este caso o casos recuperados para resolver el nuevo problema. Esto consiste en copiar o integrar la solución del caso o casos recuperados (23).
3. Revisar la solución propuesta (23).
4. Guardar la nueva solución una vez ha sido confirmada o validada. Se guardan aquellas partes de la experiencia de una manera tal que sea útil para resolver problemas futuros (23).

En un sistema con RBC se incluirá información acerca de la especificación del problema y atributos del medio que describen el entorno del problema. La descripción del problema debe incluir:

- Las metas que se deben conseguir para resolver el problema.
- Las restricciones de estas metas.
- Las características de la situación del problema y las relaciones entre sus partes.

Otra información de vital importancia que se debe almacenar es la descripción de la solución, que será usado cuando se encuentre una situación similar (25).

Dependiendo de cómo el sistema razone con los casos, esta descripción puede incluir únicamente los hechos que llevan a la solución o información sobre pasos adicionales en el proceso de obtención de la solución. Además la descripción de la solución también podrá incluir (23):

- Las justificaciones de las decisiones tomadas en la solución.
- Soluciones alternativas que no fueron elegidas y porqué
- Soluciones no admisibles y la razón por la que fueron rechazadas.
- Expectativas acerca del resultado de la solución.

Es importante incluir una medida del éxito, si en la base de casos se han logrado soluciones con diferentes niveles de éxito o fracaso. También se debe incluir información acerca del resultado (23).

## *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

- Si el resultado cumple o no con las expectativas y una explicación del por qué.
- Si el resultado fue un éxito o un fracaso.
- Estrategia de reparación.

El conocimiento que almacena un caso es específico, no abstracto, y todo el conocimiento relacionado (es decir, conocimiento aplicable en una circunstancia concreta) se encuentra cerca en la base de casos, esto quiere decir que el conocimiento que necesitemos para resolver un problema específico lo encontraremos agrupado en unos pocos casos o incluso en uno (25).

En los sistemas RBC la base de casos es la memoria de casos almacenados. A la hora de construir la memoria debemos tener en cuenta los siguientes aspectos (23):

- La estructura y representación de los casos.
- El modelo de memoria usado para organizar los casos.
- Los índices empleados para identificar cada caso.

En muchos sistemas RBC no se necesitan almacenar todos los casos existentes, sino que se sigue un criterio para decidir cuáles casos almacenar y cuales descartar.

Una vez elegida la representación de los casos, la elección del modelo de memoria es el siguiente paso. Existen principalmente dos estructuras de memoria, plana y jerárquica (23). En una estructura plana de la base, los casos se almacenan secuencialmente, mientras que en una estructura jerárquica los casos se agrupan en categorías para reducir el número de casos a buscar en una consulta (23) (26).

### **1.4.5 Ingeniería del Conocimiento**

La Ingeniería del Conocimiento es la disciplina orientada a la creación de sistemas basados en conocimiento, para el análisis y solución de problemas de un dominio específico como la visión, el lenguaje y el aprendizaje (17).

Los conocimientos pueden conseguirse de una variedad de fuentes, incluyendo la documentación y los sistemas de información existentes, pero la mayor parte se obtiene de personas expertas. El conocimiento suministrado por el experto, tiene un alto grado de especialización orientada hacia el tema en cuestión.

El Ingeniero de Conocimientos es la persona que obtiene los conocimientos del experto y los representa en el sistema basado en conocimientos. El ingeniero plantea las preguntas al experto, estructura sus conocimientos y los implementa en el sistema.

El primer paso consiste en identificar los problemas que deben ser resueltos por el sistema, es decir, se precisa el ámbito de trabajo. Aquí se incluye ya al usuario, sus requerimientos técnicos y operacionales.

## *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

Una vez delimitado el dominio, hay que extraer y representar los conocimientos del experto. El mismo debe comprobar constantemente si su conocimiento ha sido transmitido de la forma más conveniente. El ingeniero del conocimiento es responsable de una implementación correcta, pero no de la exactitud del conocimiento. La responsabilidad de esa precisión recae en el experto.

La estricta separación entre usuarios, experto e ingeniero del conocimiento no deberá estar siempre presente. Pueden surgir situaciones en las que el experto es también el usuario. Este es el caso cuando existe un tema muy complejo cuyas relaciones e interacciones deben ser determinadas una y otra vez con un gran consumo de tiempo. De esta manera el experto puede ahorrarse trabajos repetitivos. La separación entre experto e ingeniero del conocimiento permanece, generalmente, aunque hay ocasiones en que una persona juega más de un rol.

La representación del conocimiento, es una combinación de estructuras y procedimientos interpretativos que se usan de forma adecuada por un programa, para emular un comportamiento inteligente, que involucra tanto al diseño de varias clases de estructuras para almacenar conocimiento en programas, como el desarrollo del procedimiento que permitan su manipulación inteligente para hacer inferencias.

La principal consideración a la hora de representar el conocimiento consiste en el uso que se le va a dar al mismo, su empleo tiene una estrecha relación con:

- Adquisición de más conocimientos. No sólo tiene que ver con la adquisición de conocimientos sobre algo desconocido, sino también con la ampliación de lo que ya se conoce, pues cuando se sabe de un tema, con más facilidad se aprenden cosas adicionales del mismo, lo que da paso a un proceso incremental.
- Recuperación de conocimiento. Para resolver un problema determinado, es necesario identificar el conocimiento útil para resolver el problema.
- Razonamiento: Para generar soluciones a partir de conceptos establecidos confiables es necesario producir nuevos elementos de conocimiento.

Una técnica efectiva para la adquisición del conocimiento consta de las etapas que se muestran en la Figura 8.

# Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales

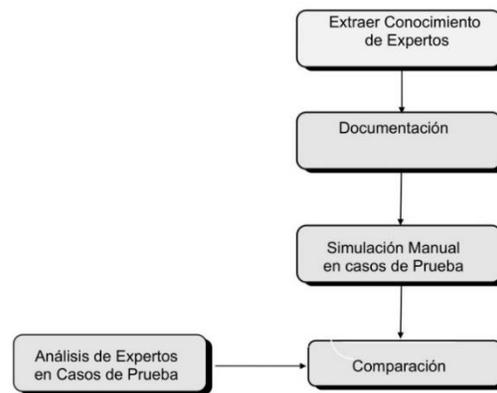


Ilustración 8. Ciclo para la adquisición inicial del conocimiento (17)

Dentro de la rama de la Ingeniería del Conocimiento existen fundamentalmente dos tipos de conocimientos, ellos son Conocimiento Declarativo y Conocimiento Procedural.

## Conocimiento Declarativo

El conocimiento se representa como una colección estática de hechos, acompañada por un conjunto de procedimientos generales para manipularlos. Los esquemas declarativos destacan el concepto de un enunciado simple que establece un elemento de verdad.

Ventajas:

- Cada hecho sólo necesita almacenarse una vez con independencia del número de formas distintas en que puede usarse.
- Es fácil añadir nuevos hechos al sistema sin cambiar los conocidos. Esto proporcionará a los sistemas de este tipo más flexibilidad y economía de recursos y medios, certidumbre en las deducciones y modificabilidad.
- **Transparencia:** El conocimiento representado se almacena en forma explícita y sin ambigüedades. Es relativamente fácil revisar los conocimientos debido a su transparencia.
- **Inferencia directa:** La naturaleza estática y directa de la representación posibilita la inferencia explícita y directa, en forma parecida a la inferencia matemática.
- Todas estas estructuras comparten la noción de que entidades complejas pueden ser escritas como una colección de atributos y sus valores asociados.
- Es flexible y variado en su representación.
- Apropiado para representar y explotar abundante información.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

Desventajas:

- La determinación del grado de detalle al que se debe representar las entidades depende de un adecuado mecanismo de inferencia.
- Necesita de un eficiente administrador.

## **Conocimiento Procedural**

El conocimiento se representa por procedimientos para su posterior uso.

Ventajas:

- Facilidad de representar el conocimiento de cómo hacer las cosas.
- Capacidad de representar el conocimiento que no se ajusta bien en muchos esquemas declarativos simples, tal como sucede con el conocimiento incierto o probabilístico.
- Adecuado en la modelización de acciones generales y cíclicas, capaces de producir resultados homogéneos.

Desventajas:

- Puede gastar muchos recursos en situaciones no deterministas.
- Depende de conocimiento auxiliar para guiar sus actividades.

## **1.5 Bases de Conocimientos**

La BC almacena y representa el conocimiento del dominio del Sistema Experto, facilita su acceso, manipulación y actualización. Es un tipo especial de base de datos para la gestión del conocimiento. Provee los medios para la recolección, organización y recuperación computarizada de conocimiento (17).

El principal beneficio que proveen las BC es que proporcionan medios de descubrir soluciones a problemas que han sido resueltos anteriormente, los cuales podrían ser aplicados como base a otros problemas dentro o fuera de la misma área de conocimiento. La determinación del tipo de información capturada y dónde se encuentra la misma en una BC está dado por los procesos que respaldan al sistema. Una estructura robusta de procesos es la columna vertebral de cualquier BC.

Existen diversas técnicas para la representación del conocimiento, encaminadas a (17):

- Proveer un formato compatible con la computadora.
- Mantener una correspondencia entre el formato de representación y el conocimiento.

## *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

- Establecer una representación que puede ser organizada, recuperada, modificada y actualizada eficientemente.

El conocimiento inicial es generado a partir de los hechos aprendidos y representados, que al ser manipulados por los procesos de búsqueda e inferencia producen nuevos elementos y actualizan otros y de esta forma se genera una nueva versión (17).

Caracterización del conocimiento (17)

- Fidelidad: Es la ausencia de distorsión durante la adquisición del conocimiento al comparar la fuente natural y la de la base de conocimientos.
- Calidad: Grado de cantidad y actualización.
- Temporalidad: Vigencia.
- Particularidad: Dependiente de un contexto o enfoque particular.
- Certidumbre: Veracidad del conocimiento.
- Fuente: Naturaleza y confiabilidad de los emisores del conocimiento.
- Detalle: Grado de desglose.

Las BC también se pueden ver como la evolución lógica de los sistemas de bases de datos tradicionales, en un intento de plasmar no ya cantidades enormes de datos, sino elementos de conocimiento (en forma de hechos y reglas) (29).

A continuación se describen los componentes de una BC (17).

