



UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Departamento de Computación

**SOFTWARE PARA EL ESPECTROFOTÓMETRO MINISCAN
XE PLUS USADO EN EL DIAGNÓSTICO DE PATOLOGÍAS
DERMATOLÓGICAS EN PACIENTES. CASO DE ESTUDIO:
CIMBUC.**

Autor:

Gabriel A. Núñez N.

Tutores:

Prof. Patricia Guerrero
Prof. Harold Vasquez

Naguanagua, Octubre de 2015.



UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Departamento de Computación

**SOFTWARE PARA EL ESPECTROFOTÓMETRO MINISCAN
XE PLUS USADO EN EL DIAGNÓSTICO DE PATOLOGÍAS
DERMATOLÓGICAS EN PACIENTES. CASO DE ESTUDIO:
CIMBUC.**

Autor:

Gabriel A. Núñez N.

Tutores:

Prof. Patricia Guerrero
Prof. Harold Vasquez

Trabajo Especial de Grado presentado ante la ilustre Universidad de Carabobo,
como credencial para optar por el título de Licenciado en Computación.

Naguanagua, Octubre de 2015.

Dedicatoria

Agradecimientos



UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Departamento de Computación

**SOFTWARE PARA EL ESPECTROFOTÓMETRO MINISCAN
XE PLUS USADO EN EL DIAGNÓSTICO DE PATOLOGÍAS
DERMATOLÓGICAS EN PACIENTES. CASO DE ESTUDIO:
CIMBUC.**

Resumen

El espectrofotómetro de reflexión difusa, denominado MiniScan XE Plus, es un instrumento de medición utilizado por el Centro de Investigaciones Médicas y Biotecnológicas de la Universidad de Carabobo (CIMBUC), que ayuda a los médicos dermatólogos a establecer diagnósticos sobre patologías en la piel de pacientes, de manera precisa y sin necesidad de realizar biopsias. No obstante, el software comercial disponible para la utilización de tal instrumento es poco amigable, difícil de utilizar e imposible de modificar y extender. La presente investigación tiene como objetivo desarrollar un software amigable, modificable y extensible, que se ajuste a las necesidades de los dermatólogos para así garantizar un mejor aprovechamiento del instrumento en cuestión.

Palabras claves: espectrofotómetro, análisis bioquímico de la piel, biopsia, ingeniería biomédica, software privativo.

Autor:
Gabriel A. Núñez N.

Tutores:
Prof. Patricia Guerrero
Prof. Harold Vasquez

Índice general

Introducción	6
I. El Problema	7
1.1. Planteamiento del problema	7
1.2. Objetivos	10
1.2.1. Objetivo general	10
1.2.2. Objetivos específicos	10
1.3. Justificación de la investigación	11
II. Marco Teórico	13
2.1. Antecedentes	13
2.1.1. Antecedentes de la investigación	13
2.1.2. Observación directa	14
2.2. Bases teóricas	15
2.2.1. Espectroscopía de reflectancia difusa	15
2.2.2. Absorbancia aparente	16
2.2.3. Iluminante estándar D65	16
2.2.4. Observador estándar de 10°	17

2.2.5. Coordinadas de cromaticidad CIE xyz	17
2.2.6. Espacio del color CIELAB	20
2.2.7. Coeficiente de absorción	21
2.2.8. Coeficiente de esparcimiento	22
2.2.9. Índice de eritema	22
III.Marco Metodológico	26
3.1. Metodología de investigación	26
3.1.1. Investigación-Acción	26
3.2. Metodología de desarrollo de software	28
3.2.1. SCRUM	29
3.2.2. Artefactos de RUP utilizados	32
IV.Resultados	34
4.1. Fases metodológicas	34
4.1.1. Visión	34
4.1.2. Pila del producto (<i>product backlog</i>)	37
4.1.3. Requerimientos no funcionales	38
4.1.4. Casos de uso	39
4.1.5. Glosario	43
4.2. Base de datos	45
4.2.1. Diagrama ER de la base de datos	45
4.2.2. Descripción de las tablas de la base de datos	46
4.3. Tecnologías y recursos utilizados	46
4.3.1. Tecnologías	46

4.3.2. Recursos	48
4.4. Comunicación con el MiniScan XE Plus	49
4.5. Fórmulas implementadas	50
4.6. Interfaz del Spectrasoft	56
4.7. Pruebas realizadas	58
V. Conclusiones y recomendaciones	59
5.1. Conclusiones	59
5.2. Recomendaciones	59
Referencias	60

Índice de figuras

2.1. HunterLab Universal Software	14
3.1. Carácter cíclico de la Investigación-Acción	27
3.2. Configuración de los artefactos a utilizar de SCRUM y RUP . .	33
4.1. Caso de uso: manejar el MiniScan XE Plus	39
4.2. Caso de uso: gestionar medición	39
4.3. Caso de uso: gestionar sesión	40
4.4. Caso de uso: gestionar historia	40
4.5. Caso de uso: gestionar muestra	41
4.6. Caso de uso: gestionar usuario	41
4.7. Diagrama ER de la base de datos	45
4.8. MiniScan XE Plus	48
4.9. Vista principal	56
4.10. Inicio de sesión	56
4.11. Curva de reflectancia	57
4.12. Historia médica de un paciente	57

Índice de tablas

1.1. Atributos esenciales de un software de calidad	12
2.1. Valores del iluminante D65	24
2.2. Valores del observador de 10°	25
3.1. Actividades del proyecto según la Investigación-Acción	28
3.2. Principios de los métodos ágiles	29
4.1. Actores del negocio	35
4.2. Actores del software	35
4.3. Beneficios y características principales del producto	36
4.4. Requerimientos funcionales del software	37
4.5. Requerimientos no funcionales del software	38

Introducción

Capítulo I

El Problema

En este capítulo se presenta la problemática que motiva el desarrollo de este trabajo de investigación, se describen los objetivos que se tiene pensado cumplir con el mismo, y se explican las razones que justifican la necesidad de su ejecución.

1.1. Planteamiento del problema

El color y la apariencia de la piel humana son importantes en el campo de la medicina. Durante el diagnóstico de enfermedades de la piel, la observación cuidadosa y la evaluación visual del área sospechada es siempre el primer paso, y el más importante. Esto es seguido generalmente por una escisión o biopsia por punción, en la que se extrae una muestra de tejido de la piel para un análisis microscópico. La observación visual suele ser subjetiva, y los pacientes a menudo se someten a cicatrices y dolor durante la biopsia. Por otro lado, las técnicas ópticas son por lo general no invasivas, y sus resultados son a menudo objetivos. Durante el diagnóstico no invasivo, no se crea ninguna ruptura en la piel, y los pacientes no se someten al dolor ni a cicatrices durante el tratamiento (Bersha, 2010).

Los avances tecnológicos de la actualidad permiten emplear técnicas de óptica, capaces de estudiar las propiedades estructurales y bioquímicas del tejido biológico de manera precisa y no invasiva. Los instrumentos que emplean tales técnicas son de gran ayuda para los médicos dermatólogos, razón por la cual dichos instrumentos han tomado suma importancia en el área de la medicina dermatológica.

Hoy en día existen diferentes tipos de estudios ópticos in-situ, in-vivo e invitro del tejido biológico, como la espectroscopía de reflectancia difusa (ERD). Pérez (2012) asegura que con esta técnica es posible estudiar las propiedades bioquímicas y las condiciones estructurales de un tejido biológico, analizando la interacción luz-tejido de una manera no invasiva.

En este sentido, el Centro de Investigaciones Médicas y Biotecnológicas de la Universidad de Carabobo (CIMBUC) dispone de un espectrofotómetro de reflexión difusa, denominado MiniScan XE Plus. Este dispositivo fue creado por la empresa HunterLab, la cual lo describe como un instrumento que aplica la técnica de ERD, utilizado para medir la reflectancia de especímenes como una función de longitud de onda.

Ahora bien, el CIMBUC hace uso de este instrumento a través del único software disponible para su utilización, designado HunterLab Universal Software. Este es un software comercial y privativo capaz de ejecutarse en sistemas operativos Windows, desde la versión 95 hasta la versión XP. Dicho software contiene funciones que abarcan la utilización del MiniScan XE Plus y de otros instrumentos ofrecidos por HunterLab; su interfaz gráfica de usuario está en inglés, además de que los resultados que genera no poseen el formato de gestión de información de pacientes con el que trabajan los dermatólogos del CIMBUC. Por último, este software fue descontinuado en el año 2008.

Tomando en cuenta lo mencionado anteriormente, se tiene que el HunterLab Universal Software es privativo y está descontinuado, por lo tanto no existe

la posibilidad de modificarlo ni de extenderlo. Dicho software está en inglés y ofrece funciones ajenas al uso exclusivo del MiniScan XE Plus, causando que su interfaz gráfica de usuario contenga más opciones de las necesarias para manejar tal instrumento, y que sea difícil de entender por los dermatólogos. Todo esto sumado al hecho de que los resultados generados por este software deben exportarse y almacenarse manualmente, porque los mismos no poseen el formato con el que trabajan los dermatólogos, produce la necesidad de contar con asistencia técnica especializada para su debida utilización y ralentiza las consultas con los pacientes.

De lo antedicho se desprende que el HunterLab Universal Software posee una interfaz gráfica de usuario poco amigable, y el costo del tiempo de capacitación para su uso correcto podría ser alto. Dicho software no podrá modificarse ni extenderse, por lo tanto no se fomentará el uso del instrumento en cuestión, disminuyendo su potencial. El formato de los resultados de este software convertirá las consultas de los dermatólogos con los pacientes en una labor ineficiente en términos de tiempo. Por último, no se fomentará el desarrollo de nuevas funciones que utilicen los resultados de dicho software como insumo, sosegando así la posibilidad de realizar análisis más complejos, y de proveer resultados que les permitan a los dermatólogos establecer diagnósticos más completos sobre patologías dermatológicas en pacientes.

Motivado a lo anterior, se desarrolló un software modificable y extensible, con una interfaz gráfica de usuario amigable, con funciones para el uso exclusivo del MiniScan XE Plus, que genera resultados relevantes para los dermatólogos empleando el formato utilizado por ellos para registrar las consultas con los pacientes, y siguiendo los lineamientos de la ingeniería del software pertinentes.

Finalmente, con esta investigación se espera fomentar la utilización del MiniScan XE Plus, reducir el tiempo de las consultas con los pacientes, aportar un software sobre el que se puedan desarrollar nuevas investigaciones que conlleven a análisis más complejos, y por último, realizar diagnósticos más

completos y diversos sobre patologías dermatológicas presentes en pacientes.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollar un software para el espectrofotómetro MiniScan XE Plus, usado en el diagnóstico de patologías dermatológicas en pacientes, tomando como caso de estudio el CIMBUC.

1.2.2. Objetivos específicos

- Investigar el estado del arte relacionado a la investigación:
 1. Técnicas de óptica y colorimetría.
 2. MiniScan XE Plus.
 3. HunterLab Universal Software.
 4. Atributos de calidad del software.
- Seleccionar una metodología de investigación y una metodología de desarrollo para el nuevo software.
- Desarrollar el nuevo software, siguiendo las metodologías seleccionadas.
- Realizar las pruebas para el nuevo software.
- Elaborar el manual de usuario para el uso del nuevo software.

1.3. Justificación de la investigación

Empezando con la interfaz gráfica de usuario, Sommerville (2005) señala que el diseño cuidadoso de la misma es una parte fundamental del proceso de diseño general del software. Si un software debe alcanzar su potencial máximo, es fundamental que su interfaz gráfica de usuario sea diseñada para ajustarse a las habilidades, experiencia y expectativas de sus usuarios previstos. Un buen diseño de la interfaz gráfica de usuario es crítico para la confiabilidad del software. Muchos de los llamados errores de usuario son causados porque las interfaces gráficas de usuario no consideran las habilidades de los usuarios reales y su entorno de trabajo.