- Entidades: Sujetos, eventos y actividades relevantes a describir del dominio del conocimiento.
- Propiedades: Atributos y valores que describen a las entidades, con base a ellas se establecen los niveles de caracterización del conocimiento.
- Relaciones: Lazos que integran a las entidades y atributos entre sí.
- Jerarquías: Bajo ciertos contextos, las entidades se agrupan en clases, super clases y subclases, permiten agrupar sus atributos y hacer generalizaciones que eviten redundancias en la descripción de un objeto en particular.
- Valores

El valor asociado a los atributos de las propiedades tiene las siguientes características:

- Información acerca del tipo de valor: Numérico, código, índice.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

- Condiciones de validación: Rangos, criterios, asociaciones.
- Fuentes: Para alcanzar, comparar y acreditar el conocimiento.
- Nivel de detalle: Son los elementos del lenguaje que serán tomados en cuenta durante la representación.
- Cobertura
- Generalidades: Se aplican para todas los casos.
- Excepciones: Son excepciones a la regla.
- Permanente: Son afirmaciones que perduran al paso del tiempo.
- Temporales: Acotados por una vigencia

La representación del conocimiento depende de varios criterios:

## 1. Naturaleza

- Descriptiva: rasgos, características, valores, elementos estáticos.
- Procedural: acciones, procedimientos, condiciones, eventos y elementos activos.

## 2. Emisión

- Fuentes: Recolección proveniente de la misma entidad por observación, interrogatorio, análisis.
- Producción: Es el resultado de la operación de ciertas acciones.
- Inferido: Resultado de la aplicación de algún formalismo de deducción.
- Inherente: Es la apropiación de cierto conocimiento de una entidad por otra, gracias a una relación de pertenencia.

## 3. Certidumbre

- Absoluta:
- Incierto:
- Emisión: Al expresarse, obtenerse o generarse.
- Certeza: Aproximación a la veracidad o negación

## 4. Representación

- Reglas.

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

- Estructura de datos.
- Primitivas de la dependencia conceptual

## **1.6 Motor de Inferencias**

Los sistemas de RBC modelan el proceso de razonamiento humano con un componente conocido como el Motor de Inferencias (MI). Este trabaja con la información contenida en la BC para interpretarlos y deducir nuevos casos. Contrasta los casos particulares de la BC con el conocimiento contenido para obtener conclusiones acerca del problema (30). La tarea fundamental del MI es obtener un nuevo conocimiento que no exista explícitamente en la BC, infiriéndolo a partir de lo que sí está representado a través del módulo recuperador de casos y luego obtener la solución mediante el módulo de adaptación. El conjunto de facilidades que proporciona, permite clasificar a los sistemas de RBC en uno de los tres niveles siguientes (24):

- Sistemas consejeros que tan sólo recuperan casos: Sin las facilidades de adaptación, evaluación y reparación. El usuario describe una situación y el sistema devuelve casos previos relevantes de los que el usuario extraerá sus propias conclusiones.
- Sistemas con recuperación y adaptación: Sin evaluación ni reparación. El usuario describe una situación, y el sistema encuentra y adapta casos previos similares.
- Sistemas con recuperación, adaptación y reparación. El usuario describe una situación y el sistema encuentra y adapta casos previos relevantes. Si la solución propuesta falla, el sistema la repara y también modifica el mecanismo de recuperación para evitar el mismo error en el futuro.

El SBC propuesto realiza la recuperación de los casos más semejantes, adaptando la solución en función de estos, lo que permite identificar las estructuras de la imagen resultante de un tomograma de Scheimpflug del Pentacam que representan opacidad en la cápsula posterior del cristalino.

La función del MI dentro de un SBC se puede detallar en una serie de pasos lógicos y secuenciales, empezando por la detección de casos aplicables al nuevo problema que se desea solucionar. Como puede haber más de un caso aplicable, es necesaria la elección de uno de ellos, para lo cual se establece algún criterio de selección que sea capaz de escoger el caso con mayor grado de similitud. En la adaptación se modifica el nuevo caso y se indexa la solución al problema que este plantea, quedando el caso listo para ser almacenado en la BC si fuese necesario, evitando siempre el establecimiento de datos redundantes, lo que vuelve ineficiente la BC (30).

Para lograr darle solución a un nuevo problema, el MI separa las funciones que realiza en 4 etapas (27):

# *Capítulo 1: Identificación de opacidad capsular en pacientes operados de catarata: Bases Conceptuales*

1. Detección de casos aplicables.
2. Elección de los casos más similares.
3. Aplicación.
4. Almacenamiento en la BC.

## **1.7 Conclusiones del capítulo**

En el presente capítulo se realizó un estudio sobre la identificación de opacidad en pacientes operados de catarata, apreciándose la necesidad de la realización de esta tarea de manera automática; además se hizo un estudio de los distintos sistemas existentes para la cuantificación de dicha anomalía. Se llevó a cabo el estudio de los SBC, para tener un mayor entendimiento de este tipo de sistemas y conocer más a fondo el RBC como técnica de la IA empleada. Se describen los conceptos principales de la ingeniería del conocimiento, los diferentes tipos de conocimiento y los elementos que componen el almacenamiento de este en las BC. Además se pudo profundizar en los componentes del MI y las tareas fundamentales que este realiza a la hora modelar el razonamiento humano.

### **Capítulo 2: Herramientas usadas durante el desarrollo**

En este capítulo se describen las herramientas y tecnologías a utilizar durante el desarrollo de la solución. Son expuestos los argumentos que conllevan a la selección de dichas herramientas y tecnologías. Se explican las principales características de las metodologías de software y las ventajas que ofrece la escogida en la presente investigación; se argumenta acerca de las características y beneficios que ofrece el lenguaje de programación seleccionado, el entorno integrado de desarrollo y las herramientas que el mismo brinda. El uso del sistema gestor de base de datos que se propone trae consigo múltiples ventajas que se explicarán con detalles en el presente capítulo, así como los principales aspectos y características de la herramienta escogida para realizar el modelado.

#### **2.1 Metodología de Software**

El desarrollo de software no es una tarea fácil; prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte tenemos aquellas propuestas más tradicionales o robustas que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán, o sea se basa en el uso exhaustivo de documentación durante todo el ciclo del proyecto. Por otra parte tenemos las metodologías ágiles o ligeras las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas (31). Seleccionar una metodología apropiada, resulta en ocasiones un factor esencial para obtener un software con la calidad esperada.

Luego del análisis de las metodologías de desarrollo y el estudio de sus características, etapas de desarrollo y ventajas que posibilitan, se determina utilizar una metodología ágil, dado que la prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software; el cliente es parte del equipo de desarrollo, el equipo de desarrollo es de solo una persona, además de la dificultad para un equipo de desarrollo pequeño el adoptar una metodología robusta a causa de la cantidad de documentación generada y la alta resistencia a los cambios durante el desarrollo.

#### **Metodología de software XP**

Durante la elaboración de la aplicación se utiliza para establecer el control y utilizar un marco de trabajo definido y de probada eficiencia, la metodología de software XP (Programación Extrema), dado que una de las prácticas más significativas que posee, es que con XP es posible simplificar el diseño para agilizar el desarrollo, facilitar el mantenimiento y descartar las ideas que no se necesiten. El ejecutar las pruebas unitarias frecuentemente permite descubrir fallos debido a cambios recientes en el código. XP tiene como

## *Capítulo 2: Herramientas usadas durante el desarrollo*

objetivo fundamental la satisfacción del cliente. Es recomendable utilizar esta metodología debido a que los requisitos tienen altas probabilidades de cambiar con el tiempo y el proyecto cuenta con un grupo pequeño de programadores. Además esta metodología ha sido la utilizada durante el desarrollo del software PANDOC.

Algunas de las características de XP son (32):

- Pruebas unitarias: se basa en las pruebas realizadas a los principales procesos con el objetivo de detectar futuros errores.
- Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.
- La tendencia de entregar software en espacios de tiempo cada vez más pequeños con exigencias de costos reducidos y altos estándares de calidad.

¿Qué nos brinda XP? (32):

- Comienza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- El cliente o el usuario se convierte en parte del equipo.

XP consta de 4 fases:

- Planificación
- Diseño
- Desarrollo
- Pruebas

### **2.2 Lenguaje de Programación**

Para el desarrollo de esta aplicación se escogió como lenguaje de programación a Java, este lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel. La selección de dicho lenguaje se debe a que es puro orientado a objetos, lo que propina una gran reusabilidad, independencia de la plataforma, esto significa que programas escritos

## *Capítulo 2: Herramientas usadas durante el desarrollo*

en el lenguaje Java pueden ejecutarse igualmente en cualquier tipo de hardware (33). Existe además variada documentación y una amplia comunidad sobre este lenguaje y el mismo se caracteriza por ser muy utilizado en la implementación de aplicaciones que incorporan técnicas de inteligencia artificial.

Características fundamentales de Java:

- **Simple:** elimina la complejidad de otros lenguajes como C y se enfoca en el contexto de los lenguajes orientados a objetos.
- **Robusto:** maneja la memoria de la computadora para que el programador no se tenga que preocupar por ello, además de realizar verificaciones en busca de errores, lo mismo en tiempo de compilación que en tiempo de ejecución.
- **Portable:** un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el intérprete de Java, ya que su código compilado es interpretado.
- **Multiproceso:** puede ejecutar diferentes líneas de código al mismo tiempo.
- **Dinámico:** no es necesario que compile todas las clases de un programa para que este funcione. Al efectuar al menos un cambio en alguna de las clases, Java se encarga de realizar un enlace dinámico o una carga dinámica para encontrar las clases.
- **Interpretado:** corre en máquina virtual.

### **2.3 Entorno integrado de desarrollo**

Para el desarrollo de la solución se seleccionó como Entorno de Desarrollo Integrado (IDE, según sus siglas en inglés) el NetBeans 8.0. Es libre y de código abierto; posee todas las herramientas necesarias para crear aplicaciones profesionales de escritorio, empresariales, web y aplicaciones móviles con la plataforma Java. Es conocido por la integración con el lenguaje Java, facilita el desarrollo utilizando funcionalidades como completamiento de código, coloreo de sintaxis, permite la utilización y edición de los componentes visuales de forma sencilla. Incluye también el control de versiones, lo cual representa una ventaja debido a que permite administrar las diferentes versiones del código fuente. Es Además un entorno de desarrollo disponible para varios sistemas operativos como Windows, Mac, Linux y Solaris (33).

### **2.4 Sistema gestor de base de datos**

Como sistema gestor de base de datos (SGBD) se utilizó SQLite, es un SGBD relacional que está contenida en una relativamente pequeña (~500kb) biblioteca en C. Es un proyecto de dominio público creado por Dr. Richard Hipp. A diferencia de los SGBD cliente-servidor, el motor de SQLite no es un

## *Capítulo 2: Herramientas usadas durante el desarrollo*

proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones (34).

Esto reduce la latencia en el acceso a la base de datos (BD), debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la BD (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de BD al principio de cada transacción. Este SGBD dispone de una completa interfaz orientada a objetos, con distintas funciones que facilitan la manipulación de datos y posee funciones muy similares a las que se pueden manejar con MySQL.

Combina el motor y la interfaz de la BD en una única biblioteca, y almacena los datos en un único archivo de texto plano. Esto hace que cada usuario pueda crear tantas BD como desee sin la necesidad de la intervención de un administrador de BD que gestione los espacios de trabajo, usuarios y permisos de acceso

La potencia de SQLite se basa fundamentalmente en la simplicidad, lo que hace que no sea una buena solución en entornos de tráfico muy elevado y/o alto acceso concurrente a datos, además encapsula toda la BD en un único fichero. El hecho de almacenar toda la BD en un único archivo, facilita la portabilidad de los datos, y solamente tiene la restricción del espacio de disco asignado.

Se puede utilizar SQLite de dos formas:

- Como gestor de base de datos local en un computador. De esta forma, se pueden gestionar BD con SQLite igual que si se estuviese trabajando con un sistema gestor de base de datos como MySQL, sin necesidad de instalar nada, ya que SQLite se compone de un único archivo ejecutable.
- Como una extensión más de PHP, utilizando las funcionalidades de SQLite configuradas, o bien como módulo de PHP, o como biblioteca; sin necesidad de tener instalado o conectar con un servidor de base de datos. Ofrece una rápida interfaz de la BD almacenada en el archivo de texto plano.

Varios procesos o hilos pueden acceder a la misma BD sin problemas. Varios accesos de lectura pueden ser servidos en paralelo. Un acceso de escritura sólo puede ser servido si no se está sirviendo ningún otro acceso concurrentemente. En caso contrario, el acceso de escritura falla devolviendo un código de error (o puede automáticamente reintentarse hasta que expira un timeout configurable). Esta situación de acceso concurrente podría cambiar cuando se está trabajando con tablas temporales. Sin embargo, podría producirse un deadlock debido al proceso multihilo.

SQLite Admin es un administrador de bases de datos SQLite que básicamente permite (35):

## *Capítulo 2: Herramientas usadas durante el desarrollo*

- Abrir, crear, compactar bases de datos.
- Crear, renombrar y eliminar tablas vacías.
- Crear, renombrar y eliminar columnas.
- Crear, modificar y eliminar registros.
- Encriptar o reencriptar bases de datos
- Correr sentencias SQL personalizadas.
- Crear y eliminar índices, Triggers y Cursores.

### **2.5 Herramienta para el modelado**

Como herramienta para el modelado de la solución se ha seleccionado Visual Paradigm for UML (VP-UML v8.0), una de las líderes del mercado de las llamadas herramientas de Ingeniería de Software Asistida por Computadora (CASE, según sus siglas en inglés).

VP-UML v8.0 soporta los principales estándares de la industria tales como el Lenguaje de Modelado Unificado (UML, según sus siglas en inglés), SysML, BPMN, XMI, entre otros. Ofrece un conjunto completo de herramientas, brindando a los equipos de desarrollo de software todo lo necesario para la captura de requisitos, planificación de software, planificación de controles, modelado de clases y modelado de datos.

Ofrece interoperabilidad entre diagramas ya que es capaz de exportar los diagramas de un modelo a otro con mucha facilidad, ahorrando de esta manera tiempo, lo cual es crucial para el desarrollo. Hace posible la generación de código Java desde los diagramas

### **2.6 Conclusiones del capítulo**

Durante el desarrollo del capítulo fueron seleccionadas y descritas las herramientas y tecnologías a ser usadas en el desarrollo de la propuesta de solución. El estudio realizado sobre metodologías de desarrollo permitió arribar a la conclusión de que XP, debido a sus características, es la metodología más apropiada para guiar el desarrollo del sistema. El estudio realizado sobre herramientas y tecnologías para el desarrollo de software, permitió seleccionar Java como lenguaje de programación, NetBeans como IDE, SQLite como SGBD y Visual Paradigm como herramienta para el modelado.

## *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

### **Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam**

En el presente capítulo se realiza una descripción detallada de la solución propuesta; además del modelado y los pasos a seguir durante el desarrollo de dicha solución. Se exponen los artefactos generados propios de la metodología de desarrollo utilizada durante la implementación de la propuesta, tales como: historias de usuario y tarjetas CRC.

#### **3.1 Fase de Planificación**

La metodología XP define como fase inicial del desarrollo de software la planificación. Durante el transcurso de esta etapa se realiza el proceso de identificación y elaboración de las historias de usuario (HU), además el equipo de trabajo se familiariza con las tecnologías y herramientas seleccionadas para el desarrollo. El cliente define el nivel de prioridad con que se deben implementar las HU, así como la estimación del esfuerzo que costará implementar las mismas. El resultado de la presente fase es un plan de entregas donde se realiza una estimación de las versiones que tendrá el producto en su elaboración, de forma tal que sea una guía durante el desarrollo (36).

##### **3.1.1 Historias de Usuario**

Entre los artefactos que define la metodología XP se encuentran las HU que son utilizadas para especificar las funcionalidades que brindará el sistema. Cada HU es una representación de un requerimiento de software escrito en una o dos frases, utilizando el lenguaje común del usuario. Representan una forma rápida de administrar los requerimientos de los usuarios, sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos (36). De acuerdo a la prioridad asignada por el cliente a cada HU y teniendo en cuenta la complejidad y riesgo determinado por el programador, se realiza la estimación de cada una de las identificadas. La unidad de estimación es el punto, un punto equivale a una semana ideal de programación.

Como resultado del trabajo realizado durante la fase se identificaron un total de 13 HU, a continuación se muestran algunas de ellas.

*Tabla 1. Historia de usuario # 1*

Historia de usuario	
<b>Número:</b> 1	<b>Nombre:</b> Construir BC

## *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se declaran los aspectos necesarios para la construcción de la base de conocimientos	
<b>Observaciones:</b>	

*Tabla 2. Historia de usuario # 7*

Historia de usuario	
<b>Número:</b> 7	<b>Nombre:</b> Definir región del caso
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se definen las características necesarias para la identificación de la región en la cual se encuentra el caso que está siendo analizado.	
<b>Observaciones:</b>	

### 3.1.2 Estimación de esfuerzos por Historias de Usuario

Para el desarrollo satisfactorio de la solución propuesta, se realizó una estimación de esfuerzo para cada una de las HU, arrojando los siguientes resultados:

*Tabla 3. Puntos de estimación por historias de usuario*

No	Historias de usuario	Puntos de estimación
1	Construir la BC	2
2	Conectar la BC con la aplicación	1
3	Reconocer imagen de tomograma	2
4	Reconocer el centro de la estructura ocular	2
5	Llenar la BC	3
6	Obtener el nuevo caso	2
7	Definir región del caso	2

## Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam

8	Cargar Casos similares de la BC	2
9	Calcular similitud de rasgos	2
10	Calcular similitud de casos	2
11	Calcular certeza de solución	2
12	Brindar datos al usuario	2
13	Recuperar el caso a la BC	2

### 3.1.3 Plan de iteraciones

Luego de identificar y definir las HU y estimar el esfuerzo propuesto para la realización de cada una de ellas, se precisa establecer el contenido de trabajo de las mismas y es aquí donde se establece el plan de iteraciones, regulando la cantidad de HU a implementar dentro del rango establecido por la estimación efectuada. Tomando como referencia los aspectos antes tratados la aplicación que se pretende construir se desarrollará en 4 iteraciones, explicadas más detalladamente a continuación:

#### Iteración 1

La iteración tiene como finalidad implementar las HU que se consideraron más necesarias atendiendo a su relevancia e impacto. En esta iteración se construye la BC y se conecta con la aplicación; además se implementan las funcionalidades que permiten reconocer imagen de tomograma y reconocer el centro de la estructura ocular.

#### Iteración 2

En esta iteración se realizan todas las HU relacionadas con la ingeniería y gestión del conocimiento, se llena la BC, se realiza la obtención del nuevo caso, además de definir la región del caso.

#### Iteración 3

En esta iteración se realizan las HU que definen la mayor parte del motor de inferencias, ya que donde se implementan los elementos y funciones de similitud, definiendo también como cargar los casos similares de la BC, calcular similitud de rasgos y calcular similitud de casos.

#### Iteración 4

En esta iteración se realizan todas las HU relacionadas con la interacción con el especialista y el método de recuperación definido en el motor de inferencias, aquí se calcula la certeza de solución, se le brindan los datos al usuario y se realiza la recuperación del caso a la BC.

## *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

A modo de resumen se presenta a continuación la tabla que muestra las 4 iteraciones analizadas previamente con las HU que incluyen y su duración:

*Tabla 4. Plan de duración de las iteraciones*

Iteraciones	Historias de usuario	Duración
Iteración 1	Construir la BC	7
	Conectar la BC con la aplicación	
	Reconocer imagen de tomograma	
	Reconocer el centro de la estructura ocular	
Iteración 2	Llenar la BC	7
	Obtener el nuevo caso	
	Definir región del caso	
Iteración 3	Cargar Casos similares de la BC	6
	Calcular similitud de rasgos	
	Calcular similitud de casos	
Iteración 4	Calcular certeza de solución	6
	Brindar datos al usuario	
	Recuperar el caso a la BC	

### 3.1.4 Plan de entrega

El plan de entregas tiene como objetivo definir el número de liberaciones que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una.