Dicho lo anterior, el diseño de la interfaz gráfica de usuario del HunterLab Universal Software es la principal razón por la cual los dermatólogos requieren de personal técnico especializado que los asista al momento de utilizarlo. Esto porque dicha interfaz está en inglés, ofrece funciones que no son necesarias para la utilización del MiniScan XE Plus, y sus resultados no poseen un formato adecuado. Por estas razones los dermatólogos perciben este software comercial como no intuitivo, ni auto descriptivo ni amigable, temiendo cometer errores al utilizarlo sin asistencia y generar resultados erróneos, poniendo en riesgo la fiabilidad del diagnóstico, y en consecuencia, la salud de los pacientes.

Con respecto a software de calidad, Sommerville (2005) explica lo siguiente: así como los servicios que proveen, los productos de software tienen un cierto número de atributos asociados que reflejan su calidad. Estos atributos no están directamente relacionados con lo que hace el software; más bien, reflejan su comportamiento durante su ejecución, en la estructura y la organización del programa fuente, y en la documentación asociada.

El conjunto específico de atributos que se esperan de un software de calidad depende obviamente de su aplicación. En la tabla 1.1 se puede apreciar la generalización de estos atributos.

Tabla 1.1: *Atributos esenciales de un software de calidad* (Fuente: Sommerville, 2005).

Característica	Descripción
Mantenibilidad	El software debe describirse de tal forma que pueda evolucionar para cumplir las necesidades de cambio de los clientes. Este es un atributo crítico, debido a que el cambio en el software es una consecuencia inevitable de un cambio en el entorno de negocios.
Confiabilidad	Este atributo tiene un gran número de características, incluyendo la fiabilidad, la protección y la seguridad. El software confiable no debe causar daños físicos o económicos en caso de que ocurra una falla del sistema.
Eficiencia	El software no debe hacer que se malgasten los recursos del sistema, como la memoria y los ciclos de procesamiento. Por lo tanto, la eficiencia incluye tiempos de respuesta y de procesamiento, utilización de la memoria, etcétera.
Usabilidad	El software debe ser fácil de utilizar, sin esfuerzo adicional por parte del usuario para quien está diseñado. Esto significa que debe tener una interfaz gráfica de usuario apropiada, y una documentación adecuada.

Debido a que el HunterLab Universal Software es privativo, el CIMBUC no dispone de su código fuente, de manera que este software no puede modificarse ni adaptarse a necesidades específicas, y por lo tanto, no posee el primer atributo esencial para un software de calidad: la mantenibilidad. Por la misma razón, no se puede determinar con certidumbre el segundo atributo: la confiabilidad. Por último, la usabilidad de este software es baja, ya que la interfaz gráfica de usuario es poco amigable.

Las razones descritas anteriormente justifican la necesidad del desarrollo de un software para el uso del MiniScan XE Plus que sea amigable, modificable, extensible, y que cumpla con los atributos esenciales de calidad; que emplee el formato de historia médica con el que trabajan dermatólogos, y que ofrezca las funciones que ellos necesitan para realizar análisis y establecer diagnósticos de patologías dermatológicas en pacientes. Por último, se ha tomado como caso de estudio el CIMBUC.

Capítulo II

Marco Teórico

Este capítulo presenta los antecedentes, la observación directa y las bases teóricas que enmarcan la teoría que sustenta el desarrollo del presente trabajo de investigación.

2.1. Antecedentes

2.1.1. Antecedentes de la investigación

En el artículo científico titulado «*Comparing Quantitative Measures of Erythema, Pigmentation and Skin Response using Reflectometry*», realizado por Wagner et al., (2002), en la Universidad del Estado de Pensilvania, Estados Unidos, y publicado por Pigment Cell Res, se obtiene el índice de eritema, que es utilizado para determinar el nivel inflamatorio de la epidermis de un paciente. El método utilizado en este artículo para su obtención fue implementado en el nuevo software.

En el artículo científico titulado «*Recuperación del Coeficiente de Absorción de la Epidermis en la Piel Humana*», realizado por Narea et al., (2015), en la

Universidad de Carabobo, Venezuela, y publicado por la Sociedad Española de Óptica, se determina el coeficiente de absorción, que es un parámetro óptico asociado a la piel, el cual indica el nivel de concentración de melanina presente en la epidermis de un paciente. La técnica empleada en dicho artículo para calcular este coeficiente fue implementada en el nuevo software.

2.1.2. Observación directa

El «*HunterLab Universal Software*», es un software comercial y privativo de 16 bits diseñado para el sistema operativo Microsoft Windows versión 3.x, con la posibilidad de ejecutarse en Windows 95, Windows 98, Windows 2000 y Windows XP. Fue creado para la utilización del MiniScan XE Plus, además de otros instrumentos de la empresa HunterLab, y descontinuado en el año 2008. Este software dispone de algunas de las funciones que fueron desarrolladas en el nuevo software, razón por la cual es una referencia importante de observación. En la figura 2.1 se puede apreciar la vista principal de la interfaz de este software.

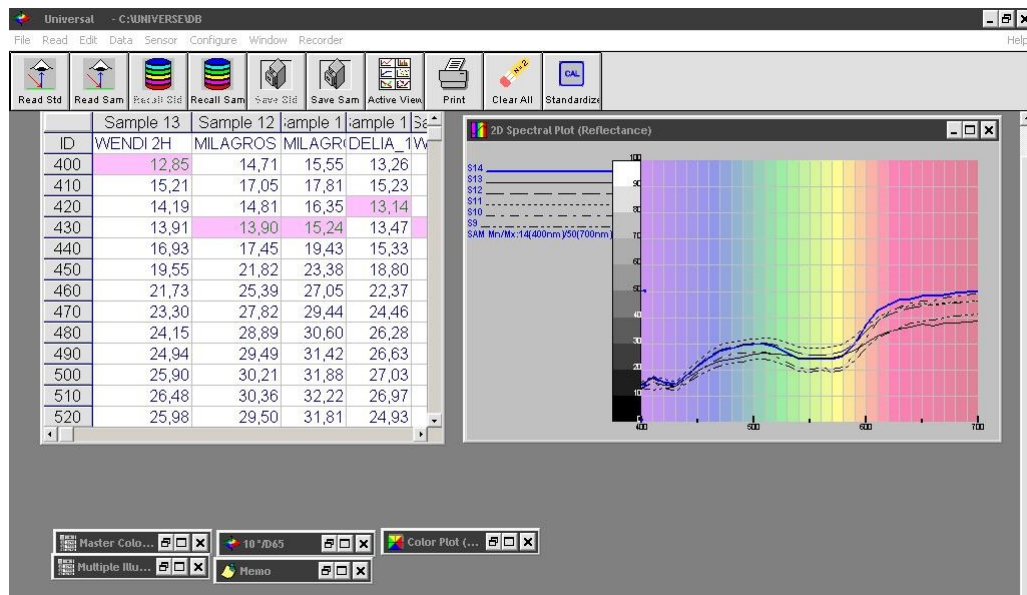


Figura 2.1: *HunterLab Universal Software* (Fuente: CIMBUC, 2015).

El archivo denominado «*MSXE + OCX*», es una hoja de cálculo habilitada para la ejecución de macroinstrucciones de Microsoft Excel, que fue proporcionada por el personal de soporte técnico de HunterLab como un ejemplo para utilizar el MiniScan XE Plus, empleando el uso de un kit de control denominado MiniScan XE Plus OCX Kit (*MSXE.ocx*). Este kit fue diseñado por la empresa HunterLab para dar acceso a las características comunmente utilizadas por dicho instrumento. El código contenido en la hoja de cálculo se empleó como referencia para el manejo del kit *MSXE.ocx*.

2.2. Bases teóricas

2.2.1. Espectroscopía de reflectancia difusa

La ERD (espectroscopía de reflectancia difusa) es una técnica con la cual se puede estudiar tejido biológico. En el campo de las aplicaciones biomédicas resulta útil para propósitos de diagnóstico, ya que se pueden estudiar tejidos de manera no invasiva, también ha demostrado ser una técnica de gran utilidad en aplicaciones de diagnóstico en varias situaciones modernas (Pérez, 2012).

Para llevar a cabo una medición con ERD se requiere hacer incidir la luz de una fuente, cuyo espectro de emisión sea conocido, sobre el tejido que se quiere estudiar. La luz que logra propagarse en el tejido y que es re-emitida por este hacia la superficie irradiada, será capturada por algún dispositivo fotosensible (en el caso de esta investigación, el MiniScan XE Plus), para ser comparada posteriormente con la luz incidente o espectro de referencia, y así poder determinar qué tanto cambió dicho espectro después de haber interactuado con el tejido.

Normalmente los datos espectrales de la reflectancia difusa $R(\lambda)$ son multiplicados por un factor de 100 por los dispositivos fotosensibles, para repre-

sentarlos en forma de curva en una escala del 0% al 100%, a lo largo de puntos discretos que representan las longitudes de onda con las que opera dicho dispositivo; tal es el caso del MiniScan XE Plus.

2.2.2. Absorbancia aparente

Según el *Random House Kernerman Webster's College Dictionary* (2010), el espectro de absorción es la radiación electromagnética en ciertas longitudes de onda que atraviesa un medio, y que es absorbida por el mismo. En cierto modo, es el opuesto del espectro de reflectancia, es decir, es la luz que logra propagarse en el tejido y que no es re-emitida por este hacia la superficie irradiada.

Por lo tanto, la absorbancia aparente es la luz que está siendo absorbida aparentemente por el tejido en estudio, y que no está siendo reflejada de vuelta al MiniScan XE Plus. La misma se puede calcular de la siguiente manera: $A(\lambda) = 100 - R(\lambda)$ (recordando que la reflectancia es multiplicada por un factor de 100 por el MiniScan XE Plus) y se puede representar en forma de curva, de la misma manera que la reflectancia difusa.

2.2.3. Iluminante estándar D65

El tipo de luz bajo el cual se observa un objeto puede afectar su apariencia. Para cuantificar estas fuentes de luz blanca, la *Commission Internationale de l'Eclairage* (CIE) desarrolló iluminantes estándares para la medición del color. HunterLab (2015) define el iluminante como una tabla cuantificable de números que representan la energía relativa en comparación con la longitud de onda de una fuente de luz.

El iluminante estándar D65, según es descrito por la CIE (2015), tiene el

propósito de representar la luz de día promedio, y tiene una temperatura de color correlacionada de aproximadamente 6500 K°. Los valores numéricos que representan este iluminante se muestran en la tabla 2.1.

2.2.4. Observador estándar de 10°

HunterLab (2008) describe que en la observación visual, el observador es el ojo humano que recibe la luz reflejada desde o a través de un objeto, y el cerebro el cual percibe la visión. Debido a que los humanos perciben el color y la apariencia de formas distintas, subjetivamente, se han hecho intentos para estandarizar el observador humano como una representación de lo que una persona promedio ve u observa.

En 1964, se desarrolló la función del observador estándar CIE de 10°, denominado así debido a que los experimentos llevados a cabo para establecer dicho estándar involucraron a sujetos que juzgaban colores, mientras observaban a través de un agujero que les permitía tener un campo de visión de 10°. Este observador estándar, en la forma de una función matemática de la respuesta humana a cada longitud de onda de luz, es utilizado en cálculos del color. Los valores numéricos de las funciones que representan este estándar se muestran en la tabla 2.2.