*Tabla 5. Plan de entrega*

	Iteración 1	Iteración 2	Iteración 3	Iteración 4
<b>Cantidad de HU</b>	4	3	3	3
<b>Fecha de entrega</b>	14/01/2015	4/03/2015	15/04/2015	27/05/2015

### 3.1.5 Implementación de la propuesta de solución

A continuación se hace referencia a los principales elementos que componen el sistema basado en casos que se propone. Siguiendo los planteamientos de la metodología XP debe ser un diseño simple y de fácil interacción, basándose fundamentalmente en el desarrollo de las tarjetas Clases, Responsabilidad y

### Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam

Colaboración (CRC). También se detallan las cuatro iteraciones llevadas a cabo, así como las tareas generadas por cada HU.

#### 3.1.6 Tarjetas CRC

La metodología XP en lugar de utilizar diagramas para desarrollar modelos representa las clases mediante tarjetas. Las tarjetas CRC (Clase, Responsabilidad y Colaboración) ayudan al equipo a definir actividades durante el diseño del sistema. Estas tarjetas permiten trabajar con una metodología basada en objetos, permitiendo que el equipo de desarrollo completo contribuya en la tarea del diseño. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda y las clases que se implican en cada responsabilidad a la derecha.

Tabla 6. Tarjeta CRC de la clase Editor

Clase: Editor	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"><li>1. Permite realizar las operaciones con la base de conocimientos.</li><li>2. Realizar el escaneo automático.</li><li>3. Obtiene la información relacionada con el pixel.</li><li>4. Realizar el cálculo de la OCP.</li></ol>	<ol style="list-style-type: none"><li>1. Conexión</li><li>2. Configuración</li><li>3. Pixel</li><li>4. Lienzo</li></ol>

#### 3.1.7 Patrones de diseño

Un patrón de diseño es una descripción de clases y objetos que se comunican entre sí, adaptada para resolver un problema general de diseño en un contexto particular permitiendo una solución a un problema en un contexto determinado. Cada patrón describe un problema que ocurre una y otra vez en el entorno, describe también el núcleo de su solución, de forma que puede utilizarse varias veces sin repetirse

En el diseño de la aplicación propuesta se utilizan los Patrones Generales de Software para Asignar Responsabilidades (GRASP), los cuales se mencionan a continuación.

- Experto: consiste fundamentalmente en asignar las responsabilidades a aquellos objetos o clases que cuenta con la información necesaria para cumplir la responsabilidad. Esto se evidencia en la clase Lienzo, ya que se encarga de distribuir las responsabilidades a cada una de las clases correspondientes.

## *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

- Bajo acoplamiento: cada clase en el sistema depende solo de las clases necesarias para su implementación. No existe una sobrecarga de dependencia entre las clases. Esto se evidencia en clases que no dependen de otras, como por ejemplo Geometría\_Computacional.
- Controlador: El patrón "controlador" establece una clara separación entre la interfaz de usuario y el corazón o núcleo de procesamiento de la aplicación, donde se halla la lógica de negocios. Esto se evidencia en la clase Lienzo ya que la misma es la clase intermedia entre las clases Recortador y Editor las encargadas de la interfaz de usuario y el núcleo de la aplicación respectivamente.

### **3.2 Componentes de los sistemas basados en casos**

Un sistema basado en casos consta de tres componentes principales: una BC, un analizador de problemas y un recuperador de casos; estos dos últimos conforman el MI (37). A partir de dichos componentes se enuncian un conjunto de pasos para el desarrollo del mismo, con el objetivo de la identificación de opacidad en pacientes operados de catarata.

Secuencia de pasos para el desarrollo (37):

1. Definir los rasgos predictores y los rasgos objetivos.
2. Determinar el dominio de definición de cada rasgo.
3. Determinar el peso informacional de cada rasgo.
4. Definir las funciones de comparación de rasgos.
5. Definir las funciones de comparación de casos.
6. Representar los casos.
7. Obtener el nuevo caso a resolver.
8. Comparar el nuevo caso con la Base de Casos.
9. Obtener los 'k' casos más semejantes.
10. Seleccionar el rasgo objetivo, siendo este el paso fundamental para la solución.

# Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam

## 3.2.1 Definición de los rasgos predictores y los rasgos objetivos

Los rasgos predictores se han definido en base a los conocimientos brindados por los especialistas, teniendo en cuenta su criterio de evaluación de la opacidad. Donde cada pixel será un nuevo caso a resolver teniendo en cuenta sus rasgos predictores y rasgo objetivo.

Se comienza por expresar la importancia de la región a evaluar dentro de la estructura ocular. Obteniéndose el rasgo: **Región**, la misma estará enfocada entre los 3 y 5 mm en correspondencia con el centro de la estructura ocular. La imagen se dividirá en 32 regiones equivalentes, lo que permitirá comparar la región que se encuentre en análisis, minimizando el espacio de búsqueda con el objetivo de encontrar las semejanzas entre casos.

Cada pixel será tratado como un objeto o caso, siendo importante la composición RGB de colores primarios Rojo, Verde y Azul de cada uno de ellos, denotado RGB dado sus siglas en inglés Red-Green-Blue, este es otro dato comparable con significatividad para el problema en cuestión. Surge así el rasgo: **Intensidad**, este resultado está dado por el promedio de la intensidad de los colores primarios que componen el pixel.

La cercanía del objeto al centro de la estructura ocular es de suma importancia a la hora de detectar opacidad, por lo que se obtiene el rasgo: **Cercanía al centro**. El valor de este rasgo está dado por el resultado del cálculo de la distancia euclidiana entre el punto y el centro de la estructura ocular a analizar.

En la descripción resalta además el comportamiento de los píxeles vecinos dentro de la imagen. Por lo que conocer si se encuentra aislado o si se encuentra dentro de una nube de opacidad, es significativo para la evaluación de si corresponde con un píxel opaco, o es parte de la estructura ocular o simplemente es un píxel azul de fondo. Siendo necesarios adquirir los rasgos: **Vecindad** e **Intensidad de los vecinos**. El primero es analizado como un número binario de 8 cifras, representando cada elemento los 8 píxeles vecinos a analizar (1-Si presenta variación el pixel y 0- en caso que sea un píxel de fondo o parte de la estructura ocular).



Ilustración 9. Representación del rasgo "Vecindad"

### Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam

Como se muestra en la figura, el número binario es representado así: ABCDEFGH, siendo X el pixel analizado y el resto, los pixeles vecinos. El segundo rasgo resultante está dado por el promedio de la intensidad de la composición de los colores primarios que componen los pixeles vecinos.

Luego de definidos los rasgos predictores se define el rasgo objetivo: **Opacidad**. Siendo este el resultado del análisis final de si el objeto (pixel) representa o no opacidad dentro de la estructura ocular

#### 3.2.3 Dominio de definición de cada rasgo

En este paso se procede a determinar el dominio de definición de cada rasgo, expresados en la siguiente tabla:

Tabla 7. Dominio de definición de cada rasgo

Rasgo	Tipo de Valor	Dominio
Región	Entero	[1,...,32]
Intensidad	Entero	[100,...,230]
Cercanía al centro	Entero	[75,...,120]
Vecindad	Binario	[00000000,...,11111111]
Intensidad de los vecinos	Entero	[0,...,255]
Opacidad	Binario	[0,1]

#### 3.2.4 Peso informacional de cada rasgo

Luego de tener los rasgos y el dominio de definición de los mismos, se hace necesario definir el peso informacional que posee cada uno de ellos para el problema. Estos pesos fueron definidos teniendo en cuenta la información que brinda para el análisis cada uno de los rasgos A continuación se muestra la conformación de dichos pesos:

Tabla 8. Peso informacional de cada rasgo

Rasgo	Peso informacional {1,...,10}
Región	8
Intensidad	5
Cercanía al centro	10
Vecindad	5
Intensidad de los vecinos	2

## *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

### **3.2.5 Funciones de comparación de rasgos**

A continuación se muestran las funciones para evaluar la semejanza de los rasgos que han sido extraídos del nuevo caso con los casos existentes en la BC, las cuales fueron tomadas del libro “Enfoque Lógico Combinatorio al Reconocimiento de Patrones” (38).

Siendo la función  $C_s(X_s(O_i), X_s(O_j)) = \begin{cases} 1 & \text{Si } X_s(O_i) = X_s(O_j) \\ 0 & \text{en otro caso} \end{cases}$  un concepto de igualdad, donde  $(O_i)$  es el valor del rasgo  $X_s$  en el objeto  $O_i$ . Aplicable al rasgo: Región y Vecindad.

La función  $C_s(X_s(O_i), X_s(O_j)) = \begin{cases} 1 & \text{Si } X_s(O_i), X_s(O_j) \in [A_p, A_{p+1}] \\ 0 & \text{en otro caso} \end{cases}$  es un concepto de intervalo, donde  $X_s(O_i)$  es el valor del rasgo  $X_s$  en el objeto  $O_i$  y  $A_p, A_{p+1}$  son los valores iniciales y finales del intervalo. Donde el conjunto de los mismos sería  $M_{int} = [100,140] \cup [141,180] \cup [181,230]$ , Siendo esta aplicable al rasgo Intensidad de la siguiente forma:

1. [100,140]      Intensidad Baja
2. [141,180]      Intensidad Media
3. [181,230]      Intensidad Alta

Y la función  $C_s(X_s(O_i), X_s(O_j)) = \begin{cases} 1 & \text{Si } (X_s(O_i) - X_s(O_j)) < \varepsilon_s \\ 0 & \text{en otro caso} \end{cases}$  es un concepto de semejanza asociado a un umbral, donde  $X_s(O_i)$  es el valor del rasgo  $X_s$  en el objeto  $O_i$  y  $\varepsilon_s$  es el umbral asociado al rasgo  $X_s$ . Es aplicable a los rasgos: Intensidad de los vecinos y Cercanía al centro con los siguientes valores de umbral:

Intensidad de los vecinos:  $\varepsilon_s = 10.0$

Cercanía al Centro:  $\varepsilon_s = 0.1$

### **3.2.6 Funciones de comparación de casos**

Después de definidas las funciones de comparación de rasgos se procede a comparar el objeto (caso) nuevo con los existentes en la BC. La función de semejanza que se utiliza para la comparación de dichos casos, tomada del libro “Enfoque Lógico Combinatorio al Reconocimiento de Patrones” (38) es:

$$\beta(\Omega(O_i), \Omega(O_j)) = (\sum_{X_i \in S} P(X_i) / \sum_{X_i \in \Omega} P(X_i))$$

Donde  $(X_i)$  es una magnitud asociada a cada rasgo  $X_i$  que refleja su relevancia (Peso Informacional),  $S$  es el conjunto de rasgos coincidentes entre los objetos  $(O_i, O_j)$ ,  $\Omega$  es el conjunto total de rasgos que se consideran y  $\Omega(O_i)$  es el conjunto de rasgos del objeto  $O_i$ .

### *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

Esta función es utilizada debido a que todos los rasgos no contienen el mismo peso informacional (no aportan la misma información) y su medida de semejanza viene dada por el promedio de los pesos asociados a cada una de las variables, que en las respectivas descripciones de los objetos que se comparan resultaron ser semejantes. Obteniéndose un valor porcentual de la semejanza de los mismos atendiendo al peso informacional de cada rasgo.

#### **3.2.7 Representación de los casos**

Esta etapa de almacenamiento consiste en registrar en la base de conocimiento la información derivada del nuevo caso.

El almacenamiento de los casos se realiza en una base de casos, la cual se nutre de los casos nuevos proporcionados por el sistema y los casos previamente almacenados en ella. Coexistiendo entonces los conocimientos, las experiencias brindadas por los especialistas y las recreadas por el sistema. En esta BC se almacena el conocimiento necesario para resolver los problemas del dominio de aplicación.

El conocimiento se representa a través de una tabla en la cual las columnas son etiquetadas por variables que representan los rasgos predictores y el rasgo objetivo (decisión) y las filas representan los casos. Para la representación de los casos, además de las columnas descritas existirá una columna con una variable (Valor Global) que mediante un valor numérico hace énfasis en la descripción general de cada uno de los rasgos predictores del caso en cuestión. Esta variable nos beneficia en la organización de la base de casos en función de su valor.

#### **3.2.8 Obtener el nuevo caso a resolver**

La obtención del nuevo caso a resolver se desencadena cuando el usuario del sistema carga la imagen médica resultante de un tomograma Scheimpflug del PENTACAM. Los casos están vinculados a cada pixel de la imagen cuya variación de colores sea distinta a la del fondo y se encuentre entre los 3 y 5 mm de radio (área de interés donde se encuentra la opacidad) con respecto al centro de la estructura ocular, siendo cada uno de estos un nuevo caso a dar solución.

#### **3.2.9 Comparación del nuevo caso con la Base de Casos**

En la base de casos todos los datos estarán organizados de menor a mayor por el valor de la variable "Valor Global", de forma que para comparar el nuevo caso con los existentes en la base de casos, se seleccionan de la base de conocimiento aquellos valores cuya descripción se ajusta más a la información presentada en el nuevo caso. A estos casos más cercanos al valor del nuevo se le aplican las funciones

### Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam

de comparación de rasgos para comparar cada uno de los rasgos de los mismos, y después se utiliza la función de comparación de casos para comparar finalmente los casos, definiendo posteriormente el valor del rasgo objetivo de este nuevo caso.

#### 3.2.10 Obtención de los ‘k’ casos más semejantes

Después de la selección de los casos más semejantes de la base de casos y la comparación cada uno de estos con el nuevo caso, se procede a escoger los “k” casos más semejantes, para lo que se propone usar (k=5) siendo la comparación en un espectro de casos más amplio demasiado costosa computacionalmente.

#### 3.2.11 Selección del rasgo objetivo

Para la selección del resultado del rasgo objetivo, revisaremos los “k” más semejantes ordenándolos de menor a mayor según su valor de semejanza. Después aplicamos la siguiente fórmula para obtener el resultado:

$$\theta(Ck) = \begin{cases} 1 & \text{Si } \left( \sum_{x=1}^k (Vn(Cx) * x) * V \right) > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Donde  $Vn(Cx)$  es una magnitud asociada a cada caso  $Cx$  develada por el resultado de su función de comparación  $\beta(\Omega(Oi), \Omega(Oj))$  con el nuevo caso,  $K$  es el conjunto de los casos con mayor valor de coincidencia,  $V$  es un valor que denota la presencia de opacidad en el caso dado por el rasgo objetivo Opacidad del caso almacenado, el cual toma valor 1 si es opacidad y -1 en caso de ausencia de la misma, y el valor de  $\theta(Ck)$  nos representa la presencia o la ausencia de opacidad en el caso en cuestión, representando este el resultado del rasgo objetivo.

### 3.3 Conclusiones del capítulo

En este capítulo se ha presentado el ciclo de vida de la solución propuesta según la metodología XP, generando las HU que caracterizan al sistema, así como la estimación del esfuerzo necesario para la implementación de las mismas. Además se construyó el plan de iteraciones, se realizó la estimación del tiempo que requiere la implementación de cada una y se llevó a cabo el plan de entregas en el cual se define qué historias de usuario formarían parte de cada entrega. Fueron realizadas las tarjetas CRC

### *Capítulo 3: Identificación de la Opacidad de la Cápsula Posterior en imágenes del Pentacam*

correspondientes a cada clase y quedan definidos cada uno de los pasos realizados durante el desarrollo del SBC.

## **Capítulo 4: Resultados y validación del sistema**

Las pruebas de software constituyen un instrumento para determinar el nivel de calidad de un producto. Dentro de las fases propuestas por la metodología XP se lleva a cabo la implementación y las pruebas del sistema. Dicha metodología divide las pruebas en dos grupos: pruebas unitarias, encargadas de verificar el código, la cual es diseñada por los programadores, y pruebas de aceptación destinadas a evaluar si al terminar una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final (39). Durante las iteraciones las historias de usuarios seleccionadas serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. Las pruebas unitarias se realizan con el objetivo de verificar el correcto funcionamiento de los procedimientos, agrupados en los distintos componentes.

### **4.1 Fase de implementación**

Se especifica en esta fase la implementación de las HU en su correspondiente iteración, obteniéndose en cada una de ellas una versión funcional del producto. Lo primero es hacer un chequeo de cada HU, en conjunto con el plan de iteraciones y se modifica en caso de ser necesario, para esto se crean tareas de desarrollo, para de esta forma poder organizar la implementación. Estas tareas, al contrario de las HU escritas en el lenguaje del cliente, son escritas en un lenguaje técnico. Como parte de la planificación realizada en el capítulo anterior se detallan a continuación las iteraciones de desarrollo sobre el sistema.

#### **4.1.2 Iteración 1**

En esta iteración se implementaron las funcionalidades de mayor prioridad para el sistema:

*Tabla 9. Tiempo de implementación de las HU de la iteración #1*

<b>Historias de usuario</b>	<b>Tiempo de implementación</b>	
	<b>Estimación</b>	<b>Real</b>
Construir la BC	2	2
Conectar la BC con la aplicación	1	0.5
Reconocer imagen de tomograma	2	1.5
Reconocer el centro de la estructura ocular	2	2

A continuación se muestran las tareas de ingeniería efectuadas para las funcionalidades implementadas en esta iteración:

## Capítulo 4: Resultados y validación del sistema

Tabla 10. Tarea de ingeniería de la funcionalidad Crear BC

Tarea	
Número de tarea: 1	Número de HU: 1
Nombre: Crear la BC	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 2
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se declaran todos los elementos necesarios para la creación de la base de conocimientos.	

Tabla 11. Tarea de ingeniería de la funcionalidad Conexión

Tarea	
Número de tarea: 2	Número de HU: 2
Nombre: Conectar la BC con la aplicación	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se conecta la BC con el software PANDOC.	

Tabla 12. Tarea de ingeniería de la funcionalidad Reconocer imagen de tomograma

Tarea	
Número de tarea: 3	Número de HU: 3
Nombre: Reconocer imagen de tomograma	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se declaran los elementos necesarios para reconocer un tomograma.	

Tabla 13. Tarea de ingeniería de la funcionalidad Reconocer centro de la estructura ocular

Tarea	
Número de tarea: 4	Número de HU: 4
Nombre: Reconocer el centro de la estructura ocular	

## Capítulo 4: Resultados y validación del sistema

Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se declaran los elementos necesarios para reconocer el centro de la estructura ocular con la utilización de trilateración celular.	

### 4.1.3 Iteración 2

*Tabla 14. Tiempo de implementación de las HU de la iteración # 2*

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Llenar la BC	3	3
Obtener el nuevo caso	2	2
Definir región del caso	2	1.5

*Tabla 15. Tarea de ingeniería de la funcionalidad Llenar BC*

Tarea	
Número de tarea: 5	Número de HU: 5
Nombre: Llenar la BC	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 2.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se llena la BC con la mayor cantidad de casos de imágenes existentes que presentan opacidad.	

*Tabla 16. Tarea de ingeniería de la funcionalidad Obtener nuevo caso*

Tarea	
Número de tarea: 6	Número de HU: 6
Nombre: Obtener el nuevo caso	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Una vez que se carga en el sistema un tomograma cada pixel que se encuentra entre los 3 y los 5 mm del centro de la estructura ocular va a significar un nuevo caso.	

## Capítulo 4: Resultados y validación del sistema

Tabla 17. Tarea de ingeniería de la funcionalidad Buscar región

Tarea	
Número de tarea: 7	Número de HU: 7
Nombre: Buscar región.	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se implementa el método Buscar_Region, que divide la imagen en 32 partes equivalentes lo que permite identificar la región a la que pertenece el nuevo caso a resolver y de esta forma minimizar el rango de búsqueda.	

### 4.1.4 Iteración 3

Tabla 18. Tiempo de implementación de las HU de la iteración # 3

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Cargar Casos similares de la BC	2	1.5
Calcular similitud de rasgos	2	2
Calcular similitud de casos	2	2

Las tareas de ingeniería generadas por la iteración 3 se encuentran en el apartado “Anexo # 3”.

### 4.1.5 Iteración 4

Tabla 19. Tiempo de implementación de las HU de la iteración # 4

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Calcular certeza de solución	2	2
Brindar datos al usuario	2	1.5
Recuperar el caso a la BC	2	2

Las tareas de ingeniería generadas por la iteración 4 se encuentran en el apartado “Anexo # 4”.

## 4.2 Pruebas

La metodología ágil XP divide las pruebas en dos grupos: pruebas unitarias y pruebas de aceptación. Las pruebas unitarias son desarrolladas por los programadores y se encargan de verificar el código

## Capítulo 4: Resultados y validación del sistema

automáticamente y las pruebas de aceptación están destinadas a verificar que al final de cada iteración las historias de usuario cumplan con la funcionalidad asignada y satisfagan las necesidades del cliente.