2.2.5. Coordenadas de cromaticidad CIE xyz

La sensación de luz es producida por radiación electromagnética visible, que cae dentro de los límites de longitud de onda de 380 nanómetros y 780 nanómetros. La radiación proveniente de la región de longitud de onda corta produce usualmente la sensación de luz azul, la radiación con longitudes de onda entre 520 nanómetros y 550 nanómetros son vistas como luz verde, y por encima de alrededor de los 650 nanómetros se percibe usualmente la luz de

color rojo. Estos límites no están bien definidos, y la percepción actual depende fuertemente del estado de adaptación del ojo y del estímulo de luz que rodea el objeto en estudio (Schanda, 2007).

La CIE definió un estándar para calcular los valores de estos estímulos, denominándolos valores triestímulo XYZ o sistema tricromático CIE XYZ. Tomando en cuenta el rango de longitudes de onda con el que opera el MiniScan XE Plus, estos valores son calculados utilizando las fórmulas descritas a continuación.

$$X = k \int_{400 \text{ nm}}^{700 \text{ nm}} R(\lambda) S(\lambda) \bar{x}(\lambda) d\lambda$$

$$Y = k \int_{400 \text{ nm}}^{700 \text{ nm}} R(\lambda) S(\lambda) \bar{y}(\lambda) d\lambda$$

$$Z = k \int_{400 \text{ nm}}^{700 \text{ nm}} R(\lambda) S(\lambda) \bar{z}(\lambda) d\lambda$$

En donde $R(\lambda)$ es el factor de reflectancia difusa, $S(\lambda)$ es la distribución de energía espectral relativa de un iluminante estándar, en este caso del iluminante D65 (véase la tabla 2.1), $\bar{x}(\lambda)$, $\bar{y}(\lambda)$ y $\bar{z}(\lambda)$ son las funciones de correspondencia del color, dado el observador estándar CIE de 10° (véase la tabla 2.2), y por último, k es una constante que se calcula con la fórmula mostrada a continuación.

$$k = \frac{100}{\sum_{\lambda} \bar{y}(\lambda) d\lambda}$$

De acuerdo con la recomendación de la CIE, la integración puede llevarse a cabo con una sumatoria numérica a intervalos de longitud de onda, $\Delta\lambda$, equivalentes a 10 nanómetros, para el caso del MiniScan XE Plus.

$$X = k \sum_{\lambda} R(\lambda) S(\lambda) \bar{x}(\lambda) \triangle \lambda$$

$$Y = k \sum_{\lambda} R(\lambda) S(\lambda) \bar{y}(\lambda) \triangle \lambda$$

$$Z = k \sum_{\lambda} R(\lambda) S(\lambda) \bar{z}(\lambda) \triangle \lambda$$

Ahora bien, el estímulo de un color se puede describir completamente por los tres valores triestímulo, pero esta descripción no es muy fácilmente concebible. Según Schanda (2007), es difícil imaginar un estímulo si solamente se dan sus valores triestímulo, y frecuentemente no se buscan los valores absolutos de los mismos. En tales casos se pueden utilizar las coordenadas de cromaticidad xyz. Finalmente, las coordenadas de cromaticidad xyz se definen con las fórmulas mostradas a continuación.

$$x = \frac{X}{X + Y + Z}$$

$$y = \frac{Y}{X + Y + Z}$$

$$z = \frac{Z}{X + Y + Z}$$

2.2.6. Espacio del color CIELAB

Según Schanda (2007), los estímulos del color son tridimensionales, y la solicitud de extender el espacio del color uniforme a un espacio tridimensional ya había sido expresada en los años 60. En 1976 se aceptó la recomendación para el diagrama de espacio del color uniforme CIELAB ($L^*a^*b^*$).

El espacio del color CIELAB no es más que un sistema para transformar las coordenadas de cromaticidad CIE xyz, a coordenadas $L^*a^*b^*$ representables en un espacio tridimensional, y está definido por las ecuaciones descritas a continuación.

$$L^* = 116f(Y/Y_n) - 16$$

$$a^* = 500[f(X/X_n) - f(Y/Y_n)]$$

$$b^* = 200[f(Y/Y_n) - f(Z/Z_n)]$$

$$\text{Donde } f(X/X_n) = (X/X_n)^{1/3} \text{ si } (X/X_n) > (24/116)^3$$

$$f(X/X_n) = (841/108)(X/X_n) + 16/116 \text{ si } (X/X_n) \leq (24/116)^3$$

$$\text{Donde } f(Y/Y_n) = (Y/Y_n)^{1/3} \text{ si } (Y/Y_n) > (24/116)^3$$

$$f(Y/Y_n) = (841/108)(Y/Y_n) + 16/116 \text{ si } (Y/Y_n) \leq (24/116)^3$$

Donde $f(Z/Z_n) = (Z/Z_n)^{1/3}$ si $(Z/Z_n) > (24/116)^3$

$f(Z/Z_n) = (841/108)(Z/Z_n) + 16/116$ si $(Z/Z_n) \leq (24/116)^3$

En donde X, Y, Z son los valores triestímulo del color considerado del objeto o tejido en estudio y X_n, Y_n, Z_n son los valores triestímulo de la fuente de luz. Para el caso del iluminante estándar D65, y tomando en cuenta el observador estándar de 10° , los valores de X_n, Y_n, Z_n son $X_n = 94,81$ $Y_n = 100,00$ y $Z_n = 107,32$.

2.2.7. Coeficiente de absorción

La piel es un medio biológico que se comporta perfectamente como un medio turbio multicapa, donde el principal agente de absorción es la melanina, la cual es producida en la epidermis, y es uno de los parámetros que determinan la coloración de la piel (Narea et al., 2015).

En la actualidad, existe el problema de calcular confiablemente los parámetros ópticos (entre ellos, el coeficiente de absorción) de los medios turbios a partir del conocimiento de su reflectancia. La determinación de los parámetros ópticos es de vital importancia para el desarrollo de las investigaciones relativas a la caracterización de tejidos biológicos, en especial de la piel, empleando métodos ópticos.

Narea et al., (2015) plantea recuperar el coeficiente de absorción de la epidermis a través del ajuste trigonométrico de los espectros de reflectancia difusa, brindando de esta forma una herramienta matemática que permite relacionar los parámetros del ajuste de las curvas espectrales con las propiedades ópticas de la piel.

La fórmula empleada para recuperar el coeficiente de absorción de la

epidermis, dada una curva espectral $R(\lambda)$, es la mostrada a continuación.

$$\mu_{a,epi}(a_0, \lambda) = Ze^{ka_0} 6,6 \cdot 10^{11} \lambda^{-3,3}$$

En donde $Z = 0,2796$, $k = -7,174$, y a_0 es un coeficiente hallado aplicando ajuste polinomial por series trigonométricas a la curva espectral $R(\lambda)$.

2.2.8. Coeficiente de esparcimiento

2.2.9. Índice de eritema

Una propiedad fundamental de la piel es su capacidad para responder a la radiación ultravioleta. En muchas personas y poblaciones estas respuestas son claramente adaptativas, en donde la primera respuesta, el eritema (enrojecimiento), es tanto una señal para la persona quemada por el sol de quedarse dentro, como una señal de que el sistema inmunológico está activo y el proceso de curación ha comenzado (Wagner et al., 2002).

La respuesta de la piel resultó ser importante clínicamente cuando los protocolos de tratamiento se establecieron en la década de los 70 para regímenes de fototerapia para la psoriasis y otras enfermedades de la piel. Para calcular el índice de eritema, primero hay que determinar el promedio ponderado de la reflectancia de la luz en el rango de longitud de onda verde, y en el rango de longitud de onda roja, utilizando las fórmulas mostradas a continuación.

$$R_{verde} = \left[\left(\frac{1}{2} R_{560 \text{ nm}} + R_{570 \text{ nm}} + \frac{1}{2} R_{580 \text{ nm}} \right) / 2 \right] / 100$$

$$R_{roja} = \left[\left(\frac{1}{2} R_{640 \text{ nm}} + R_{650 \text{ nm}} + R_{660 \text{ nm}} + \frac{1}{2} R_{670 \text{ nm}} \right) / 3 \right] / 100$$

Teniendo ya el promedio ponderado de dichas reflectancias, la fórmula para calcular el índice de eritema es la siguiente.

$$E = 100 \cdot [\log(1/R_{verde}) - \log(1/R_{roja})]$$

Tabla 2.1: *Valores del iluminante D65* (Fuente: CIE, 2004).

Longitud de onda λ	Función $S(\lambda)$
400 nm	82.7549
410 nm	91.4860
420 nm	93.4318
430 nm	86.6823
440 nm	104.865
450 nm	117.008
460 nm	117.812
470 nm	114.861
480 nm	115.923
490 nm	108.811
500 nm	109.354
510 nm	107.802
520 nm	104.790
530 nm	107.689
540 nm	104.405
550 nm	104.046
560 nm	100.000
570 nm	96.3342
580 nm	95.7880
590 nm	88.6856
600 nm	90.0062
610 nm	89.5991
620 nm	87.6987
630 nm	83.2886
640 nm	83.6992
650 nm	80.0268
660 nm	80.2146
670 nm	82.2778
680 nm	78.2842
690 nm	69.7213
700 nm	71.6091

Tabla 2.2: *Valores del observador de 10°* (Fuente: CIE, 2004).

Longitud de onda λ	Función $\bar{x}(\lambda)$	Función $\bar{y}(\lambda)$	Función $\bar{z}(\lambda)$
400 nm	0.019110	0.002004	0.086011
410 nm	0.084736	0.008756	0.389366
420 nm	0.204492	0.021391	0.972542
430 nm	0.314679	0.038676	1.553480
440 nm	0.383734	0.062077	1.967280
450 nm	0.370702	0.089456	1.994800
460 nm	0.302273	0.128201	1.745370
470 nm	0.195618	0.185190	1.317560
480 nm	0.080507	0.253589	0.772125
490 nm	0.016172	0.339133	0.415254
500 nm	0.003816	0.460777	0.218502
510 nm	0.037465	0.606741	0.112044
520 nm	0.117749	0.761757	0.060709
530 nm	0.236491	0.875211	0.030451
540 nm	0.376772	0.961988	0.013676
550 nm	0.529826	0.991761	0.003988
560 nm	0.705224	0.997340	0.000000
570 nm	0.705224	0.955552	0.000000
580 nm	1.014160	0.868934	0.000000
590 nm	1.118520	0.777405	0.000000
600 nm	1.123990	0.658341	0.000000
610 nm	1.030480	0.527963	0.000000
620 nm	0.856297	0.398057	0.000000
630 nm	0.647467	0.283493	0.000000
640 nm	0.431567	0.179828	0.000000
650 nm	0.268329	0.107633	0.000000
660 nm	0.152568	0.060281	0.000000
670 nm	0.081261	0.031800	0.000000
680 nm	0.040851	0.015905	0.000000
690 nm	0.019941	0.007749	0.000000
700 nm	0.009577	0.003718	0.000000

Capítulo III

Marco Metodológico

En este capítulo se describen la metodología de investigación y la metodología de desarrollo que se emplearon para planificar, diseñar y desarrollar el software propuesto en el presente trabajo de investigación.

3.1. Metodología de investigación

3.1.1. Investigación-Acción

Baskerville (1999) define la Investigación-Acción como un método de investigación que a finales de la década de los 90 empezó a crecer en popularidad, para el uso en investigaciones académicas de sistemas de información. Este método produce resultados de investigación altamente relevantes debido a que se fundamenta en la acción práctica, dirigida a resolver un problema mientras se informa cuidadosamente sobre la teoría.

Esta metodología tiene una doble finalidad: generar un beneficio al cliente de la investigación, y al mismo tiempo generar conocimiento de investigación relevante. Por lo tanto, es una forma de investigar de carácter colaborativo que

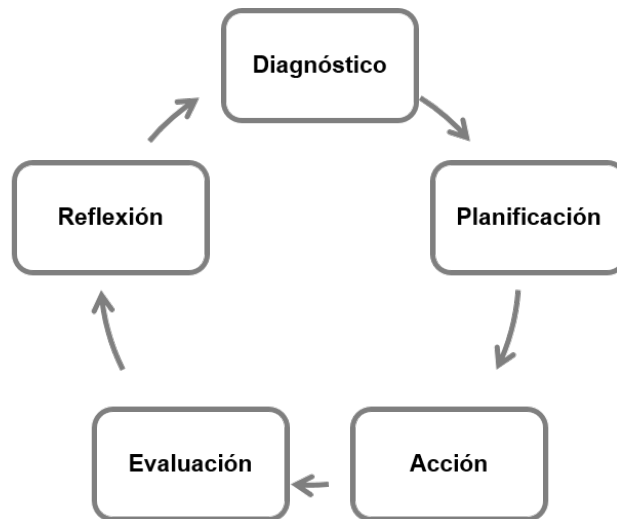


Figura 3.1: *Carácter cíclico de la Investigación-Acción* (Fuente: Baskerville, 1999).

busca unir teoría y la práctica entre investigadores y practicantes, mediante un proceso de naturaleza cíclica.

La representación más habitual de la Investigación-Acción es la descrita por Baskerville (1999), en forma de cinco fases que conforman un ciclo, las cuales se muestran en la figura 3.1 y se describen a continuación.

- **Fase de diagnóstico:** se realiza el proceso de identificación de los problemas primarios de la investigación.
- **Fase de planificación:** se especifican las acciones que se llevarán a cabo para solucionar los problemas primarios.
- **Fase de acción:** se ejecutan las acciones planificadas en la fase anterior.
- **Fase de evaluación u observación:** se efectúa una evaluación de los resultados obtenidos para observar, conocer y documentar los efectos de las acciones que fueron realizadas.
- **Fase de reflexión:** se toman los conocimientos adquiridos en la Investigación-Acción. Si las acciones ejecutadas no fueron exitosas, los conocimientos pueden proporcionar la base para el diagnóstico de un nuevo ciclo de Investigación-Acción.

Tabla 3.1: *Actividades del proyecto según la Investigación-Acción* (Fuente: Autor).

Fase	Actividades
Diagnóstico	Identificar los problemas y limitaciones que presenta el HunterLab Universal Software.
Planificación	Seleccionar la metodología de desarrollo, determinar los requisitos del software y realizar un plan de trabajo.
Acción	Desarrollar el software, tomando en cuenta los requisitos identificados previamente y los lineamientos de calidad del software.
Evaluación	Realizar las pruebas de funcionalidad e interfaz gráfica de usuario del nuevo software.
Reflexión	Presentar los resultados y los análisis de las pruebas realizadas.

En la tabla 3.1 se muestran las actividades de la presente investigación, haciendo correspondencia a cada una de las fases de la Investigación-Acción descritas por Baskerville (1999).

3.2. Metodología de desarrollo de software

Para que el desarrollo del nuevo software cumpliera con los objetivos propuestos en la presente investigación, y tomando en cuenta los atributos de calidad planteados por la ingeniería del software, se realizó una revisión del enfoque que debería tener la metodología de desarrollo a utilizar.

Según Sommerville (2005), en los años 80 y a principios de los 90, existía una opinión general de que la mejor forma de obtener un mejor software era a través de una planificación cuidadosa del proyecto, una garantía de calidad formalizada, la utilización de métodos de análisis y diseño soportados por herramientas *CASE*, y por medio de procesos de desarrollo de software controlados y rigurosos. El software que seguía lo mencionado previamente era desarrollado por grandes equipos que a veces trabajaban para compañías diferentes, que a menudo estaban dispersos geográficamente y trabajaban en el software durante largos periodos de tiempo.

Tabla 3.2: *Principios de los métodos ágiles* (Fuente: Sommerville, 2005).

Principio	Descripción
Participación del cliente	Los clientes deben estar fuertemente implicados en todo el proceso de desarrollo.
Entrega incremental	El software se desarrolla en incrementos, en los que el cliente especifica los requerimientos a incluir en cada incremento.
Personas, no procesos	Se deben reconocer y explotar las habilidades del equipo de desarrollo. A este se le debe dejar desarrollar su propia forma de trabajar, sin procesos formales.
Aceptar el cambio	Se debe contar con que los requerimientos del software cambian, por lo que el software se diseña para dar cabida a estos cambios.
Mantener la simplicidad	Se debe centrar la simplicidad tanto en el software a desarrollar como en el proceso de desarrollo. Donde sea posible, se trabaja activamente para eliminar la complejidad del software.

Ahora bien, debido a que se tenía un equipo pequeño para el desarrollo del nuevo software, y a que no se iba a trabajar en este durante un largo periodo de tiempo, se eligió la utilización de una metodología de desarrollo de enfoque ágil. De acuerdo con Sommerville (2005), los métodos ágiles dependen de un enfoque iterativo para la especificación, el desarrollo y la entrega del software, y están pensados para entregar software funcional de forma rápida a los clientes, quienes pueden entonces proponer que se incluyan en iteraciones posteriores del software nuevos requerimientos o cambios en los mismos. Si bien los métodos ágiles proponen procesos diferentes para el desarrollo y la entrega incrementales del software, comparten unos principios en común, los cuales son ilustrados en la tabla 3.2.

3.2.1. SCRUM

De acuerdo con Schwaber y Sutherland (2013), esta metodología ágil es un marco de trabajo de procesos, que ha sido utilizado para gestionar el desarrollo de productos complejos desde principios de los años 90. SCRUM muestra la

eficacia relativa de las prácticas de gestión de productos y las prácticas de desarrollo.

La estructura de desarrollo de SCRUM se basa en ciclos de trabajo llamados *sprints*. Los *sprints* son iteraciones de una a cuatro semanas que suceden una detrás de la otra, con una duración fija y con fechas de culminación previamente establecidas. Se seleccionan los requerimientos que se van a desarrollar de una lista priorizada. Todos los días el equipo se reúne, y al final del *sprint* el equipo revisa el mismo con los *stakeholders*.

Hundermark (2015) explica de forma precisa los roles que conforman el equipo de desarrollo de SCRUM:

Los Roles

- **Dueño del producto (*Product Owner*):** su responsabilidad es optimizar el retorno de la inversión, asegurando que el equipo SCRUM este ocupado en entregar las características más valiosas del producto. Su trabajo principal es concentrarse en la efectividad, esto es elaborar el producto correcto para sus clientes.
- **Equipo de desarrollo:** es una colección de personas responsables por entregar incrementos de la funcionalidad del producto al final de cada *sprint*. El trabajo principal de este equipo es concentrarse en la eficiencia, esto es elaborar el producto correcto para su *Product Owner* y sus usuarios.
- **Maestro SCRUM (*SCRUM Master*):** gestiona todos los aspectos del proceso del equipo SCRUM. Su trabajo principal es concentrarse en el progreso continuo del equipo, acortando los ciclos de retroalimentación mediante los cuales aprende.

Las Reuniones

Como es sabido, el *sprint* marca cada una de las iteraciones dentro del ciclo de desarrollo de SCRUM. Por otra parte, la planificación, la continua revisión y la retrospectiva definen el inicio y el final del *sprint*. Las reuniones que ocurren en cada *sprint* son las siguientes:

- **Reunión de planificación del *sprint*:** esta reunión marca el inicio de cada *sprint*. Su propósito para el equipo SCRUM es planear el trabajo que van a realizar durante el *sprint* actual.
- **Reunión diaria del *sprint*:** el equipo de desarrollo se reúne para comunicar su trabajo, sincronizarlo, y crear un plan para las siguientes 24 horas. Esta colaboración es esencial para asegurar el progreso continuo y evadir cualquier obstrucción de trabajo.
- **Reunión de revisión del *sprint*:** su propósito es el de inspeccionar la iteración del producto que el equipo de desarrollo ha entregado, obtener una retroalimentación de los participantes en la reunión con respecto a la misma, y adaptar el plan para el *sprint* subsiguiente. Esta reunión está abierta para todo el personal dentro de la organización.
- **Reunión de retrospectiva:** es la reunión final del *sprint*, la cual nunca es omitida, sin importar lo que haya ocurrido en dicho *sprint*. Mientras que la reunión de revisión del *sprint* está enfocada en el producto, esta reunión está enfocada en el proceso, es decir, la forma en la que el equipo SCRUM está trabajando en conjunto, incluyendo sus habilidades técnicas, las prácticas de desarrollo del software y las herramientas que están usando. Esta reunión se limita a los miembros del equipo SCRUM.

Los Artefactos

- **Pila del producto (*product backlog*):** es una lista de ítems de trabajo descritos en un nivel funcional, que necesitan ser realizados a lo largo del tiempo. Los requerimientos son emergentes, lo que significa que no se puede saber por adelantado todos los detalles acerca de qué se quiere en el producto. Por esta razón este artefacto es un documento dinámico, que requiere un refinamiento constante para mantenerlo actual y útil.
- **Pila del *sprint* (*sprint backlog*):** esta pila es visualizada por el equipo de desarrollo en un *task board*, que es la representación física de la lista de trabajo que se ha resumido para realizar durante el *sprint* actual. Este artefacto le dice al equipo SCRUM y a todos los demás qué trabajo tienen planeado hacer en el *sprint*, y su estado actual.
- **Incremento:** es la suma de todos los ítems de la pila del producto que cumplen con la definición de terminado al final del *sprint*. El equipo de desarrollo presentará este en la revisión del *sprint*, y el *Product Owner* determinará cuándo liberar este incremento.

En esta metodología se pueden emplear varias técnicas y procesos. Dicho lo anterior, adicionalmente a la utilización de SCRUM, se incluyeron algunos artefactos de la metodología RUP (Rational Unified Process) descrita por Kroll y Kruchten (2003), para así generar suficiente documentación durante el diseño y el desarrollo del nuevo software.

3.2.2. Artefactos de RUP utilizados

- **Documento de visión:** describe la visión de los *stakeholders* con respecto al producto a desarrollarse, especificado en términos de las características y las necesidades clave de los mismos.
- **Modelo de casos de uso:** describe los requerimientos funcionales del

software en términos de actores y casos de uso. Un actor representa el tipo de usuario del software, mientras que un caso de uso describe cómo va a interactuar cada actor con el software.

- **Requerimientos no funcionales:** representan los requerimientos que tienen un impacto significativo en la arquitectura y en la satisfacción del usuario.
- **Glosario:** este define la terminología empleada en todos los artefactos utilizados.

Para finalizar, en la figura 3.2 es ilustrada la configuración de la metodología SCRUM utilizada, en conjunto con los artefactos elegidos de la metodología RUP.

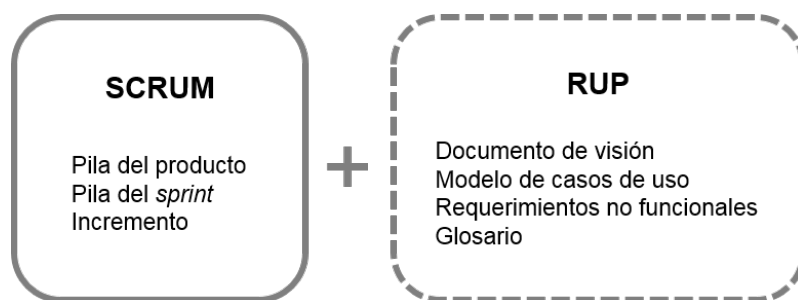


Figura 3.2: Configuración de los artefactos a utilizar de SCRUM y RUP (Fuente: Autor).