### 4.2.1 Pruebas de caja blanca

El método de pruebas de caja blanca precisa del acceso al código del programa de modo que se pueda comprobar su lógica interna. Al sistema desarrollado se le aplicaron tanto pruebas unitarias como de aceptación, tal y como propone XP.

#### Prueba del camino básico

Para la aplicación de las pruebas de caja blanca se hizo uso de la técnica camino básico. El método del camino básico permite obtener una medida de la complejidad de un diseño procedimental, y utilizar esta medida como guía para la definición de una serie de caminos básicos de ejecución, diseñando casos de prueba que garanticen que cada camino se ejecuta al menos una vez.

Se toma como ejemplo el método Buscar\_Region de la clase Geometría\_Computacional como base para realizar la técnica del camino básico.

El método Buscar\_Region consiste en identificar la región a la que pertenece el caso que está siendo analizado.

Partiendo del fragmento de código tomado se obtiene el siguiente grafo de flujo:

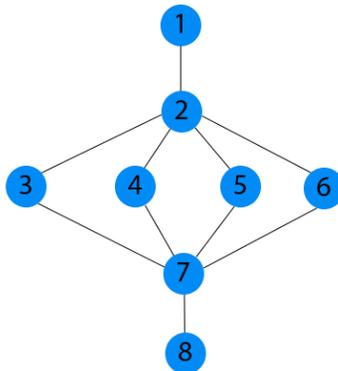


Ilustración 10. Representación del grafo de flujo de camino básico

Luego se calculó la complejidad ciclomática  $V(G)$ , obteniendo el resultado siguiente:

$$V(G) = \text{Aristas (A)} - \text{Nodos (N)} + 2$$

$$V(G) = 10 - 8 + 2$$

$$V(G) = 4$$

## Capítulo 4: Resultados y validación del sistema

El valor V (G) expresa la cantidad de caminos linealmente independientes de la estructura de control del programa, por lo que se definen los siguientes 4 caminos:

Camino básico 1: 1-> 2 -> 3 -> 7 -> 8

Camino básico 2: 1-> 2 -> 4 -> 7 -> 8

Camino básico 3: 1-> 2 -> 5 -> 7 -> 8

Camino básico 4: 1-> 2 -> 6 -> 7 -> 8

Cada camino independiente es un caso de prueba a realizar, de forma que los datos señalados causen que se visiten las sentencias vinculadas a cada nodo del camino. En el caso anterior se calcularon cuatro caminos básicos, por tanto surge la necesidad de hacer igual número de casos de prueba, para aplicar las pruebas a este método.

Tabla 20. Caso de prueba de caja blanca para el camino básico 1

Entrada	Se encuentra cargada una imagen en el sistema
Resultados Esperados	Se identifica la región a la cual pertenece el pixel que está siendo analizado.
Condiciones	Se carga una imagen de un tomograma SCHEIMPFLUG del PENTACAM

### 4.2.2 Pruebas de caja negra

Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos de prueba en que el módulo no se atiene a su especificación. Esto se refiere a que se llevan a cabo para verificar el ajuste del sistema con los requerimientos determinados. Además se enfocan especialmente en los módulos que se relacionan con la interfaz de usuario. No requieren el conocimiento de la estructura interna del programa para su puesta en marcha (40).

#### Pruebas de aceptación

Cada historia de usuario está asociada a una prueba de aceptación, conocidas también como pruebas funcionales, las mismas se realizan en esta etapa del proyecto y en ellas se describen las posibles formas de utilización del software. Las pruebas funcionales no solo validan la transformación de una entrada en una salida, sino que validan una característica completa.

En estos documentos de prueba se indican las posibles respuestas que tiene el software en la utilización de cada funcionalidad, así como los posibles mensajes de error, información o de aceptación que emite el software cuando se utiliza dicha funcionalidad.

## *Capítulo 4: Resultados y validación del sistema*

A continuación se ejemplifican algunas de las pruebas de aceptación:

*Tabla 21. Caso de prueba de aceptación para la HU Reconocer el centro de la estructura ocular*

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU4_P1	<b>Historia de Usuario:</b> 4
<b>Nombre:</b> Reconocer el centro de la estructura ocular	
<b>Descripción:</b> Prueba para la funcionalidad reconocer el centro de la estructura ocular	
<b>Condiciones de Ejecución:</b> Se debe haber cargado una imagen resultante de un tomograma Scheimpflug del PENTACAM.	
<b>Resultado Esperado:</b> El sistema reconoce el centro de la estructura ocular para posteriormente poder realizar la identificación automática de la OCP.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Tabla 22. Caso de prueba de aceptación para la HU Definir región del caso*

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU10_P2	<b>Historia de Usuario:</b> 10
<b>Nombre:</b> Definir región del caso	
<b>Descripción:</b> Prueba para la funcionalidad definir región del caso.	
<b>Condiciones de Ejecución:</b> Se debe haber cargado una imagen resultante de un tomograma Scheimpflug del PENTACAM. Se debe haber reconocido automáticamente el centro de la estructura ocular.	
<b>Resultado Esperado:</b> El sistema calcula y devuelve la región donde se localiza el caso que se analiza.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Tabla 23. Caso de prueba de aceptación para la HU Calcular certeza de solución*

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU11_P3	<b>Historia de Usuario:</b> 11

## Capítulo 4: Resultados y validación del sistema

<b>Nombre:</b> Calcular OCP en áreas definidas
<b>Descripción:</b> Prueba para la funcionalidad que calcula la opacidad en el área seleccionada por el sistema.
<b>Condiciones de Ejecución:</b> Se debe haber cargado una imagen resultante de un tomograma Scheimpflug del PENTACAM. Se debe haber reconocido automáticamente el centro de la estructura ocular. Se debe haber seleccionado automáticamente el área que presenta opacidad.
<b>Resultado Esperado:</b> El sistema guarda la configuración seleccionada y muestra un mensaje informando los resultados del cálculo.
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.

Como parte de las pruebas de aceptación se procedió a la creación de 8 casos de estudio para verificar los resultados del software, en los mismos se tomaron imágenes que corresponden a diferentes pacientes, a los que llamaremos (Paciente 1, Paciente 2, Paciente 3,..., Paciente 8). A continuación se presenta una muestra de 4 casos de estudio de los 8 realizados. Las pruebas arrojaron los siguientes resultados:

1. Caso de estudio correspondiente a la imagen resultante del tomograma Scheimpflug del ojo derecho del paciente "Paciente 1" tomado el 28/09/2010.



Ilustración 11. Análisis realizado por PANDOC al tomograma del Paciente 1

## Capítulo 4: Resultados y validación del sistema

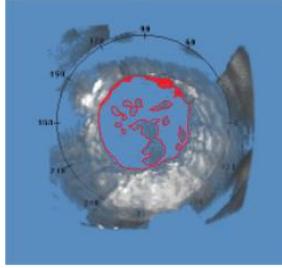


Ilustración 12. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 1

2. Caso de estudio correspondiente a la imagen resultante del tomograma Scheimpflug del ojo derecho del paciente "Paciente 2" tomado el 21/09/2010

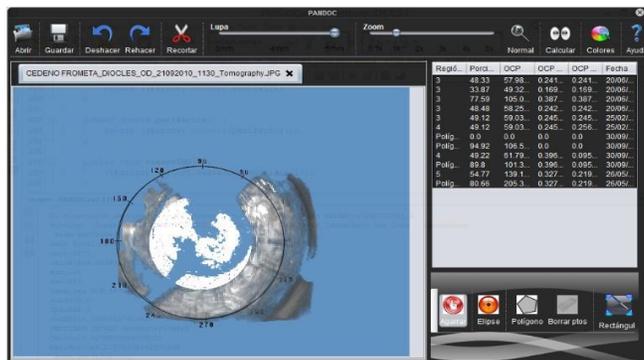


Ilustración 13. Análisis realizado por PANDOC al tomograma del Paciente 2

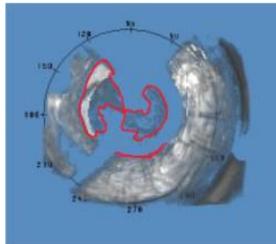


Ilustración 14. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 2

3. Caso de estudio correspondiente a la imagen resultante del tomograma Scheimpflug del ojo izquierdo del paciente "Paciente 3" tomado el 18/05/2010

## Capítulo 4: Resultados y validación del sistema

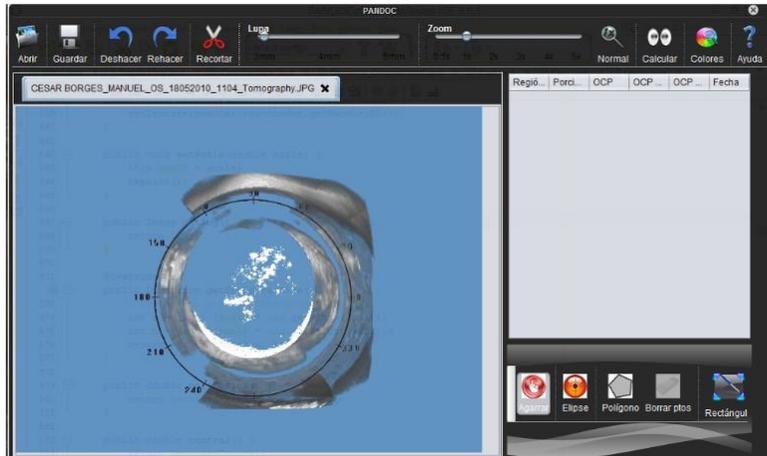


Ilustración 15. Análisis realizado por PANDOC al tomograma del Paciente 3

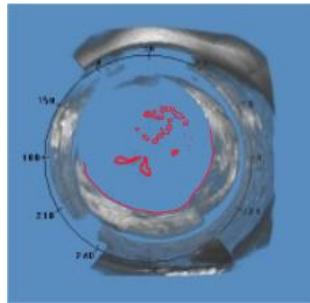


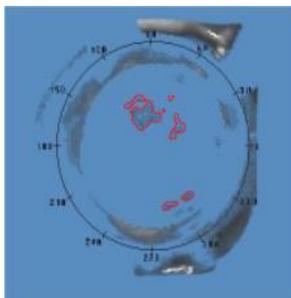
Ilustración 16. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 3