Capítulo IV

Resultados

En este capítulo se muestran los resultados que fueron alcanzados en el desarrollo del presente trabajo de investigación, incluyendo las fases metodológicas, las tecnologías y los recursos utilizados, la interfaz gráfica de usuario del software resultante y las pruebas realizadas.

4.1. Fases metodológicas

4.1.1. Visión

Enunciado del problema

El problema que se presenta es que se está utilizando el HunterLab Universal Software para el manejo del espectrofotómetro de reflexión difusa MiniScan XE Plus. Dicho software está en inglés, es comercial, privativo y fue descontinuado; esto afecta a los dermatólogos del Centro de Investigaciones Médicas y Biotecnológicas de la Universidad de Carabobo (CIMBUC).

El impacto causado por esto es que los dermatólogos encuentran el Hun-

terLab Universal Software difícil de utilizar, e imposible de adaptar a sus necesidades, lo que ralentiza la actividad de consulta con sus pacientes, genera la necesidad de disponer de personal especializado para su debido uso, y disminuye el potencial de dicho instrumento.

Una solución satisfactoria sería disponer de un software para el uso del MiniScan XE Plus que esté en español, que sea amigable y mantenible, permitiendo que se adapte a las necesidades de los dermatólogos.

Descripción de los usuarios

Tabla 4.1: *Actores del negocio* (Fuente: Autor).

Actor	Descripción
Administrador	Realiza mediciones. Consulta las historias médicas de los pacientes y las muestras. Gestiona los usuarios.
Dermatólogo	Realiza mediciones. Gestiona las historias médicas de los pacientes y las muestras. Realiza diagnósticos a partir de los resultados de las muestras.
Investigador	Realiza mediciones. Consulta las historias médicas de los pacientes y las muestras. Realiza análisis sobre los resultados de las mediciones.

Tabla 4.2: *Actores del software* (Fuente: Autor).

Actor	Responsabilidad	Experiencia	Uso
Administrador	Manejar el MiniScan XE Plus. Crear, consultar, modificar y eliminar usuarios. Consultar historias médicas de pacientes. Consultar muestras de pacientes.	Alta	Alto
Dermatólogo	Manejar el MiniScan XE Plus. Crear, consultar, modificar y eliminar historias médicas de pacientes. Crear, consultar, modificar y eliminar muestras de pacientes.	Baja	Alto
Investigador	Manejar el MiniScan XE Plus. Consultar historias médicas de pacientes. Consultar muestras de pacientes.	Media	Alto

Resumen del producto

El software desarrollado, denominado a partir de ahora Spectrasoft, es una aplicación para el uso del MiniScan XE Plus, la recuperación de los datos de medición de dicho instrumento, la generación de resultados relevantes y la gestión de los mismos, el cual está orientado a las actividades médicas dermatológicas del Centro de Investigaciones Médicas y Biotecnológicas de la Universidad de Carabobo (CIMBUC). La tabla 4.3 resume los beneficios y las características más importantes que provee el producto.

Tabla 4.3: *Beneficios y características principales del producto* (Fuente: Autor).

Beneficio al cliente	Característica que lo soporta
Se puede conectar, calibrar y realizar mediciones con el MiniScan XE Plus.	Comunicación con el MiniScan XE Plus y acceso a las características comunmente utilizadas por el mismo.
Se dispone de información relevante para el análisis y diagnóstico de patologías dermatológicas en la piel de los pacientes.	Muestra de los datos espectrales obtenidos de las mediciones, representación gráfica de dichos datos, y cálculo de valores adicionales asociados a los mismos.
Se pueden gestionar los usuarios, las historias médicas y las muestras generadas de las mediciones.	Manejo de una base de datos que almacena toda la información referente a los usuarios, las historias médicas y las muestras, permitiendo su gestión por medio del Spectrasoft.
El Spectrasoft se puede utilizar con facilidad.	Interfaz gráfica de usuario en español, que ofrece únicamente las funciones necesarias para gestionar la información que necesitan los usuarios.
El Spectrasoft se puede adaptar a las futuras necesidades de sus usuarios.	Código abierto del proyecto disponible en su totalidad para realizar cualquier modificación y/o extensión.

Principales restricciones

El software se desarrolla utilizando el lenguaje de programación C++, empleando únicamente tecnologías gratuitas, y en la medida de lo posible, de código abierto. Este se ejecuta en sistemas operativos Windows actuales. Por último, la comunicación entre el software y el MiniScan XE Plus se logra tanto por medio de un puerto serial, como por medio de un adaptador USB.

4.1.2. Pila del producto (*product backlog*)

La lista de requerimientos funcionales que necesitan ser realizados a lo largo del desarrollo del Spectrasoft, son el producto de una reunión que se tuvo con los dermatólogos y los investigadores del CIMBUC, al igual que de la observación directa efectuada a las funciones que el HunterLab Universal Software ofrece para el manejo del MiniScan XE Plus. En la tabla 4.4 se muestran dichos requerimientos.

Tabla 4.4: *Requerimientos funcionales del software* (Fuente: Autor).

Código	Requerimiento	Prioridad
RF01	Conectar y desconectar el MiniScan XE Plus.	Esencial
RF02	Calibrar el MiniScan XE Plus.	Esencial
RF03	Recuperar los 31 puntos espectrales de una medición con el MiniScan XE Plus y mostrarlos en su forma numérica.	Esencial
RF04	Graficar una curva de reflectancia difusa a partir de los 31 puntos espectrales recuperados.	Esencial
RF05	Graficar una curva de absorbancia aparente a partir de los 31 puntos espectrales recuperados.	Esencial
RF06	Calcular y mostrar las coordenadas de cromaticidad CIE xyz.	Esencial
RF07	Calcular y mostrar las coordenadas del espacio CIELAB.	Esencial
RF08	Calcular y graficar el coeficiente de absorción de la epidermis.	Esencial
RF09	Calcular y graficar el coeficiente de esparcimiento de la epidermis.	Esencial
RF010	Calcular y mostrar el índice de eritema.	Esencial
RF11	Almacenar la información de los usuarios, las historias médicas, las muestras y los resultados de las mediciones en una base de datos.	Esencial
RF12	Gestionar la creación, consulta, modificación y eliminación de los usuarios.	Esencial
RF13	Gestionar la creación, consulta, modificación y eliminación de las historias médicas de pacientes.	Esencial
RF14	Gestionar la creación, consulta, modificación y eliminación de las muestras pertenecientes a las historias médicas.	Esencial
RF15	Exportar los datos de una muestra a un archivo portátil.	Esencial

4.1.3. Requerimientos no funcionales

De la misma forma que los requerimientos funcionales, la lista de los requerimientos no funcionales fue definida en la misma reunión con los dermatólogos y los investigadores, tomando en cuenta las restricciones del entorno en donde se va a ejecutar el Spectrasoft. En la tabla 4.5 se describen estos requerimientos no funcionales.

Tabla 4.5: *Requerimientos no funcionales del software* (Fuente: Autor).

Código	Requerimiento	Prioridad
RNF01	El software debe ser fácil de utilizar, por lo que debe cumplir con el atributo de usabilidad de un software de calidad.	Esencial
RNF02	El software debe ser capaz de adaptarse a las necesidades de los dermatólogos, razón por la cual debe cumplir con el atributo de mantenibilidad de un software de calidad.	Esencial
RNF03	El software debe desarrollarse empleando únicamente herramientas y tecnologías gratuitas, y en la medida de lo posible, de código abierto.	Esencial
RNF04	El software debe ser capaz de ejecutarse en sistemas Windows actuales, con arquitecturas de 32 bits y 64 bits.	Esencial
RNF05	El software debe conectarse con el MiniScan XE Plus por medio de un puerto serial o de un adaptador USB.	Esencial
RNF06	El archivo portátil al que se exportan los resultados de una medición debe ser abierto por un visualizador/editor de hojas de cálculo.	Esencial
RNF07	El software debe desarrollarse utilizando el lenguaje de programación orientada a objetos C++.	Esencial

4.1.4. Casos de uso

■ Manejar el MiniScan XE Plus:

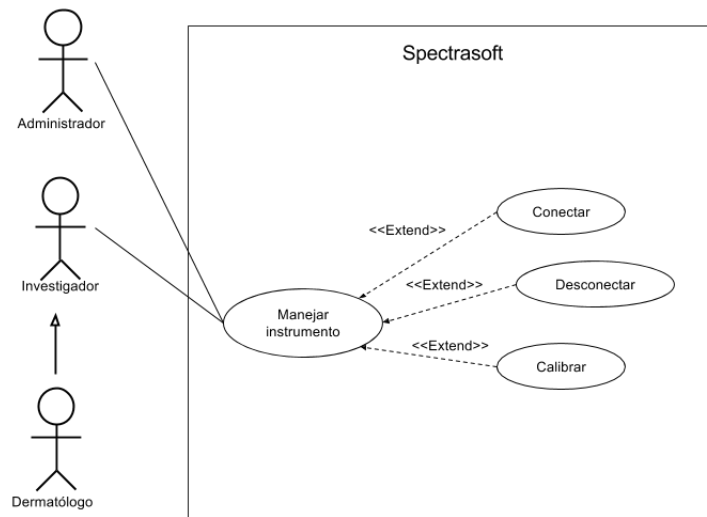


Figura 4.1: Caso de uso: manejar el MiniScan XE Plus (Fuente: Autor).

■ Gestionar medición:

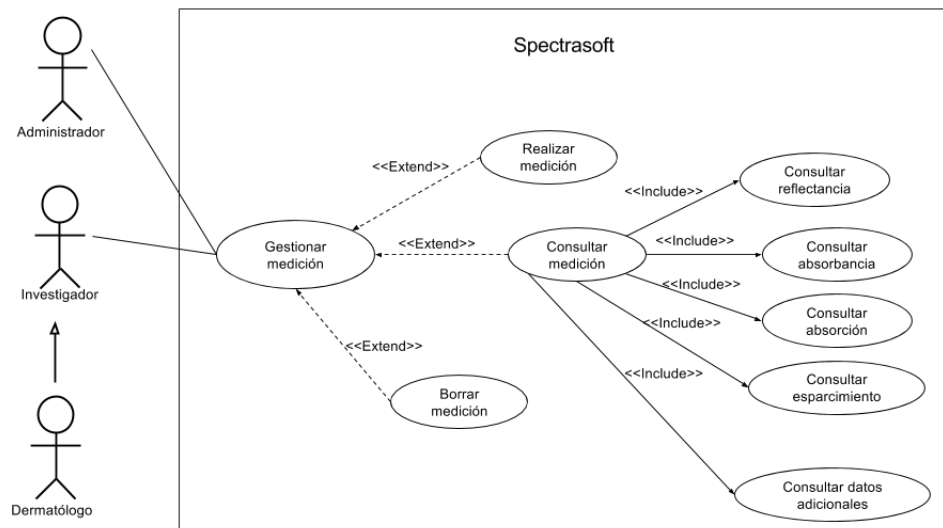


Figura 4.2: Caso de uso: gestionar medición (Fuente: Autor).

■ Gestionar sesión:

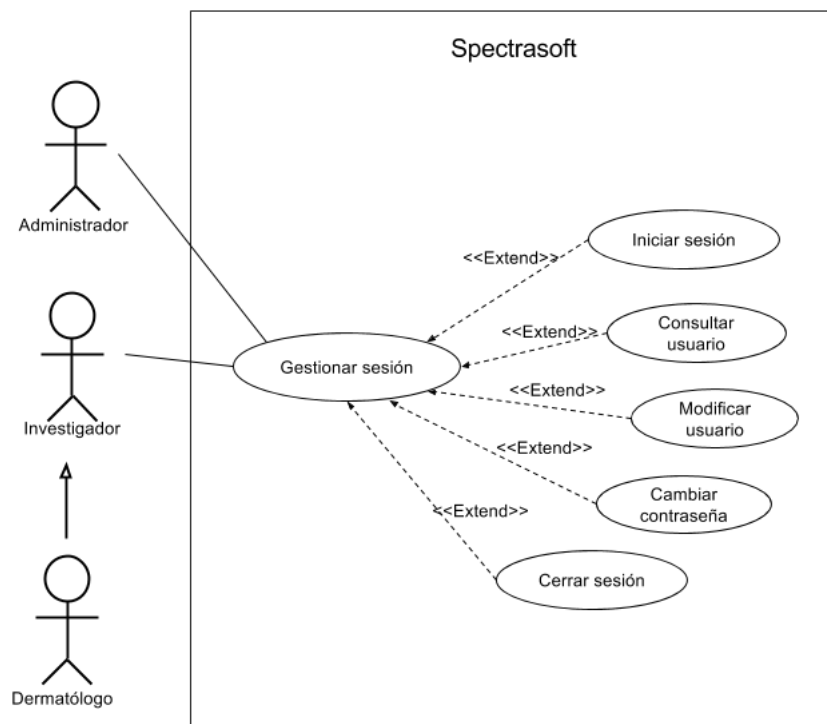


Figura 4.3: Caso de uso: *gestionar sesión* (Fuente: Autor).

■ Gestionar historia:

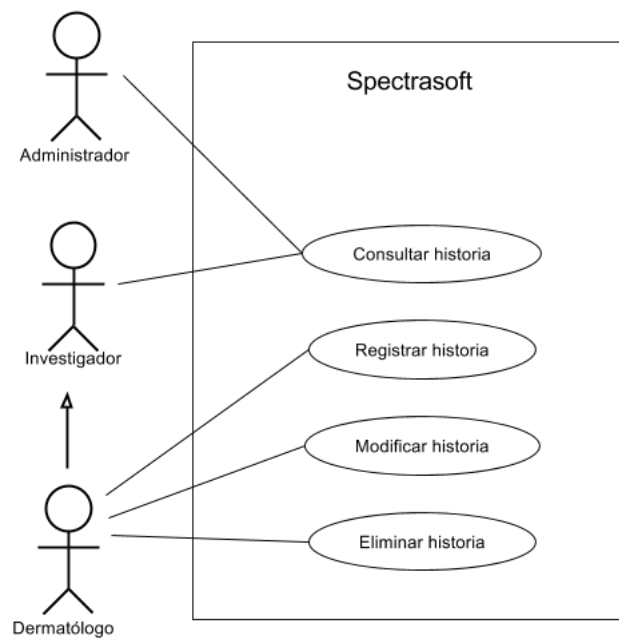


Figura 4.4: Caso de uso: *gestionar historia* (Fuente: Autor).

■ Gestionar muestra:

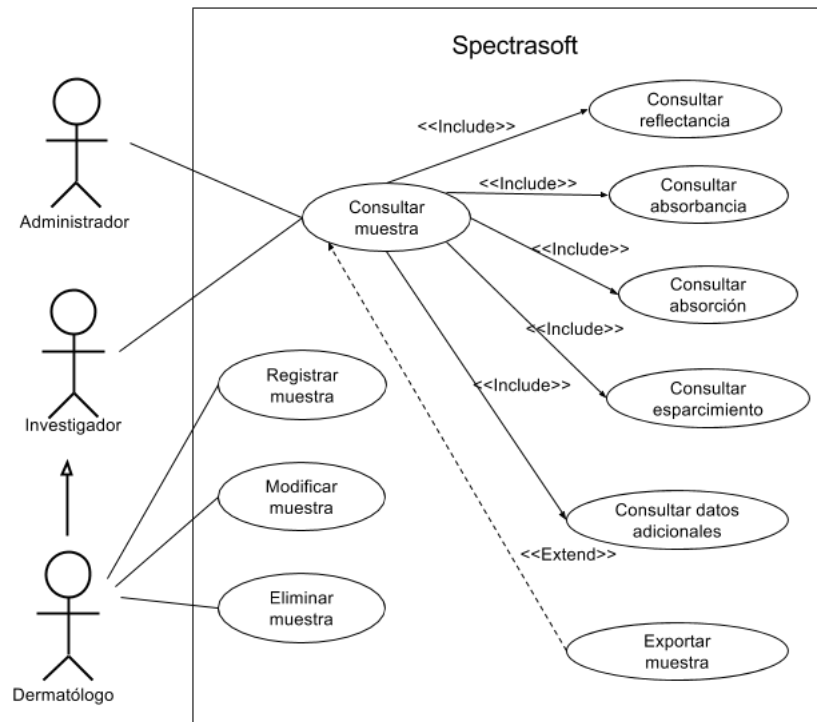


Figura 4.5: Caso de uso: gestionar muestra (Fuente: Autor).

■ Gestionar usuario:

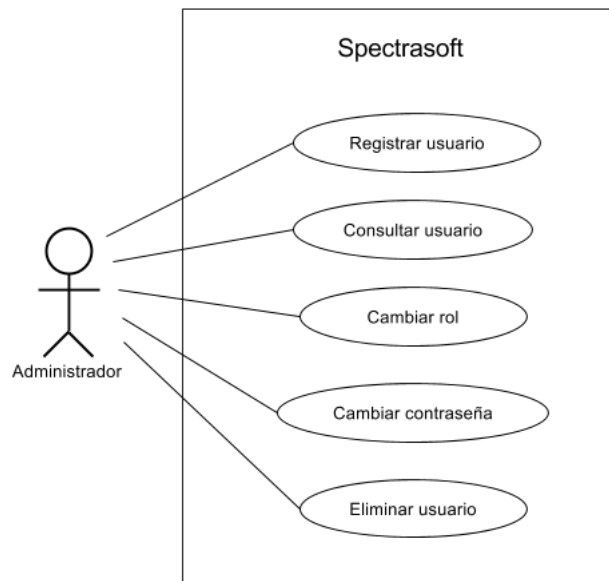


Figura 4.6: Caso de uso: gestionar usuario (Fuente: Autor).

Descripción de los casos de uso

- **Manejar instrumento:** Permite a todos los usuarios conectar, desconectar y calibrar el MiniScan XE Plus, esto último haciendo uso de una trampa de luz y una cerámica blanca.
- **Gestionar medición:** Permite a todos los usuarios efectuar una medición con el MiniScan XE Plus conectado, visualizar los resultados obtenidos de la misma, y borrarlos.
- **Consultar medición:** Permite a todos los usuarios visualizar tanto la información obtenida directamente del MiniScan XE Plus, como la información calculada a partir de la misma, como la curva de reflectancia difusa, la curva de absorbancia aparente, las coordenadas de cromaticidad CIE xyz, las coordenadas CIELAB y el índice de eritema de una medición realizada.
- **Gestionar sesión:** Permite a cualquier usuario iniciar sesión para acceder a las funciones pertinentes para su rol, consultar y modificar su información, cambiar su contraseña, y cerrar sesión.
- **Gestionar historia:** Permite a todos los usuarios consultar la información perteneciente a las historias médicas, y sólo permite a los usuarios con el rol de dermatólogo registrar, modificar y eliminar dichas historias.
- **Gestionar muestra:** Permite a todos los usuarios consultar la información referente a las muestras pertenecientes a una historia médica de un paciente determinado, así como exportar dicha información, y sólo permite a los usuarios con el rol de dermatólogo registrar, modificar y eliminar tales muestras.
- **Gestionar usuario:** Permite registrar usuarios, consultar su información, cambiar los roles de dichos usuarios, cambiar sus contraseñas y eliminarlos. Esto solamente puede ser efectuado por usuarios con el rol de administrador.

4.1.5. Glosario

- Espectrofotómetro de reflexión difusa:
- Software:
- Software privativo:
- Medición:
- Calibración:
- Usuario:
- Rol de usuario:
- Sesión:
- Historia médica:
- Muestra:
- Patología dermatológica:
- Datos espectrales:
- Base de datos:
- Interfaz gráfica de usuario:
- Código abierto:
- Lenguaje de programación:
- Sistema operativo:
- Puerto serial:
- Adaptador USB:
- Reflectancia difusa:
- Absorbancia aparente:

- Coordenadas de cromaticidad CIE xyz:
- Coordenadas del espacio CIELAB:
- Epidermis:
- Coeficiente de absorción:
- Coeficiente de esparcimiento:
- Índice de eritema:
- Hoja de cálculo:

4.2. Base de datos

4.2.1. Diagrama ER de la base de datos

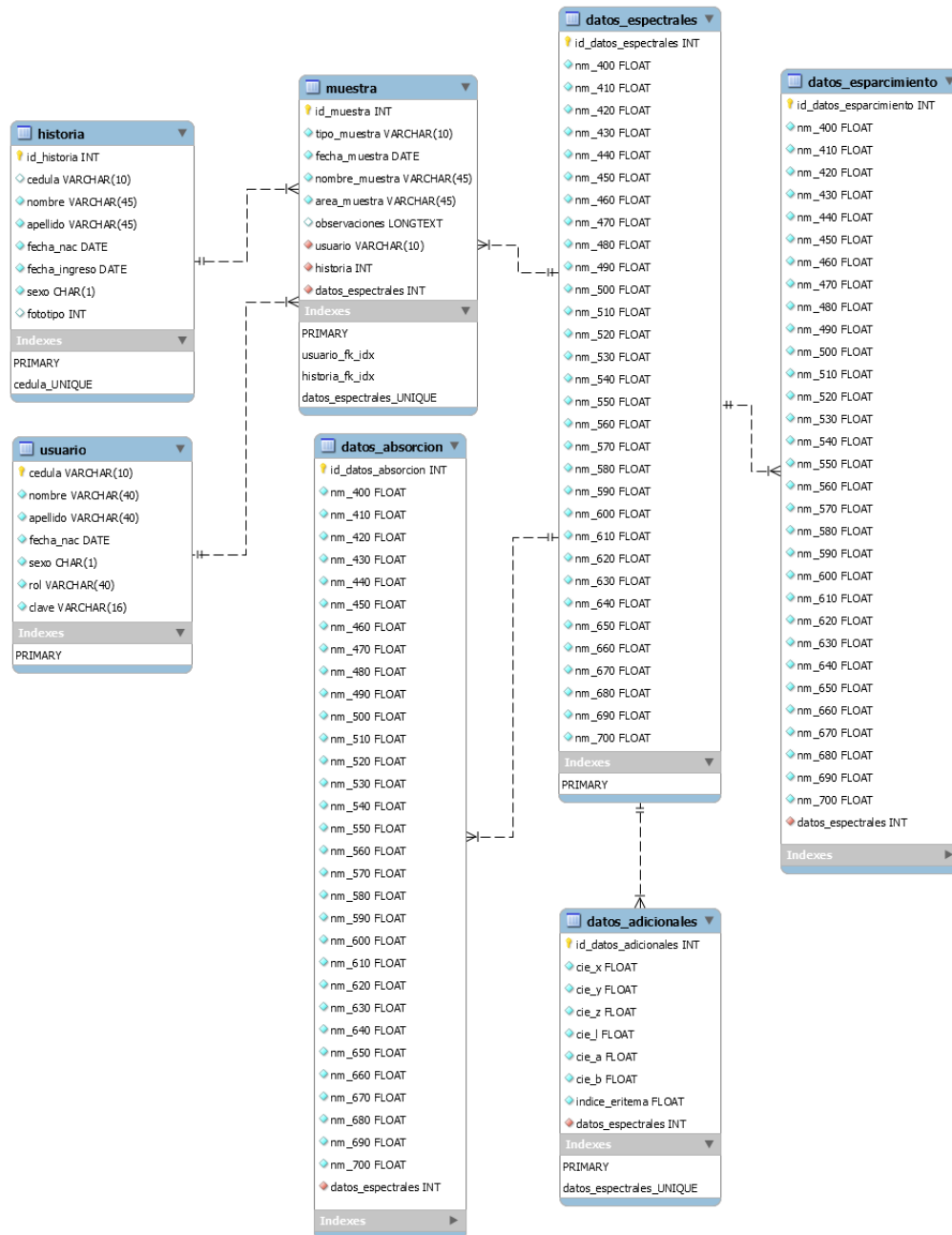


Figura 4.7: Diagrama ER de la base de datos (Fuente: Autor).