4. Caso de estudio correspondiente a la imagen resultante del tomograma Scheimpflug del ojo izquierdo del paciente "Paciente 4" tomado el 29/06/2010



Ilustración 17. Análisis realizado por PANDOC al tomograma del Paciente 4

## Capítulo 4: Resultados y validación del sistema



*Ilustración 18. Región considerada por el especialista como opacidad según su percepción al tomograma del Paciente 4*

Los resultados de las comparaciones se muestran en la tabla 24, donde se presentan los resultados de los 8 pacientes tomados como casos de estudio y el resultado arrojado después de realizadas las comparaciones

*Tabla 24. Comparación de resultados*

	Estructura ocular	Área seleccionada con opacidad	Resultados de cálculo	Comparación de opacidad central	Comparación de opacidad periférica	Comparación de % de áreas opacificadas
Paciente 1	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 1	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 2	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 2	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 3	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 3	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 4	Ojo izquierdo	Idénticos	Correctos	Correcta	Correcta	Muy similares
Paciente 4	Ojo derecho	Idénticos	Correctos	Correcta	Correcta	Muy similares
Paciente 5	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 5	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 6	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 6	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 7	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares

## *Capítulo 4: Resultados y validación del sistema*

Paciente 7	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 8	Ojo izquierdo	Muy similares	Correctos	Correcta	Correcta	Muy similares
Paciente 8	Ojo derecho	Muy similares	Correctos	Correcta	Correcta	Muy similares

El grado de satisfacción del cliente atendiendo a sus necesidades se pudo obtener a partir de cada entrega realizada. Para ello se realizaron 4 iteraciones de las pruebas de aceptación y se detectaron no conformidades significativas referentes a errores de validación, no significativas que se centraron fundamentalmente en errores ortográficos y varias recomendaciones. Las no conformidades detectadas quedan registradas en la siguiente tabla:

Tabla 25. Resultado de las pruebas de aceptación

Número de iteración	No conformidades		Recomendaciones	No conformidades resueltas
	Significativas	No significativas		
1	1	1	0	2
2	0	2	2	4
3	1	0	1	2
4	0	0	1	1

Las pruebas se realizaron de forma iterativa e incremental, y se comprobó en cada una de ellas la corrección de los errores detectados en la iteración anterior, lo que contribuyó a mejorar la calidad y funcionalidad del sistema, por lo que el sistema quedó aprobado por parte del cliente, quien evaluó el 100 % de los casos de prueba de forma satisfactoria.

### **4.3 Conclusiones del capítulo**

En el presente capítulo se abordan los temas referentes a la fase de pruebas, además se demuestra que una historia de usuario es un resumen de una funcionalidad del sistema. Se evidenció la necesidad de realizar pruebas al sistema para corregir errores no deseados; donde el cliente comprueba el cumplimiento y la satisfacción de lo que esperaba con el mismo, por lo que se realizaron las pruebas de aceptación y las pruebas unitarias con el empleo de los métodos de caja blanca y caja negra. Se pudo constatar después de 4 iteraciones, que el sistema quedó libre de no conformidades, por lo que el cliente puede utilizarlo en un entorno real.

## **Conclusiones generales**

La identificación de las diferentes regiones con opacidad puede ser utilizada para expresar la densidad de OCP que presenta el paciente.

Con el uso de la solución, se dotará al oftalmólogo de una herramienta de apoyo al diagnóstico y en la identificación de incidencia de OCP en pacientes operados de catarata, disminuyendo los sesgos entre un observador y otro.

El sistema puede ser utilizado en el análisis de tomogramas Scheimpflug por parte del personal menos experimentado y como software de apoyo en investigaciones médicas del campo de la OCP.

## **Recomendaciones**

Incluir un almacén de datos a la aplicación que le permita almacenar los resultados de los cálculos realizados por los especialistas, de manera que mediante técnicas de minería de datos, le puedan servir de ayuda para estudios y diagnóstico por parte de aquellos menos experimentados.

Implementar el mecanismo de reparación, ya que esto permitirá que el sistema repare y modifique el módulo de recuperación de errores.

## **Referencias bibliográficas**

1. **Acosta R, Hoffmeister L, Roman R, Comas M, Castilla M, Castells X.** *Revisión sistemática de estudios poblacionales de prevalencia de cataratas.* Barcelona, España : s.n., 2006, Vol. 81. 509-516.
2. **Hernández López Iván, Hernández Silva Juan Raúl, Castro González Yadira, Garcés Fernández Ailén, Veitía Rovirosa Zucell, Pérez Candelaria Eneida.** *Estrategias de prevención de la opacidad de la cápsula posterior.* Ciudad de la Habana : s.n., 2010, Revista Cubana de Oftalmología, Vol. 23.
3. **Findl Oliver, Buehl Wolf, Bauer Peter, Sycha Thomas.** *Intervenciones para la prevención de la opacificación de la cápsula posterior.* 2, s.l. : Biblioteca Cochrane Plus, 2010.
4. **Díaz Alemán Valentín Tinguaro, Perera Sanz D, Lozano López V, Rodríguez Martín Javier.** *Anillos de tensión capsular. Nuestra experiencia.* 16, 2005, Sociedad Canaria de Oftalmología.
5. **Barman SA.** *Quantification of posterior capsular opacification in digital images after cataract surgery.* San Diego, California : s.n., 2000, Vol. 3979.
6. **Findl Oliver, Buehl Wolf.** *Comparison of 4 methods for quantifying posterior capsule opacification.* 2003, Journal of Cataract & Refractive Surgery, Vol. 29.
7. **Tariq M Aslan, Niall Patton, Baljean Dhillon.** *Assessment of systems of analyzing PCO.* 2005, Journal of Cataract and Refractive Surgery, Vol. 31.
8. **Friedman D. S, Duncan D D.** *Digital image capture and automated analysis of posterior capsular opacification.* 8, 1999, Investigative Ophthalmology & Visual Science, Vol. 40.
9. **Buehl Wolf, Findl Oliver.** *Reproducibility of standardized retroillumination photography for quantification of posterior capsule opacification.* 2, 2002, Journal of Cataract & Refractive Surgery, Vol. 28.
10. **R Tetz Manfred, Gerd U Gerd U, Sperker Martina, Blum Marcus, Völcker Hans E.** *Photographic image analysis system of posterior capsule opacification.* 10, 1997, Journal of Cataract & Refractive Surgery, Vol. 23.
11. **Hernández López, Iván.** *Cuantificación objetiva de la opacidad de la cápsula posterior mediante tomogramas Scheimpflug del Pentacam.* 2, Ciudad de la Habana : s.n., 2011, Revista Cubana de Oftalmología, Vol. 24.
12. **Alvarez Cancio, Michel.** *PANDOC: Sistema basado en casos para la cuantificación objetiva de la opacidad capsular de un paciente operado de catarata.* La Habana : s.n., 2014.

## Referencias bibliográficas

13. **Alvarez Cancio Michel, Hernández Barrios Adrián, Rodríguez Puentes Rafael, Hernández López Iván.** *PANDOC: SOFTWARE PARA LA CUANTIFICACIÓN OBJETIVA DE LA OPACIDAD DE LA CÁPSULA POSTERIOR MEDIANTE TOMOGRAMAS SCHEIMPFLUG DEL PENTACAM.* La Habana : s.n., 2013.
14. **López, Hernández Iván.** *Estrategias de prevención de la opacidad de la cápsula posterior.* 1, La Habana : s.n., 2010, Vol. 23.
15. **TM Aslam, Niall Patton.** *A freely accessible, evidence based, objective system of analysis of posterior capsular opacification; evidence for its validity and reliability.* 9, s.l. : BMC Ofhthalmology, 2005, Vol. 5.
16. **Grewall D, Jain R.** *Pentacam tomograms: A Novel Method for Quantification of Poste-* 5, 2008, Vol. 49.
17. **Ayala, Peña Alejandro.** *Sistemas basados en Conocimiento:Una Base para su Concepción y Desarrollo.* México : s.n., 2006, Vol. 1.
18. **Pino Diez Raúl, Gómez Gómez Alberto, de Abajo Martínez Nicolás.** *Introducción a la inteligencia artificial : sistemas expertos, redes neuronales artificiales y computación evolutiva.* 2011.
19. **Gámez Martín José Antonio, Puerta Callejón José Miguel.** *Sistemas Expertos Probabilísticos.* 1998.
20. **Sánchez Ortiz Susana, Pérez Benitez Alfredo.** *Sistema Experto para la gestión de la Base de Conocimientos de NovaDesk.* 2011.
21. **Soria Francis Sindy, Martínez Sánchez Natalia.** *Modelo para diseñar Mapas Conceptuales Inteligentes utilizando el Razonamiento Basado en Casos.* 2010.
22. **Pérez Lara Alejandro, Almenares Alfonso Yasel.** *Implementación de un sistema basado en el conocimiento para el diagnóstico y el tratamiento médico de pacientes con dislipidemias.* 2011.
23. **Lozano Laura, Fernández Javier.** *Razonamiento Basado en Casos: Una Visión General.* España : s.n., 2008.
24. **Bregón Anibal, Simón Arancha, Alonso Carlos, Pulido Belarmino, Moro Isaac, Rodríguez Juan José.** *Un sistema de razonamiento basado en casos para la clasificación de fallos en sistemas dinámicos.* 2005.
25. **Gutiérrez Martínez Iliana, Bello Pérez Rafael E, Tellería Rodríguez Andrés.** *UN SISTEMA BASADO EN CASOS PARA LA TOMA DE DECISIONES EN CONDICIONES DE INCERTIDUMBRE.* 2, Santa Clara : s.n., 2002, Vol. 23.