4.2.2. Descripción de las tablas de la base de datos

- **historia:** almacena los datos referentes a la historia médica de cada uno de los pacientes registrados.
- **usuario:** guarda la información de cada uno de los usuarios que pueden acceder al software, que pueden ser administradores, dermatólogos o investigadores.
- **muestra:** contiene los datos relevantes de las muestras que son tomadas a los pacientes. Dichas muestras siempre están interrelacionadas con la historia médica del paciente al que pertenece, y la cédula del usuario que la tomó.
- **datos_espectrales:** contiene los 31 puntos espectrales resultantes de la medición realizada de cada muestra por medio del MiniScan XE Plus.
- **datos_absorcion:** contiene los 31 puntos espectrales resultantes del cálculo del coeficiente de absorción asociado con los datos espectrales.
- **datos_esparcimiento:** contiene los 31 puntos espectrales resultantes del cálculo del coeficiente de esparcimiento asociado con los datos espectrales.
- **datos_adicionales:** almacena los datos que son calculados a partir de los 31 puntos espectrales de cada muestra, estos son las coordenadas de cromaticiad CIE xyz, las coordenadas del espacio del color CIELAB y el índice de eritema.

4.3. Tecnologías y recursos utilizados

4.3.1. Tecnologías

- **Qt:** es un *framework* de desarrollo de aplicaciones multiplataforma para sistemas operativos de escritorio, sistemas integrados y sistemas móviles.

Se utilizó la versión *open source* 5.4.1 de este *framework* para el desarrollo del Spectrasoft.

- **Visual Studio:** es un entorno integrado de desarrollo o *IDE* para crear aplicaciones en varias plataformas, como Windows, Android y iOS. La versión 2013 de este *IDE* fue utilizada para desarrollar una librería escrita en Visual Basic.NET, la cual actúa como intermediaria entre el kit *MSXE.ocx* y el *framework* Qt, para así utilizar las características del MiniScan XE Plus en el Spectrasoft.
- **PostgreSQL:** es un sistema *open source* multiplataforma de bases de datos relacionales. Posee más de 15 años de desarrollo activo y una arquitectura comprobada que se ha ganado una fuerte reputación por confiabilidad, integridad de datos y correctitud. Este sistema se utilizó para desarrollar y administrar la base de datos con la que opera el Spectrasoft.
- **Gitlab:** es un servicio de control de versiones que ofrece alojamiento gratuito, tanto público como privado, de repositorios para proyectos. Se utilizó la versión en línea de este servicio para llevar un control de versiones durante el desarrollo del proyecto.
- **QCustomPlot:** es un *widget open source* para Qt que permite realizar el trazado y la visualización de datos. Este *widget* fue empleado por el Spectrasoft para visualizar la curva de reflectancia difusa y la curva de absorbancia aparente asociadas a los 31 puntos espectrales resultantes de las mediciones.
- **QtXlsx:** es una librería *open source* para Qt que permite leer y escribir archivos con extensión *xlsx*. Esta librería fue utilizada para implementar en el Spectrasoft la opción de exportar los resultados de una muestra a un archivo portátil, manejable por medio de aplicaciones de hojas de cálculo.

4.3.2. Recursos

- **MiniScan XE Plus:** es un instrumento de medición del color creado por la empresa HunterLab, de diseño compacto y portátil, que emplea la técnica de espectroscopía de reflectancia difusa, el cual se puede apreciar en la figura 4.8. Este instrumento mide la cantidad de luz que refleja una muestra dentro de un rango de longitudes de onda que va desde los 400 hasta los 700 nanómetros, generando como resultado 31 puntos espectrales dentro de ese rango, que son el insumo principal del Spectrasoft.
- **Adaptador RS232-USB:** es un cable adaptador que habilita la comunicación de dispositivos que utilizan puerto serial con computadoras que disponen de puertos USB, creando puertos COM virtuales con las mismas mientras se realiza dicha comunicación. Este cable es utilizado como adaptador para el cable de comunicación RS232 DB-9 hembra a RJ-45 del MiniScan XE Plus, y así habilitar la utilización de este instrumento en computadoras que no poseen puerto serial.

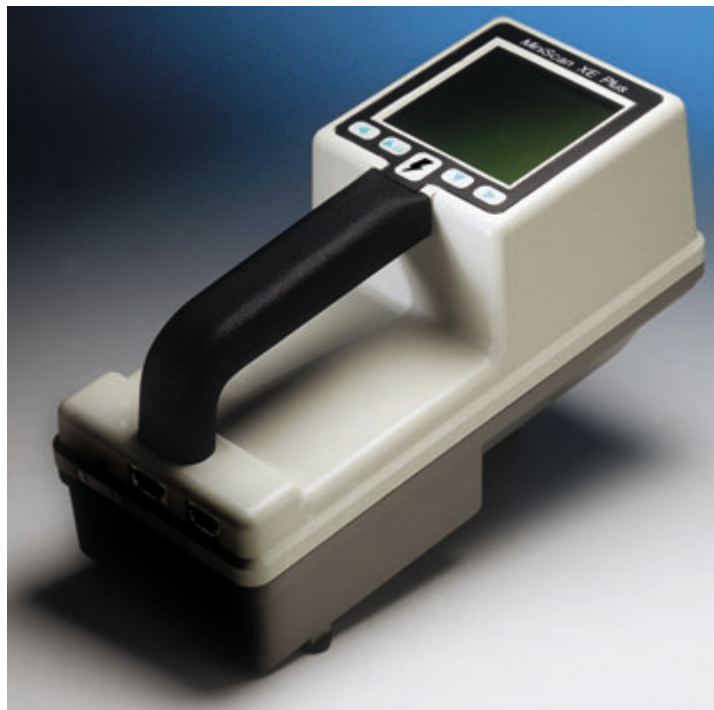


Figura 4.8: *MiniScan XE Plus* (Fuente: HunterLab, 2006).

4.4. Comunicación con el MiniScan XE Plus

Para establecer la comunicación entre el MiniScan XE Plus y el Spectrasoft, se recurrió a la documentación del instrumento, en la cual se describe el MSXE.ocx, que es un archivo que implementa las funciones comunmente utilizadas por dicho instrumento. Se contactó al personal de soporte técnico de HunterLab por correo electrónico, para solicitarle el código fuente de dicho archivo y la documentación relativa a su utilización que se pudiera proporcionar para la investigación.

Si bien el personal no compartió el código fuente del archivo, sí envió la documentación solicitada y un ejemplo de su uso escrito en Visual Basic for Applications (VBA). Primero se intentó cargar el archivo y utilizarlo directamente en Qt; sin embargo, ocurría un error de compatibilidad de datos al invocar algunas de sus funciones. La solución a este problema fue desarrollar una librería escrita en Visual Basic .NET, con la cual se pueden invocar todas las funciones de este archivo sin problema alguno.

Así pues, por medio del cable adaptador RS232-USB, y empleando la librería escrita en Visual Basic .NET, se logró establecer la comunicación entre el software que está siendo desarrollado en Qt y el MiniScan XE Plus, por medio del archivo MSXE.ocx.

4.5. Fórmulas implementadas

A continuación se muestra el código fuente de las fórmulas de óptica y colorimetría que fueron definidas en las bases teóricas, e implementadas en el Spectrasoft.

- **Función de índice de eritema:**

```
1 //Calcula el indice de eritema
2 float eritema(QVector<float> medicion){
3     //promRojo: promedio ponderado del color rojo
4     float promRojo = (medicion[24]/2.0 + medicion[25]
5         + medicion[26] + medicion[27]/2.0) /3.0;
6
7     //promVerde: promedio ponderado del color verde
8     float promVerde = (medicion[16]/2.0 + medicion[17]
9         + medicion[18]/2.0) /2.0;
10
11     float resultado = 100.0*(log(1.0/promVerde)
12         - log(1.0/promRojo));
13
14     return resultado;
15 }
```

■ **Función de absorbancia aparente:**

```
1 //Calcula los datos de absorbancia aparente
2 QVector<float> absorbancia(QVector<float> medicion){
3
4     QVector<float> resultado;
5
6     for(int i = 0; i < 31; ++i)
7         resultado.push_back(100.0 - medicion[i]);
8
9     return resultado;
10 }
```

■ **Función de coordenadas de cromaticidad CIE xyz:**

```
1 //Calcula las coordenadas de cromaticidad CIE xyz
2 QVector<float> CIExyz(QVector<float> medicion){
3
4     QVector<float> resultado;
5     QVector<float> XYZ = CIEXYZ(medicion);
6     float x, y, z;
7
8     x = XYZ[0]/(XYZ[0] + XYZ[1] + XYZ[2]);
9     y = XYZ[1]/(XYZ[0] + XYZ[1] + XYZ[2]);
10    z = XYZ[2]/(XYZ[0] + XYZ[1] + XYZ[2]);
11
12    resultado.push_back(x);
13    resultado.push_back(y);
14    resultado.push_back(z);
15
16    return resultado;
17 }
```

■ Función de valores triestímulo CIE XYZ:

```

1 //Calcula los valores triestímulo CIE XYZ
2 QVector<float> CIEXYZ(QVector<float> medicion){
3
4     QVector<float> resultado;
5     float auxK, auxX, auxY, auxZ, k, X, Y, Z;
6
7     auxK = auxX = auxY = auxZ = 0.0;
8
9     //realiza las sumatorias indicadas de las formulas
10    for(int i = 0; i < 31; ++i){
11
12        auxK+= iluCIED65[i]*yCIE10[i];
13        auxX+= medicion[i]*iluCIED65[i]*xCIE10[i];
14        auxY+= medicion[i]*iluCIED65[i]*yCIE10[i];
15        auxZ+= medicion[i]*iluCIED65[i]*zCIE10[i];
16    }
17
18    //calcula la constante k
19    k = 100.0/auxK;
20
21    //calcula los valores triestímulo XYZ
22    X = k*auxX;
23    Y = k*auxY;
24    Z = k*auxZ;
25
26    resultado.push_back(X);
27    resultado.push_back(Y);
28    resultado.push_back(Z);
29
30    return resultado;
31 }

```

■ **Función de coordenadas CIELAB:**

```

1 //Calcula las coordenadas de del espacio CIELAB
2 QVector<float> CIELAB(QVector<float> medicion){
3
4     QVector<float> resultado;
5     QVector<float> XYZ = CIEXYZ(medicion);
6     float constante, aux, fXfYfZ[3], L, a, b;
7
8     //calcula la constante utilizada en la formula
9     constante = 24.0/116.0;
10    constante = pow(constante, 3);
11
12    //calcula las funciones X/Xn, Y/Yn, Z/Zn
13    for(int i = 0; i < 3; ++i){
14
15        aux = XYZ[i]/XnYnZn[i];
16
17        if(aux > constante){
18            fXfYfZ[i] = pow(aux, 1.0/3.0);
19        }else{
20            fXfYfZ[i] = (841.0/108.0)*aux + (16.0/116.0);
21        }
22    }
23
24    //calcula las coordenadas L*a*b*
25    L = 116.0*fXfYfZ[1] - 16.0;
26    a = 500.0*(fXfYfZ[0] - fXfYfZ[1]);
27    b = 200.0*(fXfYfZ[1] - fXfYfZ[2]);
28
29    resultado.push_back(L);
30    resultado.push_back(a);
31    resultado.push_back(b);
32
33    return resultado;
34 }
```

■ Función de coeficiente de absorción:

```

1 //Calcula el coeficiente de absorcion
2 QVector<float> absorcion(QVector<float> medicion){
3
4     QVector<float> resultado;
5     float rango, aux, Z, k, a0;
6     int n, grado;
7     bool op = false; //el sistema de ecuaciones a resolver ←
        tiene solucion
8
9     //son 31 datos, y el grado del polinomio de ajuste es 8
10    n = 31; grado = 8;
11
12    double *x = new double[n];
13    double *y = new double[n];
14
15    //cargando los datos en x, y
16    for(int i = 0; i < n; ++i){
17        x[i] = i;
18        y[i] = medicion.at(i);
19    }
20
21    double **matriz = new double*[grado + 1];
22
23    for (int i = 0; i < grado + 1; ++i)
24        matriz[i] = new double[grado + 2];
25
26    //realizando el ajuste polinomial y calculando los ←
        coeficientes
27    coeficientes(x, y, matriz, grado, n);
28    recorrido(matriz, grado + 1, op);
29
30    //a0 esta en la primera fila de la matriz resultante ←
        del ajuste
31    a0 = matriz[0][grado + 1];
32    Z = 0.2796; k = -7.174;

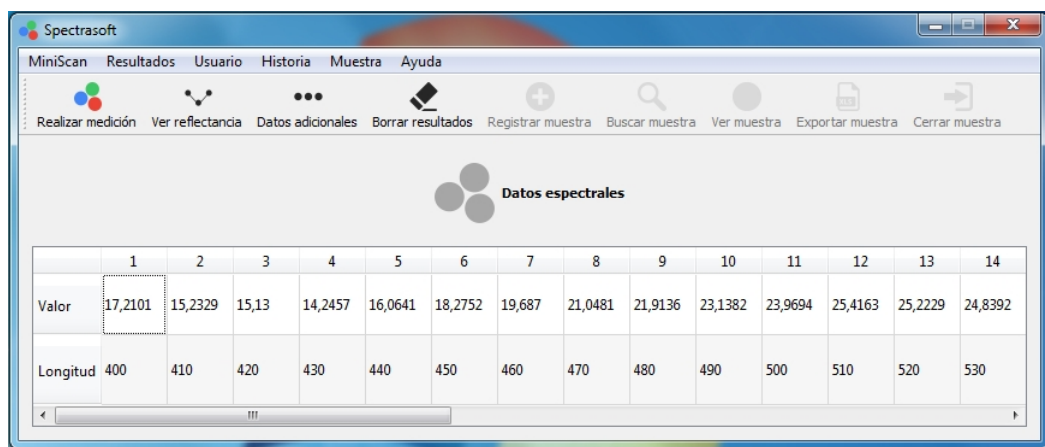
```

```
33     rango = 400;
34
35     //calculando el coeficiente de absorcion
36     for(int i = 0; i < n; ++i){
37         aux = Z*exp(k*a0)*6.6*pow(10, 11)*pow(rango, -3.3);
38         resultado.push_back(aux);
39         rango+=10;
40     }
41
42     //liberando la memoria dinamica asignada
43     delete x;
44     delete y;
45
46     for(int i = 0; i < grado + 1; ++i)
47         delete matriz[i];
48
49     delete matriz;
50
51     return resultado;
52 }
```

4.6. Interfaz del Spectrasoft

A continuación se muestran algunas de las vistas de la interfaz gráfica de usuario del Spectrasoft, que comprenden la vista principal, la ventana de inicio de sesión de un usuario, la visualización de algunos de los resultados de una medición realizada, y los detalles de la historia médica de un paciente.

■ Vista principal:

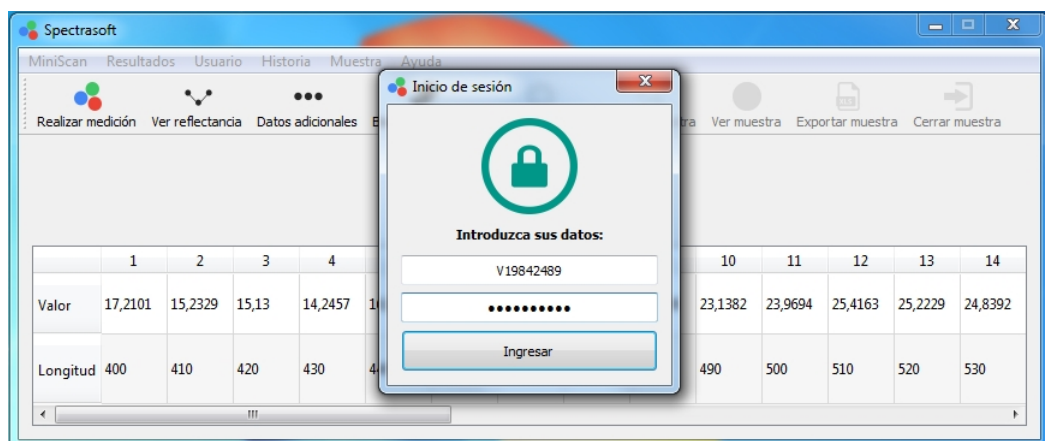


The screenshot shows the main window of the Spectrasoft application. It has a menu bar with 'MiniScan', 'Resultados', 'Usuario', 'Historia', 'Muestra', and 'Ayuda'. Below the menu is a toolbar with icons for 'Realizar medición', 'Ver reflectancia', 'Datos adicionales', 'Borrar resultados', 'Registrar muestra', 'Buscar muestra', 'Ver muestra', 'Exportar muestra', and 'Cerrar muestra'. The main area is titled 'Datos espectrales' and contains a table with 14 columns and 3 rows of data.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Valor	17,2101	15,2329	15,13	14,2457	16,0641	18,2752	19,687	21,0481	21,9136	23,1382	23,9694	25,4163	25,2229	24,8392
Longitud	400	410	420	430	440	450	460	470	480	490	500	510	520	530

Figura 4.9: *Vista principal* (Fuente: Autor).

■ Inicio de sesión:



The screenshot shows the login window of the Spectrasoft application overlaid on the main interface. The login window has a title bar 'Inicio de sesión' and a lock icon. It contains the text 'Introduzca sus datos:' followed by two input fields: one for a username (containing 'V19842489') and one for a password (containing dots). Below the input fields is an 'Ingresar' button.

Figura 4.10: *Inicio de sesión* (Fuente: Autor).

- Curva de reflectancia:

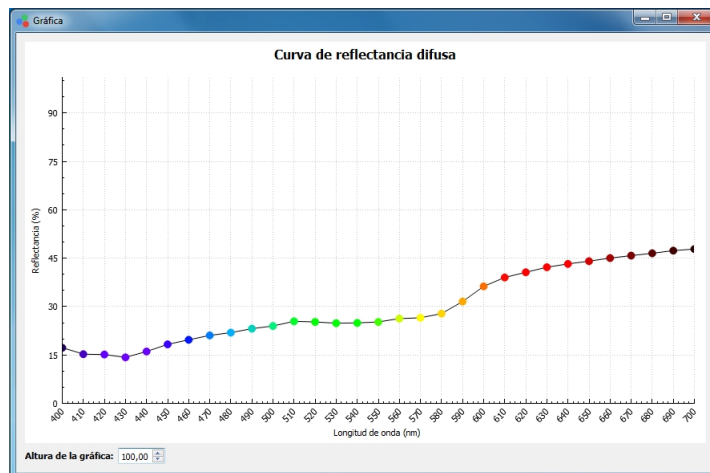


Figura 4.11: *Curva de reflectancia* (Fuente: Autor).

- Historia médica de un paciente:

Ver historia

Ver historia

Número de la historia: 26

Nombre(s): JOSE

Apellido(s): GUILLERMO

Cédula de identidad: N/A

Fecha de nacimiento: 11-10-2010

Fecha de ingreso: 11-10-2015

Sexo: Masculino

Fototipo cutáneo: III

○

Listo

Figura 4.12: *Historia médica de un paciente* (Fuente: Autor).

4.7. Pruebas realizadas

Capítulo V

Conclusiones y recomendaciones

5.1. Conclusiones

5.2. Recomendaciones

Referencias

- (2010). «*Absorption spectrum. (n.d.) Random House Kernerman Webster's College Dictionary*». K Dictionaries LTD.
- BASKERVILLE, RICHARD L. (1999). «*Investigating Information Systems with Action Research*». Association for Information Systems, Atlanta – Georgia, Estados Unidos.
- BERSHA, K. S. (2010). «*Spectral Imaging And Analysis Of Human Skin*». Universidad del Este de Finlandia, Finlandia.
- CIE (2004). «*CIE 15: Technical Report: Colorimetry, 3rd edition*». Viena, Austria.
- (2015). «*Commission Internationale de l'Eclairage, International Commission on Illumination*». Viena, Austria.
<http://www.cie.co.at/index.php>
- GITLAB. «*Gitlab: Create, review, and deploy code together*». <https://about.gitlab.com/>
- HUNDERMARK, P. (2015). «*Do Better SCRUM*». Agile42 - The Agile Coaching Company.
- HUNTERLAB (2001). «*Universal Software Versions 4.10 and Above User's Manual*». Hunter Associates Laboratory, Reston – Virginia, Estados Unidos.

— (2006). «*MiniScan XE Plus User's Guide Version 2.4*». Hunter Associates Laboratory, Reston – Virginia, Estados Unidos.

— (2008). «*Applications Note*». Hunter Associates Laboratory, Reston – Virginia, Estados Unidos.

— (2015). «*HunterLab, The World's true measure of color*». Hunter Associates Laboratory, Reston – Virginia, Estados Unidos.

<http://www.hunterlab.com/about-us.html>

KROLL, P. y KRUCHTEN, P. (2003). «*The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*». Addison-Wesley, Boston – Massachusetts, Estados Unidos.

MICROSOFT. «*Visual Studio Community, a fully-featured, extensible IDE*».

<https://www.visualstudio.com/products/visual-studio-community-vs>

NAREA, F.; VIVAS, S. y MUÑOZ, A. (2015). «*Recuperación del coeficiente de absorción de la epidermis en la piel humana*». Sociedad Española de Óptica, Universidad de Carabobo, Venezuela.

PÉREZ, A. D. (2012). «*Estudio de la Reflexión Óptica Difusa en Tejido Biológico*». Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco, México.

POSTGRESQL. «*PostgreSQL: The world's most advanced open source database*».

<http://www.postgresql.org/about/>

QCUSTOMPLOT. «*A Qt C++ widget for plotting and data visualization*».

<http://www.qcustomplot.com/index.php/introduction>

QT. «*Qt, a Cross-Platform Framework for Application Development*».

https://wiki.qt.io/About_Qt

QTXLSX. «*A Qt library to read and write excel files*».

<http://www.qtxlsx.debao.me/>

RESEARCH, MAGNETO TECH. «*USB to Serial adapters Wiki*».

<http://www.usb-serial-adapter.org/>

SCHANDA, J. (2007). «*Colorimetry: understanding the CIE system*». John Wiley & Sons, Hoboken – Nueva Jersey, Estados Unidos.

SCHWABER, K. y SUTHERLAND, J. (2013). «*The Definitive Guide to Scrum: The Rules of the Game*».

SOMMERVILLE, I. (2005). «*Ingeniería del Software*». Pearson Education, Madrid, España.

WAGNER, J.; JOVEL, C.; NORTON, H.; PARRA, E. y SHRIVER, M. (2002). «*Comparing Quantitative Measures of Erythema, Pigmentation and Skin Response using Reflectometry*». Pigment Cell Res, Universidad del Estado de Pensilvania, Estados Unidos.