## Referencias bibliográficas

26. **Martínez Sánchez Natalia, García Lorenzo María M, García Valdivia Zoila Zenaida, Ferreira Lorenzo Gheisa.** *EL PARADIGMA DEL RAZONAMIENTO BASADO EN CASOS EN EL ÁMBITO DE LOS SISTEMAS DE ENSEÑANZA/APRENDIZAJE INTELIGENTES.* 30, Santa Clara : s.n., 2009.
27. **Cortez Vásquez Augusto, Navarro Depaz Carlos, Pariona Quispe Jaime.** *Sistemas de razonamiento basado en casos aplicado a sistemas de líneas de productos de software.* 2, 2010, Vol. 7.
28. **Martínez Sánchez Natalia, García Lorenzo María Matilde, García Valdivia Zoila Zenaida.** *Modelo para diseñar sistemas de enseñanza-aprendizaje inteligentes utilizando el razonamiento basado en casos.* 3, Colombia : s.n., 2009, Vol. 6.
29. **Bernad, Juan Antonio.** *Análisis y representación del conocimiento:Aportaciones de la Psicología Cognitiva.* Universidad de Zaragoza : s.n., 1995.
30. **Suárez, Javier De Andrés.** *Técnicas de Inteligencia Artificial aplicadas al análisis de la solvencia empresarial.* 206, Universidad de Oviedo : s.n., 2000.
31. **Canós José H, Letelier Patricio, Panadés M Carmen.** *Métodologías Ágiles en el Desarrollo de Software.* Universidad Politécnica de Valencia : s.n., 2003.
32. **Sánchez Mendoza, Maria A.** *Metodologías del desarrollo de software.* 2004.
33. **Oracle.** Java. <http://www.java.com/es/about/>. [Online] [Cited: 1 14, 2015.] <http://www.java.com/es/about/>.
34. **SQTLite.** *SQTLiteManagenet.org.* [Online] [Cited: febrero 5, 2015.] <http://www.sqlitemanager.org/>.
35. **SQLite Admin.** *SQLite Administrador.* [Online] [Cited: febreo 10, 2015.] [sqliteadmin.orbmu2k.de/](http://sqliteadmin.orbmu2k.de/).
36. **Beck, Kent.** *Una explicación de la programación extrema: aceptar el cambio .* s.l. : Addison Wesley, 2002.
37. **Álvarez Cancio Michel, Rodríguez Puente Rafael, Hernández López Iván.** *DISEÑO DE UN SISTEMA BASADO EN CASOS PARA LA IDENTIFICACIÓN DE OPACIDAD MEDIANTE EL PENTACAM.* La Habana : UCIENCIA, 2014.
38. **Ruiz Shulcloper José, Guzmán Arenas Adolfo, Martínez Trinidad José Francisco.** Selección de variables y clasificación supervisada. [book auth.] José Ruiz Shulcloper. *Enfoque Lógico Combinatorio al Reconocimiento de Patrones.* Mexico : Instituto Politécnico Nacional, 1999, Vol. 1, p. 150.
39. **Gutiérrez J .J, Escalona M. J, Mejías M, Torres J.** *PRUEBAS DEL SISTEMA EN PROGRAMACIÓN EXTREMA.* Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla. : s.n., 2010.

## *Referencias bibliográficas*

**40. Hernán Marcelo, Schenone.** *Diseño de una Metodología Ágil de Desarrollo de Software.* Facultad de Ingeniería: Universidad de Buenos Aires : s.n., 2004.

## Anexos

## Anexo 1: Historias de Usuarios

Tabla 26. Historia de usuario #2

Historia de usuario	
<b>Número:</b> 2	<b>Nombre:</b> Conectar la BC con la aplicación
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> Se conecta la BC con el software PANDOC.	
<b>Observaciones:</b>	

Tabla 27. Historia de usuario #3

Historia de usuario	
<b>Número:</b> 3	<b>Nombre:</b> Reconocer imagen de tomograma
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema reconoce la imagen del tomograma que ha sido cargada.	
<b>Observaciones:</b>	

Tabla 28. Historia de usuario #4

Historia de usuario	
<b>Número:</b> 4	<b>Nombre:</b> Reconocer el centro de la estructura ocular
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se reconoce el centro de la estructura ocular para delimitar el espacio de búsqueda.	
<b>Observaciones:</b>	

Tabla 29. Historia de usuario #5

Historia de usuario	
<b>Número:</b> 5	<b>Nombre:</b> Llenar la BC
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 3 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 3 semanas
<b>Descripción:</b> Se llena la base de conocimientos.	
<b>Observaciones:</b>	

Tabla 30. Historia de usuario #6

Historia de usuario	
<b>Número:</b> 6	<b>Nombre:</b> Obtener el nuevo caso
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se obtiene el nuevo caso a resolver a partir de una imagen de tomograma SCHEIMPFLUG del PENTACAM.	
<b>Observaciones:</b> Tiene que haber sido cargada una imagen de tomograma Scheimpflug del PENTACAM.	

Tabla 31. Historia de usuario #8

Historia de usuario	
<b>Número:</b> 8	<b>Nombre:</b> Cargar Casos similares de la BC
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se cargan de la BC los casos que resultaron más semejantes al nuevo caso.	
<b>Observaciones:</b>	

Tabla 32. Historia de usuario # 9

Historia de usuario	
<b>Número:</b> 9	<b>Nombre:</b> Calcular similitud de rasgos

<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se calcula la similitud de cada uno de los rasgos del nuevo caso con los casos cargados de la BC.	
<b>Observaciones:</b>	

Tabla 33. Historia de usuario # 10

Historia de usuario	
<b>Número:</b> 10	<b>Nombre:</b> Calcular similitud de casos
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se calcula la similitud entre el nuevo caso y los casos cargados de la BC teniendo en cuenta cada uno de los rasgos que resultaron semejantes.	
<b>Observaciones:</b>	

Tabla 34. Historia de usuario # 11

Historia de usuario	
<b>Número:</b> 11	<b>Nombre:</b> Calcular certeza de solución
<b>Iteración asignada:</b> 4	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se calcula la certeza de la solución.	
<b>Observaciones:</b>	

Tabla 35. Historia de usuario # 12

Historia de usuario	
<b>Número:</b> 12	<b>Nombre:</b> Brindar datos al usuario
<b>Iteración asignada:</b> 4	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas

<b>Descripción:</b> Se le brinda al usuario los datos obtenidos del tomograma SCHEIMPFLUG del PENTACAM
<b>Observaciones:</b>

Tabla 36. Historia de usuario # 13

Historia de usuario	
<b>Número:</b> 13	<b>Nombre:</b> Recuperar el caso a la BC
<b>Iteración asignada:</b> 4	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> Se almacena el nuevo caso en la BC.	
<b>Observaciones:</b>	

## Anexo 2: Tarjetas CRC

Tabla 37. Tarjeta CRC de la clase Conexión

Clase: Conexión	
Responsabilidades	Colaboradores
1. Permite realizar consultas y operaciones con la base de conocimientos.	1. Editor

Tabla 38. Tarjeta CRC de la clase Lienzo

Clase: Lienzo	
Responsabilidades	Colaboradores
1. Permite las operaciones de extracción de los nuevos casos 2. Realiza los cálculos sobre la opacidad señalada por el sistema en la imagen 3. Establece los nuevos casos a comparar.	1. Editor 2. Recortador

Tabla 39. Tarjeta CRC de la clase Recortador

Clase: Recortador	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Realiza las operaciones sobre la imagen cargada.</li> <li>2. Realiza los cálculos de la imagen cargada.</li> <li>3. Manea la configuración de la aplicación.</li> </ol>	<ol style="list-style-type: none"> <li>1. Lienzo</li> <li>2. Configuración</li> </ol>

Tabla 40. Tarjeta CRC de la clase Configuración

Clase: Configuración	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Realiza las operaciones de configuración sobre las herramientas.</li> <li>2. Realiza las operaciones de configuración sobre los resultados.</li> </ol>	<ol style="list-style-type: none"> <li>1. Recortador</li> <li>2. Pixel</li> </ol>

Tabla 41. Tarjeta CRC de la clase Geometría Computacional

Clase: Geometria_Computacional	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Permite crear la elipse simulada con las rectas que delimitan cada una de las regiones.</li> <li>2. Permite identificar la región donde se localiza el caso que se está analizando</li> </ol>	<ol style="list-style-type: none"> <li>1. Lienzo</li> <li>2. Pixel</li> </ol>

Tabla 42. Tarjeta CRC de la clase Pixel

Clase: Pixel	
Responsabilidades	Colaboradores
1. Contiene la información relacionada con cada uno de los pixeles de la imagen.	1. Geometría Computacional

Tabla 43. Tarjeta CRC de la clase Caso

### Anexo 3: Tareas de Ingeniería generadas en la iteración 3

Tabla 44. Tarea de ingeniería de la funcionalidad Cargar casos similares de la BC

Tarea	
Número de tarea: 8	Número de HU: 8
Nombre: Cargar Casos similares de la BC	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 2
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se cargan de la BC los casos que resultaron más similares teniendo en cuenta el valor de la variable "Valor Global".	

Tabla 45. Tarea de ingeniería de la funcionalidad Calcular similitud de rasgos

Tarea	
Número de tarea: 9	Número de HU: 9
Nombre: Calcular similitud de rasgos	
Tipo de tarea: configuración - desarrollo	Puntos de estimación: 1.5
Programador responsable: Maidevis Rodríguez Moreno	
Descripción: Se aplican las funciones de comparación de rasgos a aquellos casos de la BC que resultaron semejantes teniendo en cuenta el valor de la variable "Valor Global".	

## **Glosario de Términos**

Anillo de tensión capsular: Dispositivo de forma anular, que se coloca dentro de la cápsula del cristalino para mantener una tensión adecuada.

Biocompatibilidad: Es la capacidad del implante para coexistir con los tejidos sin ocasionarles daño.

Cleanbagjet: Técnica usada para disminuir la incidencia de la OCP tras cirugía de catarata.

Cristalino: Medio refringente del ojo con alto poder dióptrico encargado de la acomodación (Enfoque para visión cercana o lejana).

Fotólisis láser: Ruptura del tejido de la cápsula posterior por acción térmica del láser.

Lente intraocular: Lente artificial que se coloca dentro del ojo para sustituir el cristalino extraído.

Pentacam: Equipo oftalmológico de alta tecnología capaz de reconstruir imágenes tridimensionales de alta resolución del polo anterior del ojo.

Imágenes de Purkinje: Efecto producido por la reflexión de la luz en la córnea y las caras anterior y posterior del cristalino.

Saco capsular: Estructura que rodea el cristalino dándole soporte.

Sistema Sheimpflug: Sistema de cámaras de alta definición acopladas a diferentes equipo médicos como el Pentacam.

PANDOC: Programa analizador de opacidad capsular.

Capsulorrexix: Apertura circular de borde continuo, que se realiza en la cápsula anterior del cristalino, por donde se extrae la catarata y se introduce la lente intraocular.